Video-Object Segmentation: from MPEG-4 Coding to Behaviour Analysis

Dirk Farin

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

Introduction

Today:

Video-processing still at the signal level

- Video is considered as 2D signal over rectangular area
- No semantic analysis to differentiate between objects, ROI

Future:

Manipulation and compr. at object level

- Higher compression ratios
- New possibilities for interaction
- Image understanding

MPEG-4 Systems: Scene Composition

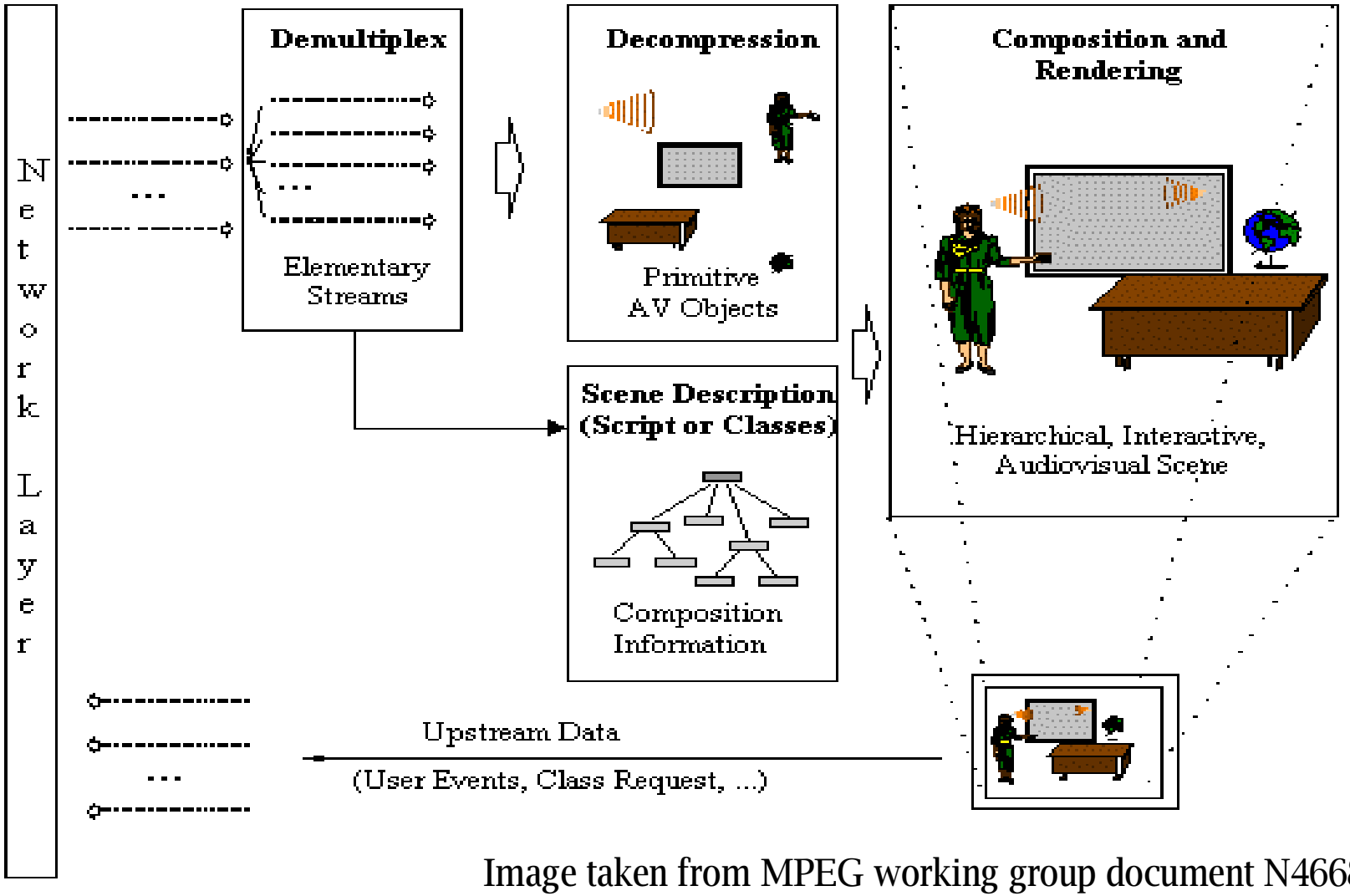
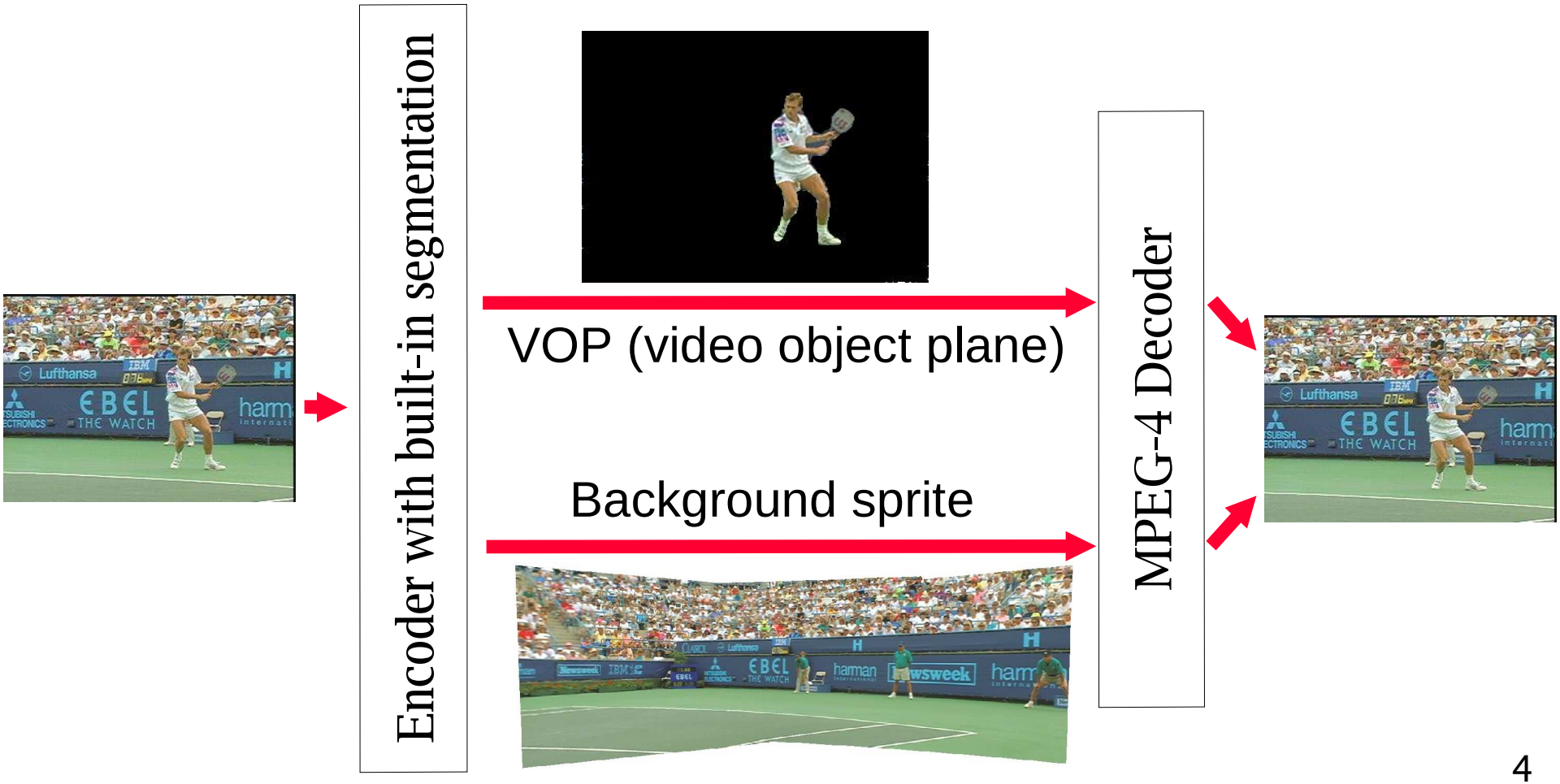


