Automatic Video-Object Segmentation for MPEG-4 Coding and Object-Behaviour Analysis

Dirk Farin, Wolfgang Effelsberg
Gerald Kühne, Stephan Kopf, Thomas Haenselmann

Contact address:
Dirk Farin
University of Mannheim
Dept. of Computer Science IV
L 15,16, 68131 Mannheim, Germany
farin@informatik.uni-mannheim.de

demo 0

Introduction

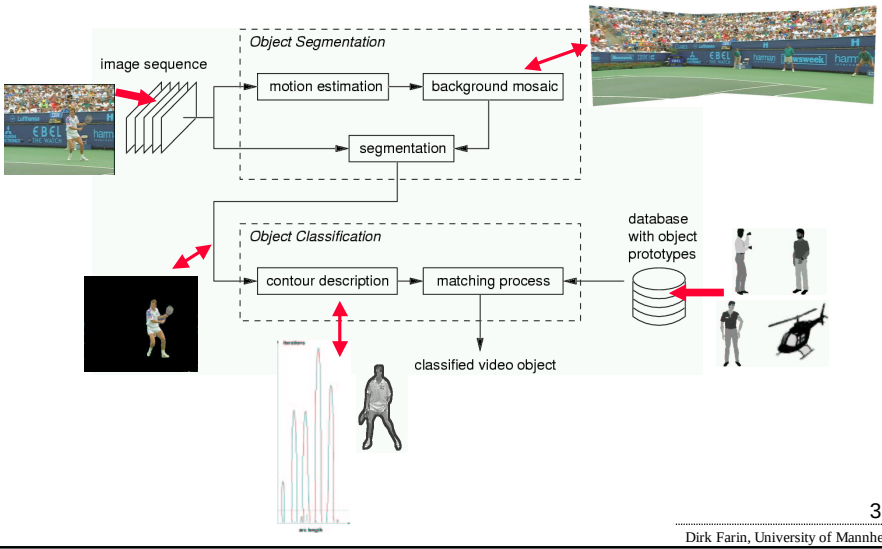
- Motivation: separate video sequence into video objects.

input video

automatically extracted foreground object

- Applications:
  - MPEG-4 sprite coding
    - higher coding efficiency (background is only transmitted once)
    - background replacement
  - behaviour analysis

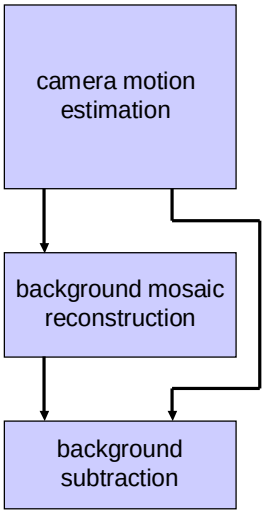
Segmentation and Classification Overview



3

Dirk Farin, University of Mannheim

Segmentation Algorithm



1. short term prediction (feature based)
   - corner extraction
   - robust motion estimation (LTS)
2. long term prediction (dense)
   - gradient descent
3. moving object(s) removal
   - e.g., temporal median filtering
4. differencing
5. regularization

4

Dirk Farin, University of Mannheim

Presentation Outline

- automatic segmentation
  - camera motion model
  - corner feature extraction
  - feature-based motion estimation
  - dense motion estimation
- background mosaic reconstruction
- background subtraction
  - shape regularization with Markov Random Fields
- MPEG-4 coding application
- Object behaviour analysis
  - object silhouette description
  - matching to behaviour model

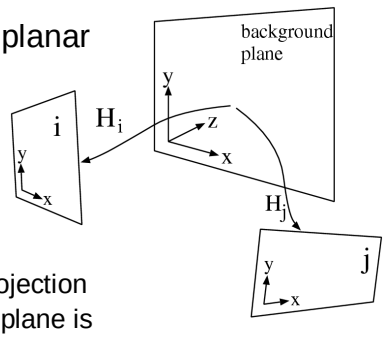
5

Dirk Farin, University of Mannheim

Camera Motion: Motion Model 1

Assumption: video background is planar

➔ choose coordinate system such that background plane is located at z=0.



Using homogeneous coordinates, the projection from the background plane to the image plane is

$$\begin{pmatrix} x' \\ y' \\ w \end{pmatrix} = \begin{pmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{pmatrix} \begin{pmatrix} x \\ y \\ 0 \\ 1 \end{pmatrix} = \underbrace{\begin{pmatrix} p_{11} & p_{12} & p_{14} \\ p_{21} & p_{22} & p_{24} \\ p_{31} & p_{32} & p_{34} \end{pmatrix}}_{H_i} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

6

Dirk Farin, University of Mannheim

Camera Motion: Motion Model 2

Transformation between two input frames:

$$H_{ij} = H_j H_i^{-1} = \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{pmatrix} \qquad H'_{ij} = H_{ij} / h_{33} = \begin{pmatrix} a_{11} & a_{12} & t_x \\ a_{21} & a_{22} & t_y \\ p_x & p_y & 1 \end{pmatrix}$$

scaling invariance !

and we can write explicitly

$$x' = \frac{a_{11}x + a_{12}y + t_x}{p_x x + p_y y + 1}, \qquad y' = \frac{a_{21}x + a_{22}y + t_y}{p_x x + p_y y + 1}.$$

- Motion model for MPEG-4 GMC, sprite warping.
- Compatible motions:
  - planar background, arbitrary camera motion, or
  - rotating/zooming camera, arbitrary background depth.

Camera Motion: Motion Estimation Principle

First phase: **feature-based** motion estimation

- short term prediction
  - motion between successive frames
- can handle large displacements
- robust estimation, insensitive to local minima
- fast approximate solution

Second phase: **dense** motion estimation

- long term prediction
  - registration to background mosaic
  - prevents error accumulation
- locks to local minimum
- accurate estimation (sub-pixel accuracy)

Presentation Outline

- automatic segmentation
  - camera motion model
  - corner feature extraction
  - feature-based motion estimation
  - dense motion estimation
- background mosaic reconstruction
- background subtraction
  - shape regularization with Markov Random Fields
- MPEG-4 coding application
- Object behaviour analysis
  - object silhouette description
  - matching to behaviour model

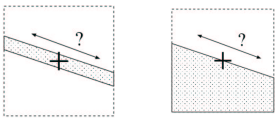
9

Dirk Farin, University of Mannheim

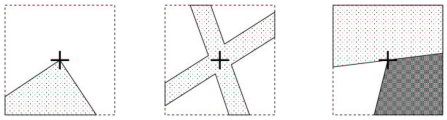
Corner Localization for Reliable Motion Estim.

- No motion information can be obtained for regions with low texture.

- Along edges, only one motion-vector component is reliable (perpendicular to edge).



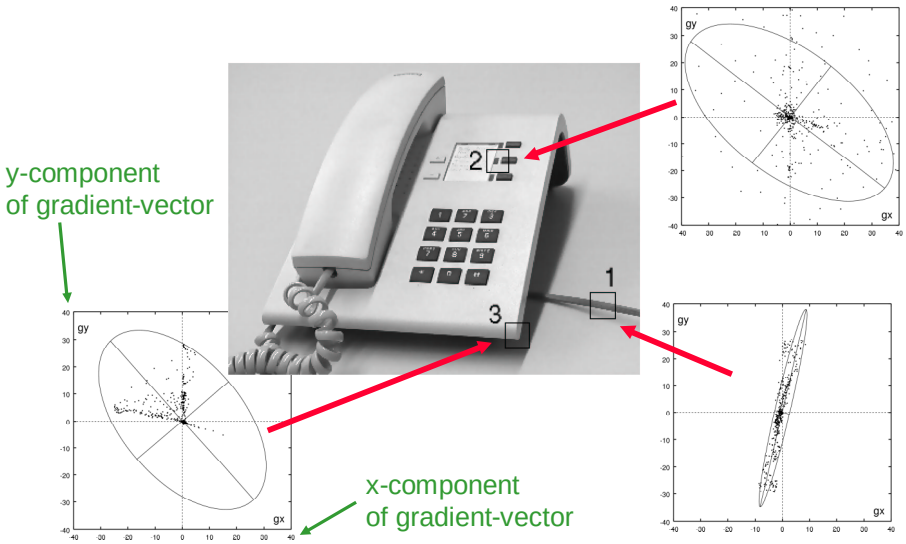
- Only use motion-vectors located at „corners“.
  - strong texture variation in two directions



10

Dirk Farin, University of Mannheim

Gradient-Vector Distribution Examples



11

Dirk Farin, University of Mannheim

Principal axes of gradient-vector distribution

- Compute center-weighted correlation matrix of gradient-vector components.

$$C = \begin{pmatrix} \sum_{i \in N(p)} w(i) g_x(i) g_x(i) & \sum_{i \in N(p)} w(i) g_x(i) g_y(i) \\ \sum_{i \in N(p)} w(i) g_x(i) g_y(i) & \sum_{i \in N(p)} w(i) g_y(i) g_y(i) \end{pmatrix}$$

gradient-vector components

Gaussian weighting kernel

pixel neighborhood

- Greater Eigenvector points to principal gradient direction.