Image taken from MPEG working group document N4668

Dirk Farin, University of Mannheim

MPEG-4 Visual: Shaped-Object Coding

MPEG-4 video supports coding of shaped video-objects.
Compositing is done at system level.



Dirk Farin, University of Mannheim

demo 0

Increasing Compression Ratio 1/2

A non-changing background only has to be transmitted once.

Background view can be synthesized using the background sprite and a few camera parameters.

Large reduction of data-rate in

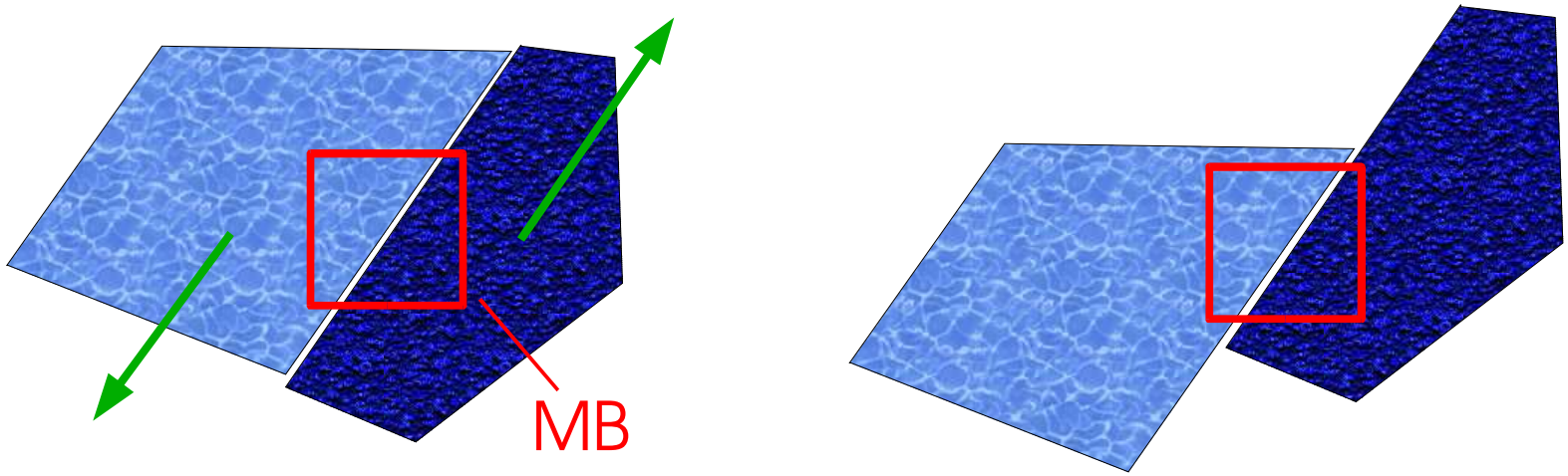
- Tele-conferencing applications
- News broadcasts
- Sports

Dirk Farin, University of Mannheim

Increasing Compression Ratio 2/2

In MPEG-2, motion compensation is applied to semantically meaningless macroblocks.

At object boundaries, object motion differs from background motion. Motion-compensation efficiency is reduced.



In MPEG-4, foreground and background parts will be predicted independently. Residual will be smaller.

Suggested topic for discussion: how is this problem solved in H.26L ?

Dirk Farin, University of Mannheim

Further Applications using Segmented Video

Automatic segmentation is a prerequisite for several applications:

- Scene compositing
  placing video objects into a virtual environment

- Object classification
  intelligent search in video databases

- Surveillance applications
  recognizing object behaviour (e.g., crime prevention)

Opposed to the usage for MPEG-4 compression, an accurate segmentation is required in these cases.

Central Problem: Automatic Segmentation

What is a video-object ?

- Shadows ?
- Small movements, wiggling trees in the background ?
- Object status change (parking cars)?
- Occlusions ?
- Reflections ?
- Multi-body objects (flocking birds) ?
- Hierarchical objects (driver in a car) ?
- It depends on the context !

Let us first consider a limited class of sequences:

- non-changing background
- pan/tilt/zoom – camera

Presentation Outline

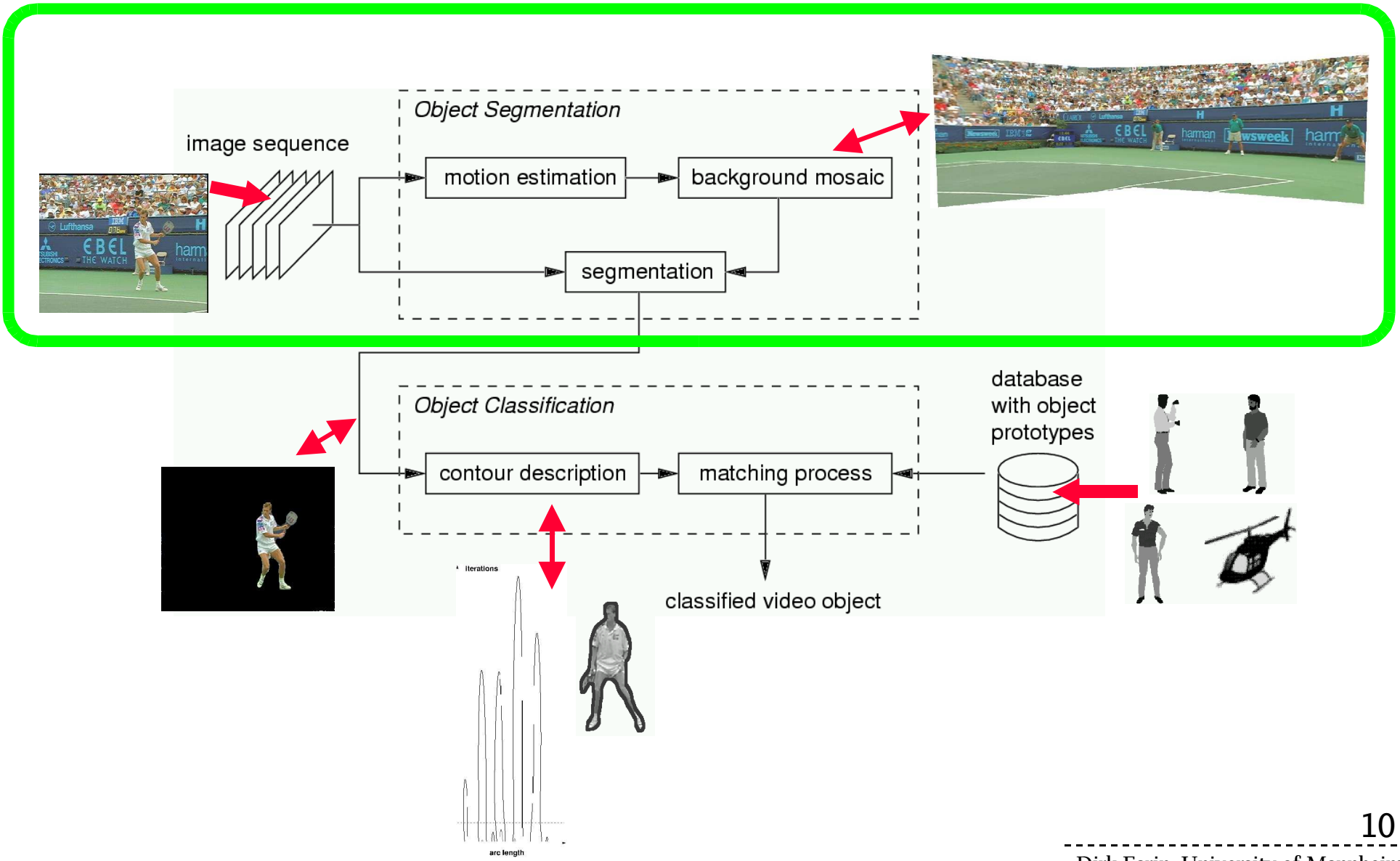
I. Automatic segmentation

- camera-motion estimation
- background reconstruction, subtraction

II. Integration of model knowledge

III. Object classification

Segmentation and Classification Overview



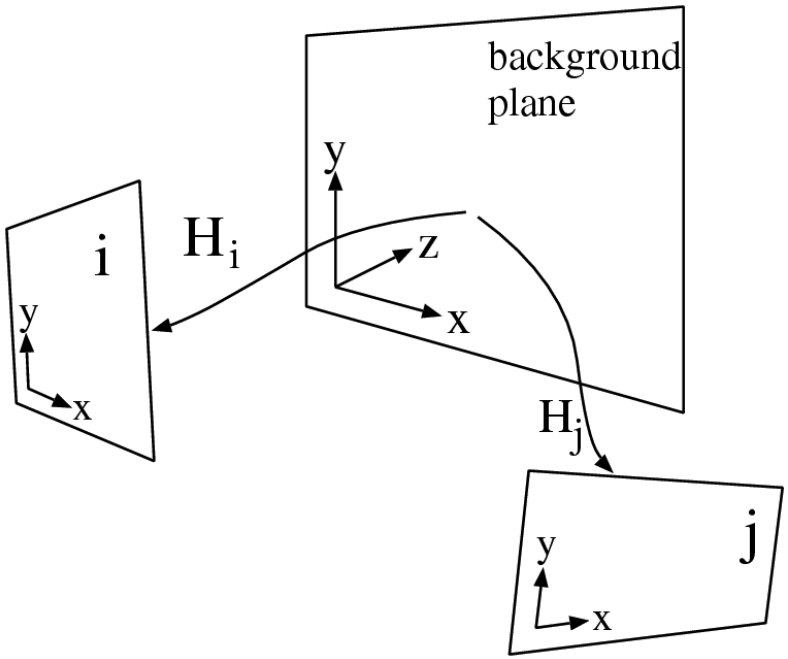
Camera-Motion Model

Assumption: video background is planar

Transformation from one projection to another:

$$x' = \frac{a_{11} x + a_{12} y + t_x}{p_x x + p_y y + 1}$$

$$y' = \frac{a_{21} x + a_{22} y + t_y}{p_x x + p_y y + 1}$$



Compatible motions:

a) planar background, arbitrary camera motion, or

b) rotating/zooming camera, arbitrary background depth.