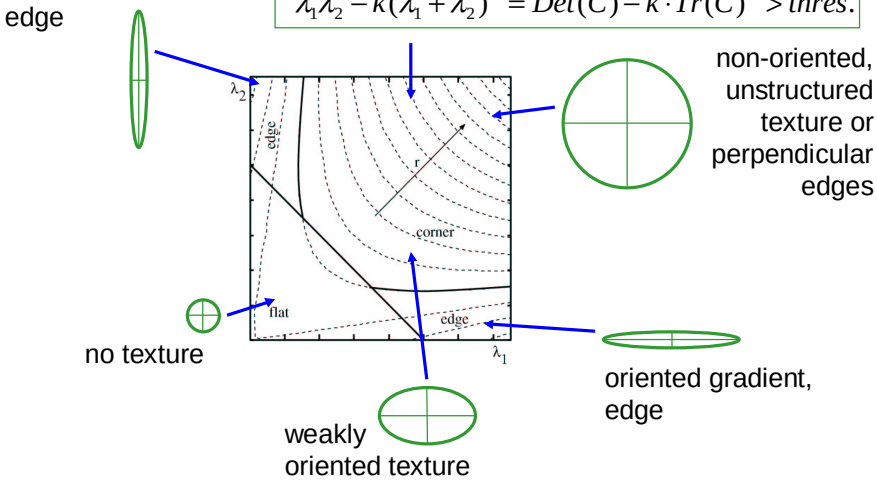
12

Dirk Farin, University of Mannheim

Texture Classification using Eigenvalues

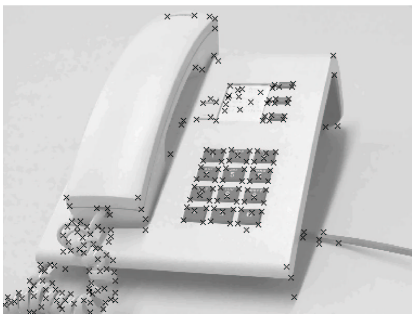
corner-texture response-function:

$$\lambda_1\lambda_2 - k(\lambda_1+\lambda_2)^2 = Det(C) - k \cdot Tr(C)^2 > thres.$$



Detected Corner Features: Results

- Extract local maxima of corner response function:



Compute Corner-Feature Correspondences

- Cross-correlate small windows around features,
- sort feature-pairs according to decreasing correlation,
- establish correspondences if both features are not assigned yet. („Highest Confidence First“ - principle)

Result:
sparse frame-to-frame motion-vector field

Reliable motion-vectors!

15

Dirk Farin, University of Mannheim

Parametric Motion-Estimation

- Each correspondence gives a set of data (x,y, x‘,y‘).
- Stack the equations obtained from all the data to get an overdetermined equation system

$$x' = \frac{a_{11}x + a_{12}y + t_x}{p_x x + p_y y + 1}, \qquad y' = \frac{a_{21}x + a_{22}y + t_y}{p_x x + p_y y + 1}.$$

$$\begin{pmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -x_1\hat{x}_1 & -y_1\hat{x}_1 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -x_1\hat{y}_1 & -y_1\hat{y}_1 \\ x_2 & y_2 & 1 & 0 & 0 & 0 & -x_2\hat{x}_2 & -y_2\hat{x}_2 \\ 0 & 0 & 0 & x_2 & y_2 & 1 & -x_2\hat{y}_2 & -y_2\hat{y}_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n & y_n & 1 & 0 & 0 & 0 & -x_n\hat{x}_n & -y_n\hat{x}_n \\ 0 & 0 & 0 & x_n & y_n & 1 & -x_n\hat{y}_n & -y_n\hat{y}_n \end{pmatrix} \begin{pmatrix} a_{11} \\ a_{12} \\ t_x \\ a_{21} \\ a_{22} \\ t_y \\ p_x \\ p_y \end{pmatrix} = \begin{pmatrix} \hat{x}_1 \\ \hat{y}_1 \\ \hat{x}_2 \\ \hat{y}_2 \\ \vdots \\ \hat{x}_n \\ \hat{y}_n \end{pmatrix}$$

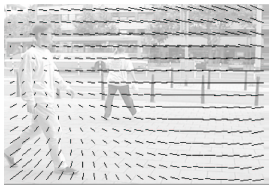
- Solve in the least-squares sense using, e.g., SVD.