Principle of Camera-Motion Estimation

First phase: **feature-based** motion estimation
short term prediction, motion between successive frames

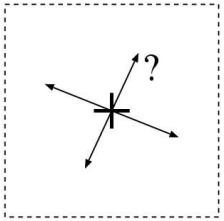
- Can handle large displacements
- robust estimation, insensitive to local minima
- fast approximate solution

Second phase: **dense** motion estimation
long term prediction, registration to background mosaic

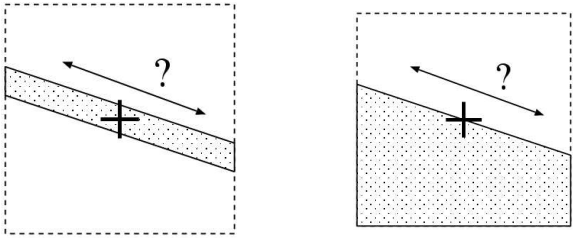
- prevents error accumulation
- locks to local minimum
- accurate estimation (sub-pixel accuracy)

Corner Localization for Reliable Motion Estim.

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

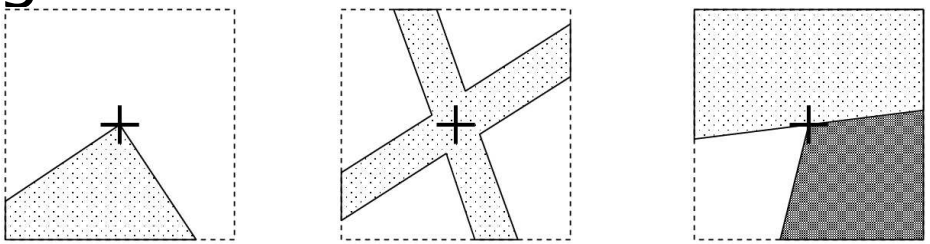


Along edges, only one motion-vector component is reliable (perpendicular to edge).



Only use motion-vectors located at „corners“.

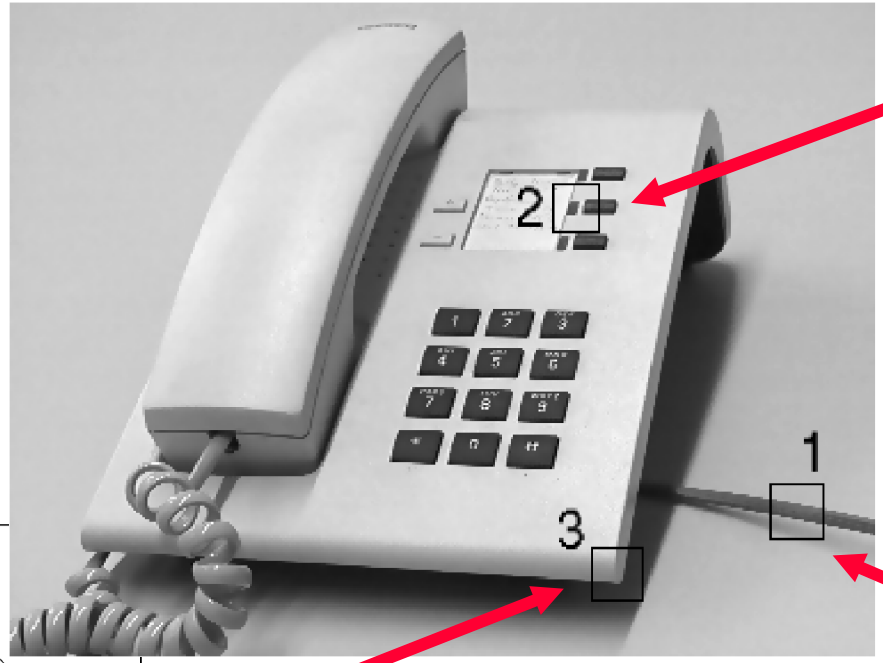
- strong texture variation in two directions

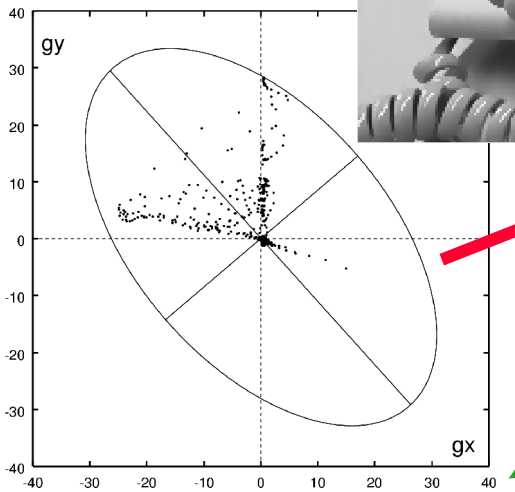


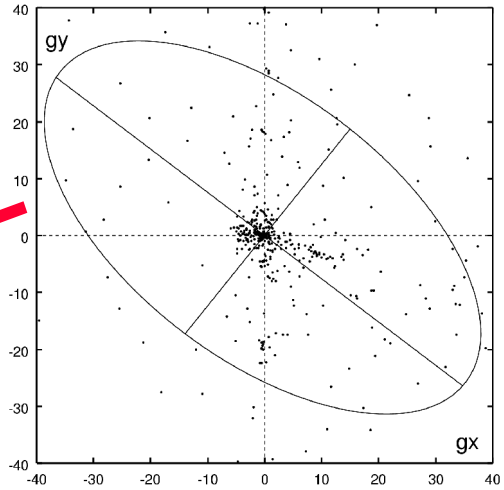
Gradient-Vector Distribution Examples

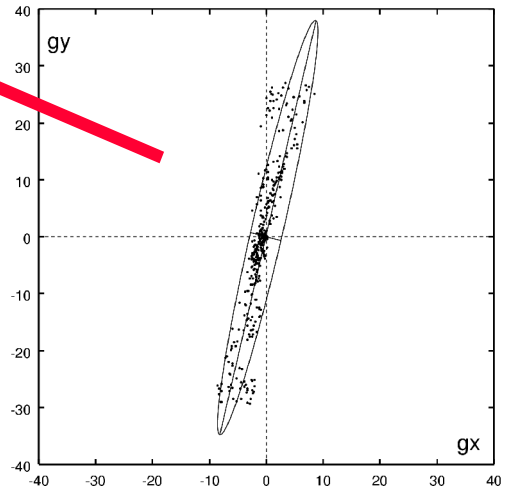
Harris corner detector

y-component
of gradient-vector





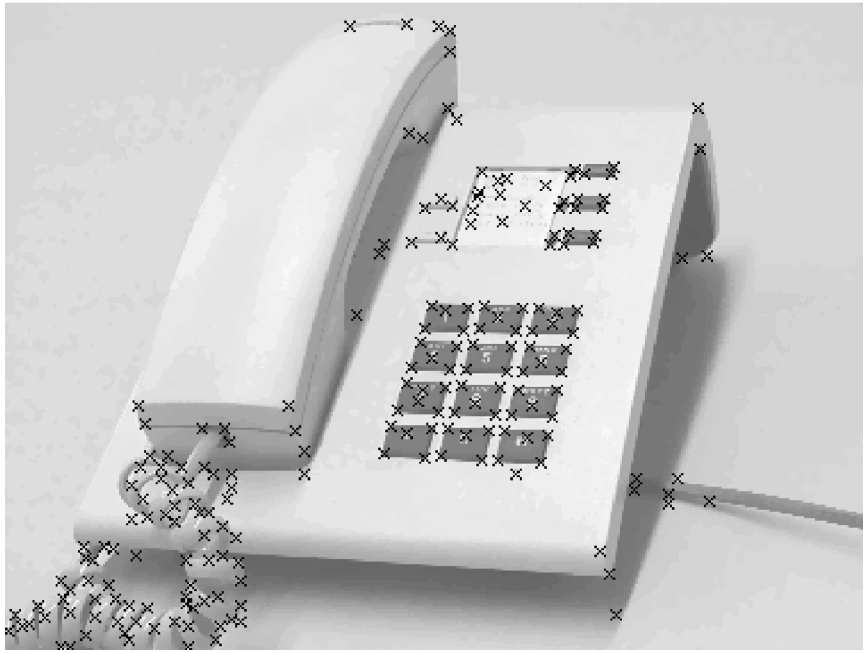




x-component
of gradient-vector

Detected Corner-Features: Results

Extract local maxima of corner response function:



Compute Corner-Feature Correspondences

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

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

Reliable motion-vectors!

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.

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:

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

Robust Background-Motion Estimation 2

1. Repeat several times:

2a. randomly select four correspondences to initialize model,

2b. calculate all model residuals, sort them,

2c. refine model using LS over best-fitting data.

3. Choose best model.

background motion

foreground motion

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

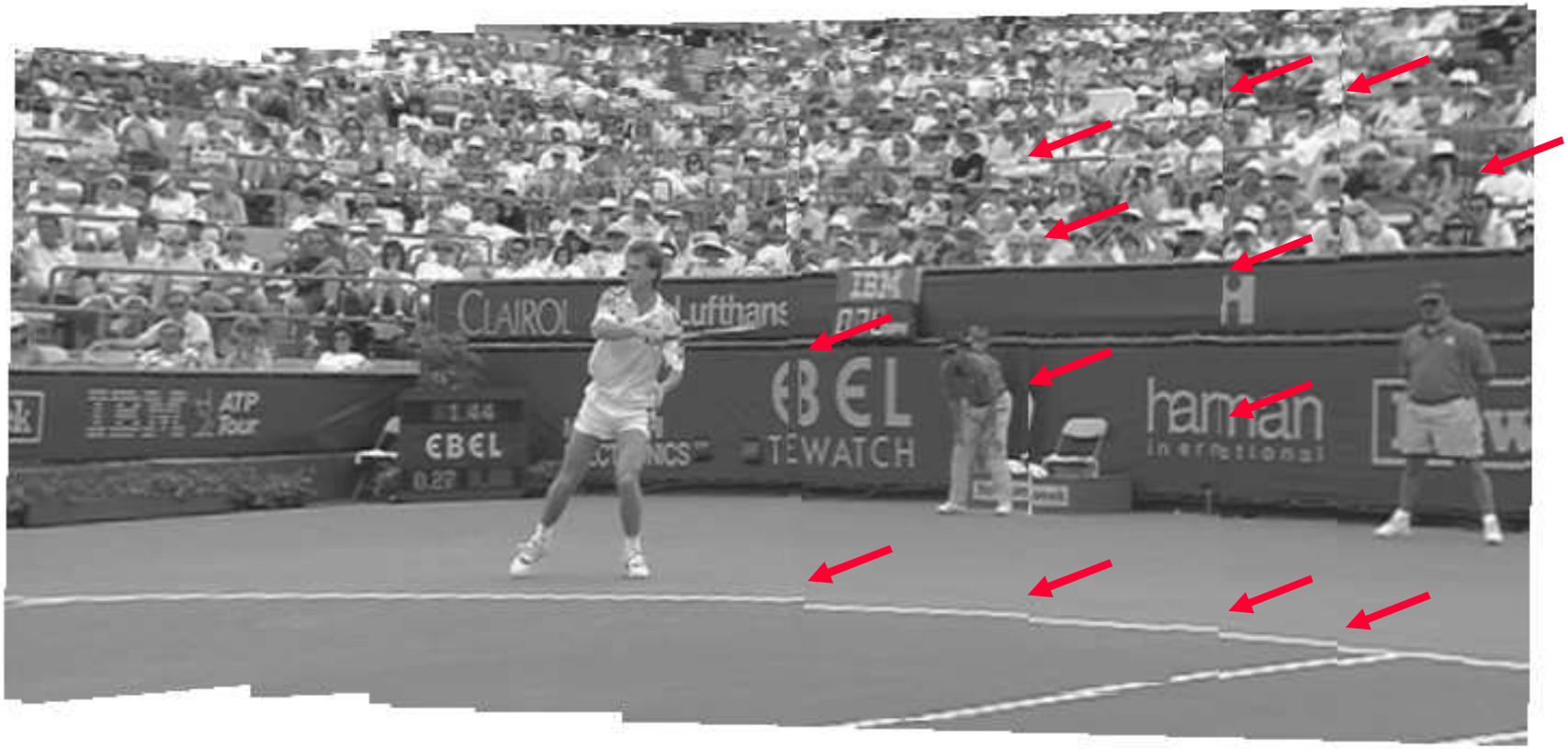
result:

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 for this mosaic)