16

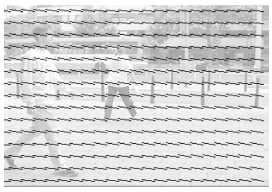
Dirk Farin, University of Mannheim

Parametric Motion Estimation

- Least-squares fitting of motion-model to vectors does not yield good results:

- Separation of background-motion vectors and foreground-object motion is required:

17

Dirk Farin, University of Mannheim

Robust Background-Motion Estimation 1

- Assume that background-motion is the **dominant** motion.
- Use robust regression algorithm (RANSAC, LMedS, ...)
  - ➔ robustness against outliers (here: foreground motion)
- we used

  Least Trimmed Squares (LTS).
- LTS minimizes sum of squared distances, but only considers the best-fitting fraction of data.

use for LS solution

ignore

$$\begin{pmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -x_1\hat{x}_1 & -y_1\hat{x}_1 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -x_1\hat{y}_1 & -y_1\hat{y}_1 \\ x_2 & y_2 & 1 & 0 & 0 & 0 & -x_2\hat{x}_2 & -y_2\hat{x}_2 \\ 0 & 0 & 0 & x_2 & y_2 & 1 & -x_2\hat{y}_2 & -y_2\hat{y}_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n & y_n & 1 & 0 & 0 & 0 & -x_n\hat{x}_n & -y_n\hat{x}_n \\ 0 & 0 & 0 & x_n & y_n & 1 & -x_n\hat{y}_n & -y_n\hat{y}_n \end{pmatrix} \begin{pmatrix} a_{11} \\ a_{12} \\ t_x \\ a_{21} \\ a_{22} \\ t_y \\ p_x \\ p_y \end{pmatrix} = \begin{pmatrix} \hat{x}_1 \\ \hat{y}_1 \\ \hat{x}_2 \\ \hat{y}_2 \\ \vdots \\ \hat{x}_n \\ \hat{y}_n \end{pmatrix}$$

increasing model error

18

Dirk Farin, University of Mannheim

Robust Background-Motion Estimation 2

Repeat several times:

- randomly select four correspondences to initialize model,
- calculate all model residuals, sort them,
- refine model using LS over best-fitting data.

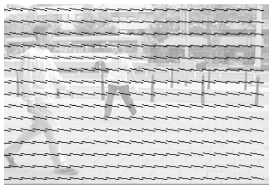
Choose best model.

background motion

foreground motion

$$\begin{pmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -x_1\hat{x}_1 & -y_1\hat{x}_1 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -x_1\hat{y}_1 & -y_1\hat{y}_1 \\ x_2 & y_2 & 1 & 0 & 0 & 0 & -x_2\hat{x}_2 & -y_2\hat{x}_2 \\ 0 & 0 & 0 & x_2 & y_2 & 1 & -x_2\hat{y}_2 & -y_2\hat{y}_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n & y_n & 1 & 0 & 0 & 0 & -x_n\hat{x}_n & -y_n\hat{x}_n \\ 0 & 0 & 0 & x_n & y_n & 1 & -x_n\hat{y}_n & -y_n\hat{y}_n \end{pmatrix} \begin{pmatrix} a_{11} \\ a_{12} \\ t_x \\ a_{21} \\ a_{22} \\ t_y \\ p_x \\ p_y \end{pmatrix} = \begin{pmatrix} \hat{x}_1 \\ \hat{y}_1 \\ \hat{x}_2 \\ \hat{y}_2 \\ \vdots \\ \hat{x}_n \\ \hat{y}_n \end{pmatrix}$$

result:



19

Dirk Farin, University of Mannheim

Presentation Outline

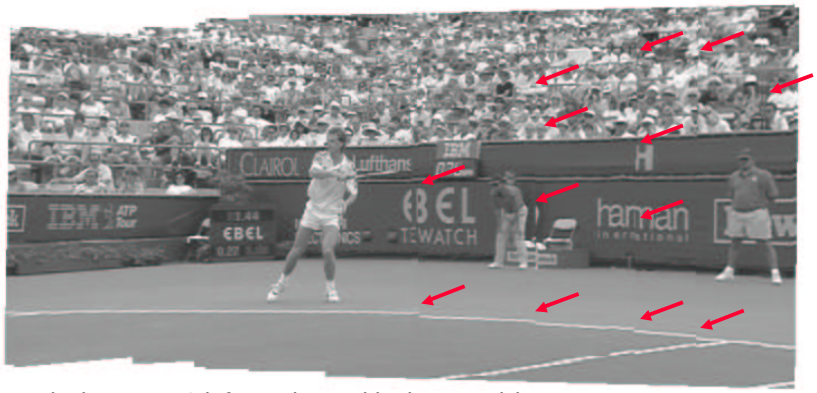
- automatic segmentation
  - camera motion model
  - corner feature extraction
  - feature-based motion estimation
  - dense motion estimation
- background mosaic reconstruction
- background subtraction
  - shape regularization with Markov Random Fields
- MPEG-4 coding application
- Object behaviour analysis
  - object silhouette description
  - matching to behaviour model

20

Dirk Farin, University of Mannheim

Long-term Prediction: why ?

- Motion-model is used to construct background-mosaic over long sequences.
- Errors accumulate to alignment errors.



(only every 10th frame is used in the mosaic)

21

Dirk Farin, University of Mannheim

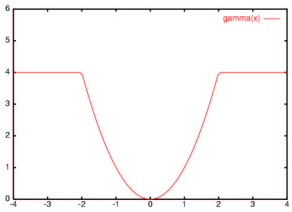
Long-term Prediction: Principle

- Dense registration of input frame to background mosaic.

$$\min_{\theta} E_{\theta} = \min_{\theta} \sum_{i=(x,y)} (I'(x',y') - I(x,y))^2$$

- Solve using Levenberg-Marquardt gradient descent.
- A robust error function can be used to improve background registration accuracy.
  - Residuals from foreground objects do not disturb estimated motion-model.

$$\gamma(e) = \begin{cases} e^2 & \text{for } |e| < t \\ t^2 & \text{else.} \end{cases}$$

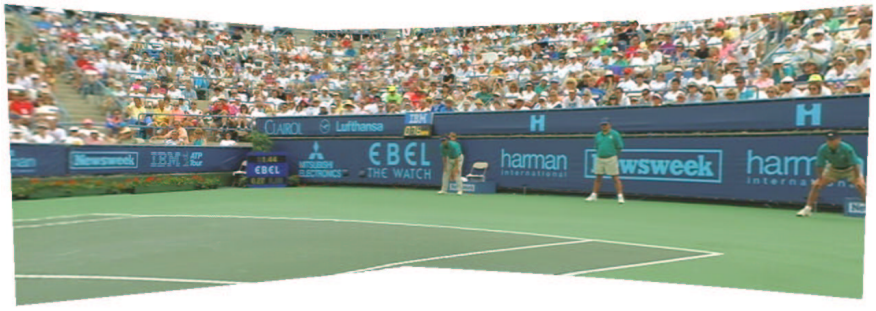


22

Dirk Farin, University of Mannheim

demo 1

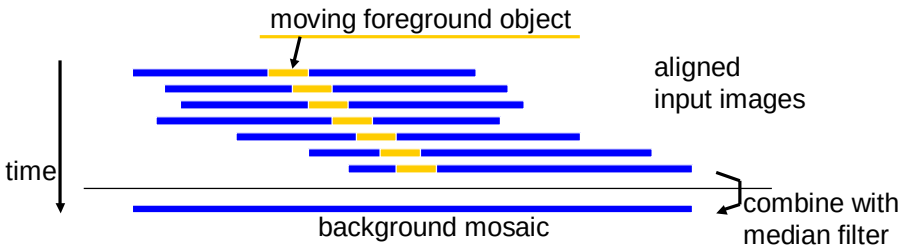
Long-term Prediction: Results



No alignment errors visible.

Background-Mosaic Reconstruction

- Combine all input images into a single panoramic background view.
- Apply a pixel-wise temporal median filter to remove moving foreground objects.