Long-term Prediction: Principle

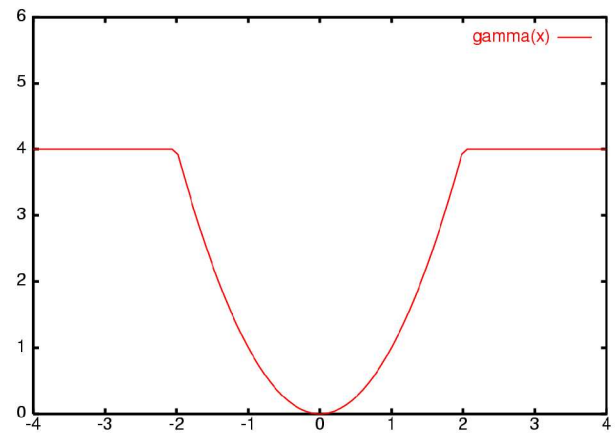
Dense registration of input frame to background mosaic.

$$min \sum_{i=(x,y)} \left(I(x,y) - I'(T(x),T(y)) \right)^2$$

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.



demo 1

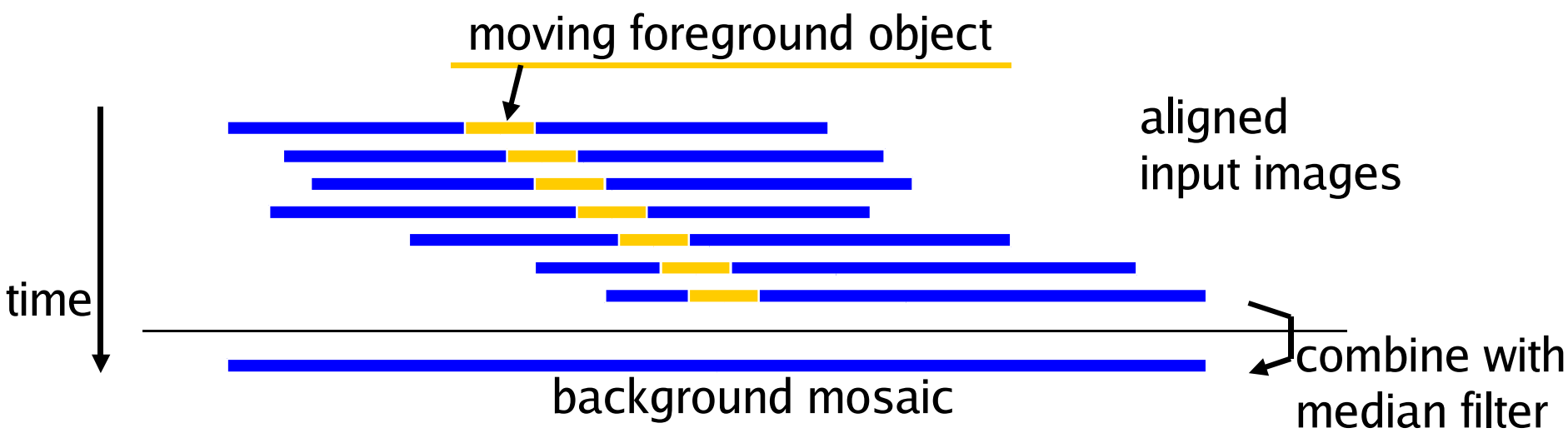
Long-term Prediction: Results

No alignment errors visible.

Dirk Farin, University of Mannheim

Background-Mosaic Reconstruction

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



Better algorithm for background reconstruction:

D. Farin et al.: Robust Background Estimation for Complex Video Sequences, IEEE International Conference on Image Processing, 2003 (to appear)

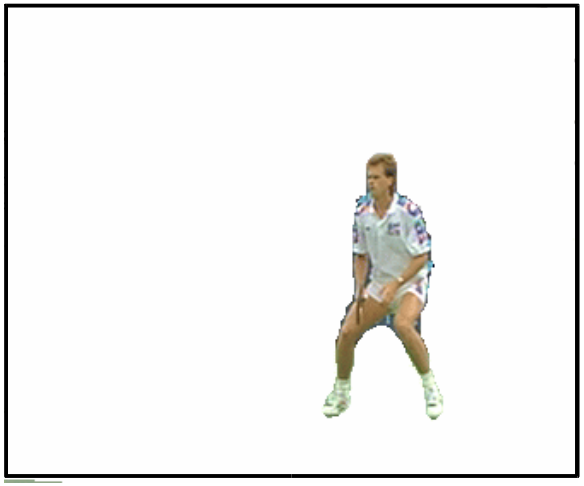
Dirk Farin, University of Mannheim

Background Subtraction

Compute pixel-difference between background image and motion-compensated input image.

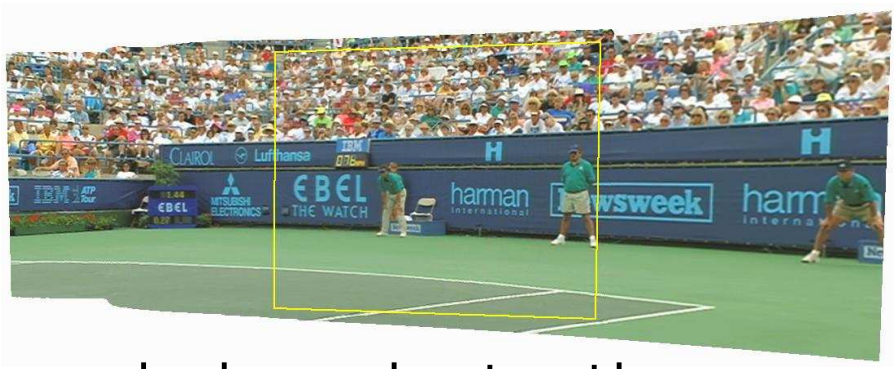
A simple threshold on pixel differences produces too much pixel noise:

Better approach: use Markov-Random Field for regularization.



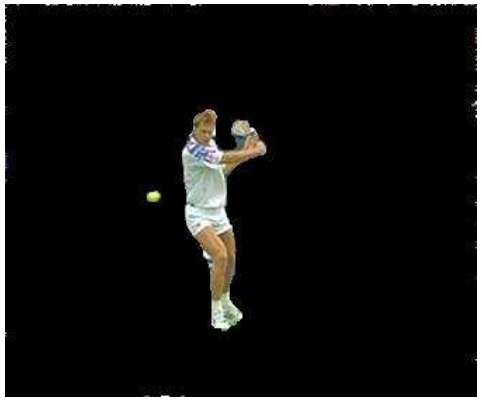
demo 2

MPEG-4 Sprite Coding: Reduced Bit-rate



background sprite with current view parameters

reconstructed view at MPEG decoder



MPEG-4 video object plane (VOP)

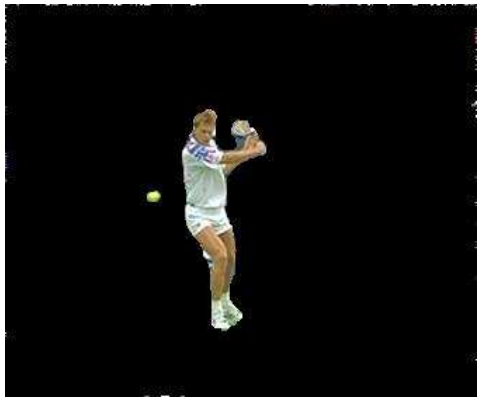
reconstructed video

demo 3

MPEG-4: Background Replacement

different background image

reconstructed view at MPEG decoder



MPEG-4 video object plane (VOP)

reconstructed video

Possible Problems with this Approach

If the foreground object looks like bkg.
or no complete bkg.-image is available,
object will have “holes”.

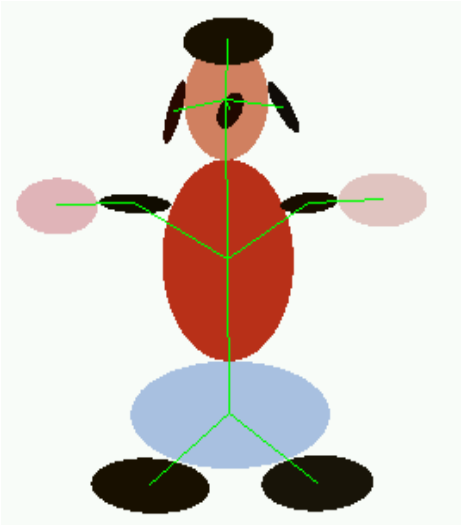
Same problem for other
algorithms like
color segmentation.

Our thesis: low-level video segmentation is prone to errors
that cannot be prevented without high-level knowledge.

Consequence:
going step by step to higher abstraction levels will not work.

Integrating Object-Models

Solution: **integrate object-models early !**



Build an abstract model of the object
to be segmented.

We used a graph description of the object.

- allows articulated object motion
- not too complicated object matching
  (if graph has no cycles)
- natural object description that is easy to generate

Dirk Farin, University of Mannheim

Generating Object-Models

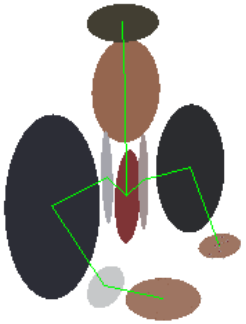
Since the object-model is a high-level description, it has to be defined manually.

1. Define main object regions

2. App. computes features

3. Define object skeleton

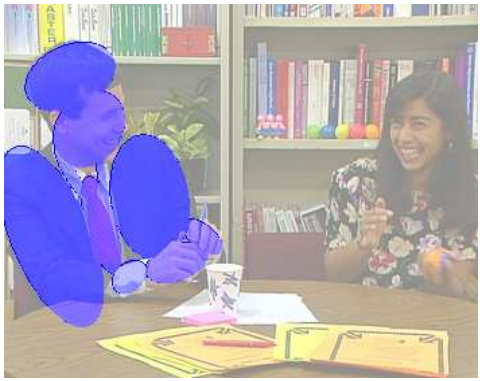
4. We get the final object model



Model Matching

1. Find a reduced set of probable locations for each model region (here: the tie).

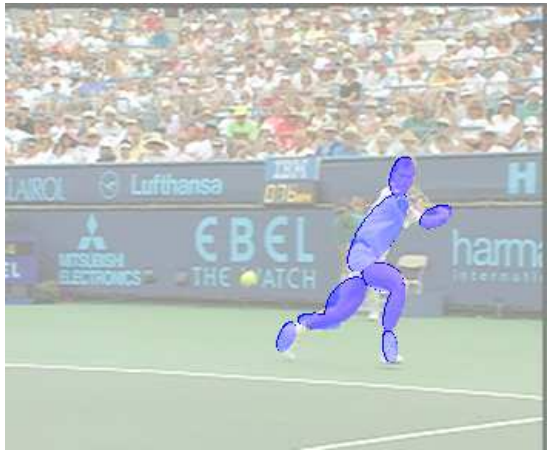
2. Use a dynamic programming algorithm to find the best-matching complete model.



3. Collect all color regions covered by the object model.

Object-Model for Specific Object Extraction

Object-model allows to specify the object of interest:



Matching the model of the tennis-player

Extracted player, but no ball !

Extraction of the ball only.

Extracting Object-Behaviour

What will be the next step after automatic segmentation ?