- Better algorithms for background reconstruction have been developed (submitted to ICIP-2003).

Background Subtraction

- Compute pixel-difference between background image and motion-compensated input image.
- A simple threshold on pixel differences produces too much pixel noise:

Better approach: use Markov-Random Field for regularization.

25

Dirk Farin, University of Mannheim

Bkg. Subtraction: Shape Regularization

- Model MRF using second-order Gibbs energies:

$$P(f) = Z^{-1} \cdot e^{-\frac{1}{T}U(f)} \qquad U(f) = \sum_p V_1(f_p) + \sum_p \sum_{p' \in N(p)} V_2(f_p, f_{p'})$$

normalization; annealing temperatur; single site energies; site-pair energies

- V_1: foreground/background decision

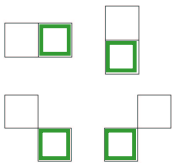
$$V_1(f_p) = \begin{cases} \beta \cdot e^{-d(p)^2} & \text{for } f_p = \text{foreground, and} \\ \beta \cdot (1.0 - e^{-d(p)^2}) & \text{for } f_p = \text{background.} \end{cases}$$

input/background difference

2	1	2
1	p	1
2	1	2

- V_2: shape regularization

$$V_2(f_p, f_{p'}) = \begin{cases} -\mu & \text{if } f_p = f_{p'} \\ \mu & \text{if } f_p \neq f_{p'} \end{cases}$$



26

Dirk Farin, University of Mannheim

Presentation Outline

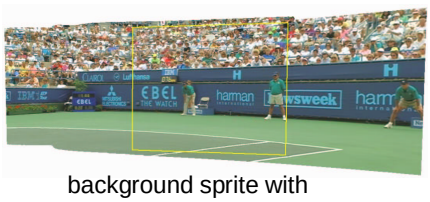
- automatic segmentation
  - camera motion model
  - corner feature extraction
  - feature-based motion estimation
  - dense motion estimation
- background mosaic reconstruction
- background subtraction
  - shape regularization with Markov Random Fields
- MPEG-4 coding application
- Object behaviour analysis
  - object silhouette description
  - matching to behaviour model

27

Dirk Farin, University of Mannheim

demo 2

MPEG-4 Sprite Coding: Reduced Bit-rate



background sprite with current view parameters

reconstructed view at MPEG decoder

MPEG-4 video object plane (VOP)

reconstructed video

28

Dirk Farin, University of Mannheim

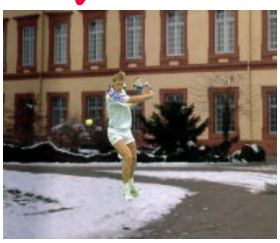
demo 3

MPEG-4: Background Replacement

different background image

reconstructed view at MPEG decoder

MPEG-4 video object plane (VOP)



reconstructed video

29

Dirk Farin, University of Mannheim

Presentation Outline

- automatic segmentation
  - camera motion model
  - corner feature extraction
  - feature-based motion estimation
  - dense motion estimation
- background mosaic reconstruction
- background subtraction
  - shape regularization with Markov Random Fields
- MPEG-4 coding application
- Object behaviour analysis
  - object silhouette description
  - matching to behaviour model

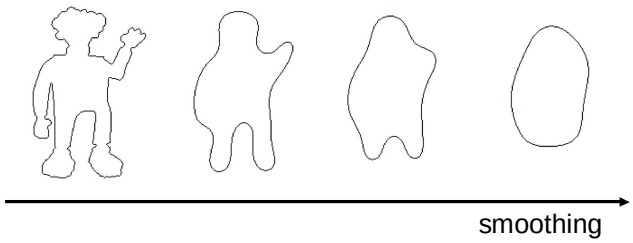
30

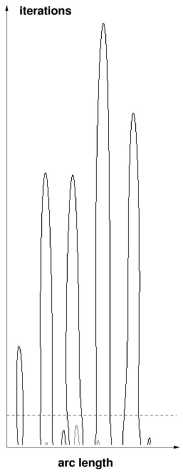
Dirk Farin, University of Mannheim

Object Shape Description

- Describe object silhouette with *Curvature Scale Space* (CSS) descriptors (as defined in MPEG-7).

- Mark inflection points over path length.
- Iteratively smooth boundary until no inflection points remain.
- Compare peaks in CSS diagram for shape comparion.





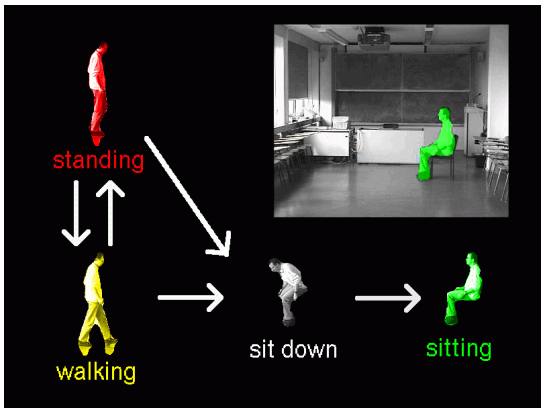
31

Dirk Farin, University of Mannheim

demo 4

Example Behavior Model

Typical object silhouettes and their temporal relationship:

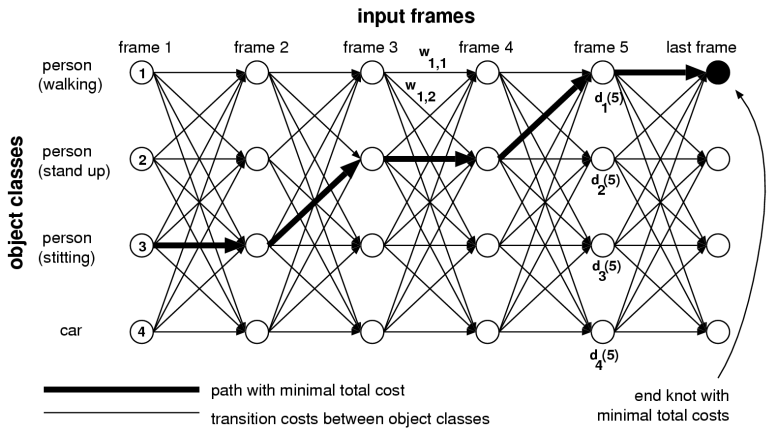


32

Dirk Farin, University of Mannheim

Behaviour Analysis from Video Sequence

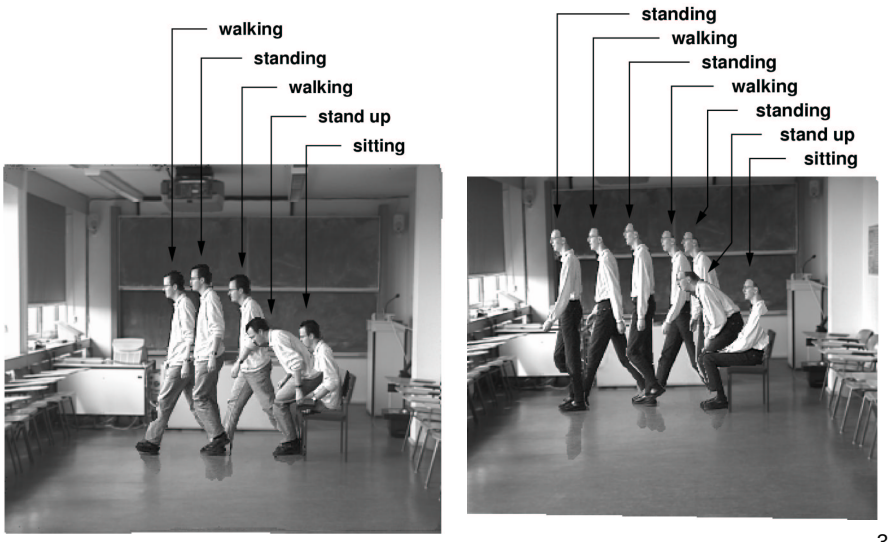
- Object silhouettes from automatic segmentation are matched using dynamic-programming approach



33

Dirk Farin, University of Mannheim

Example Results



34

Dirk Farin, University of Mannheim

Conclusions

- Automatic Video-Object Segmentation
  - pan/tilt/zoom camera model
  - two step motion estimation
    - feature-based short-term prediction
    - dense long-term prediction
  - segmentation based on background subtraction
- MPEG-4 sprite coding
- Object behaviour analysis
  - describe shapes by MPEG-7 CSS descriptors
  - match sequence of shape descriptors to behaviour model

35

Dirk Farin, University of Mannheim