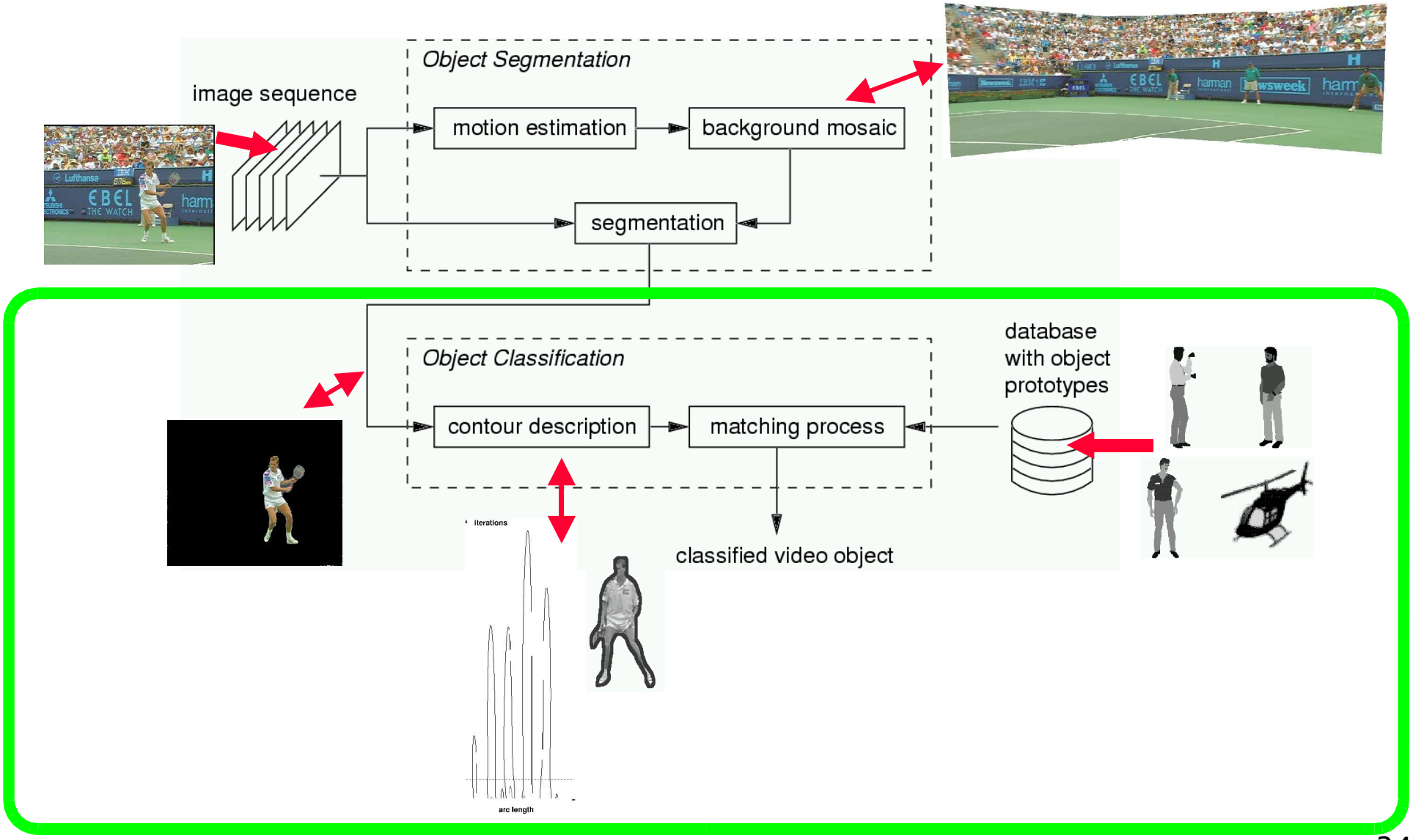
Analysis of Object-Behaviour

Not only use model for the visual appearance, but also for the object's change of *state* over time.

Parameters extracted from extracted model will allow to draw conclusions on the object-behaviour.

Dirk Farin, University of Mannheim

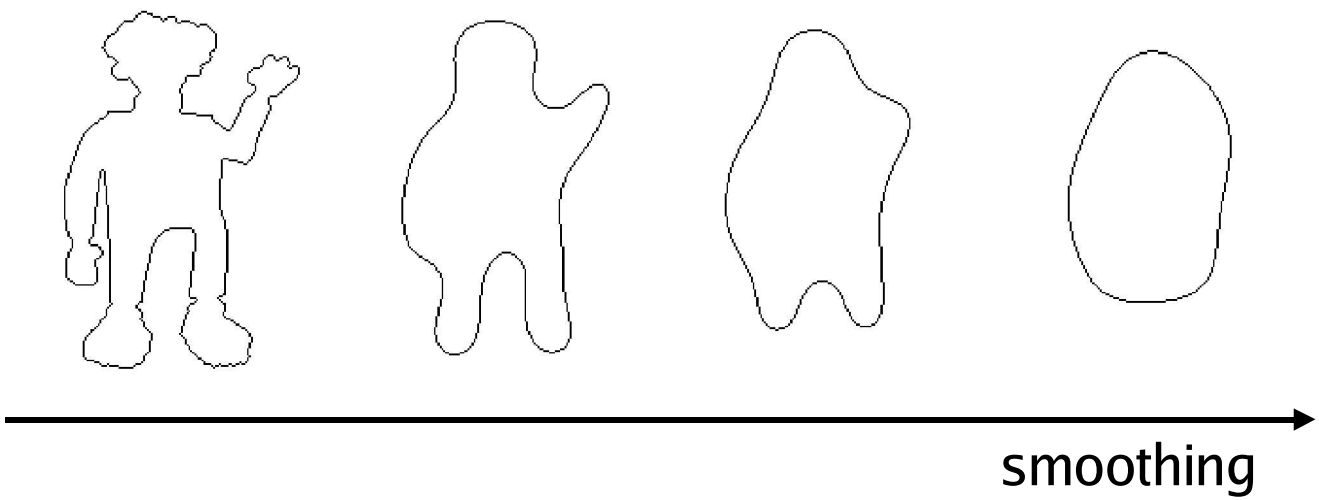
Segmentation and Classification Overview

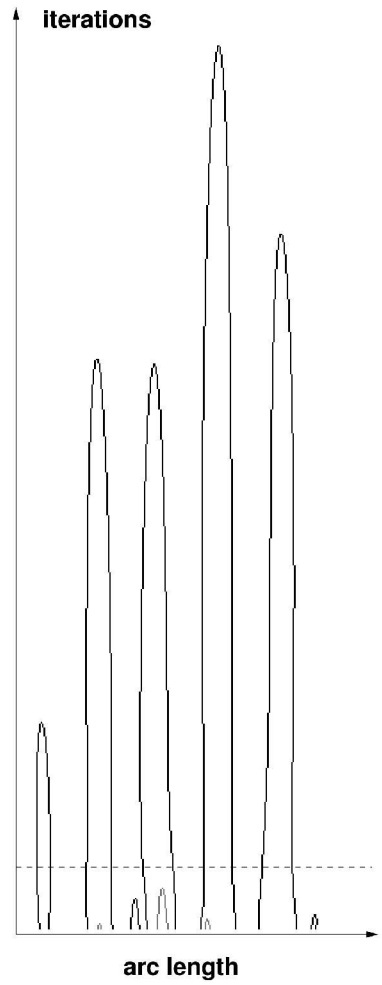


Object-Shape Description

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

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

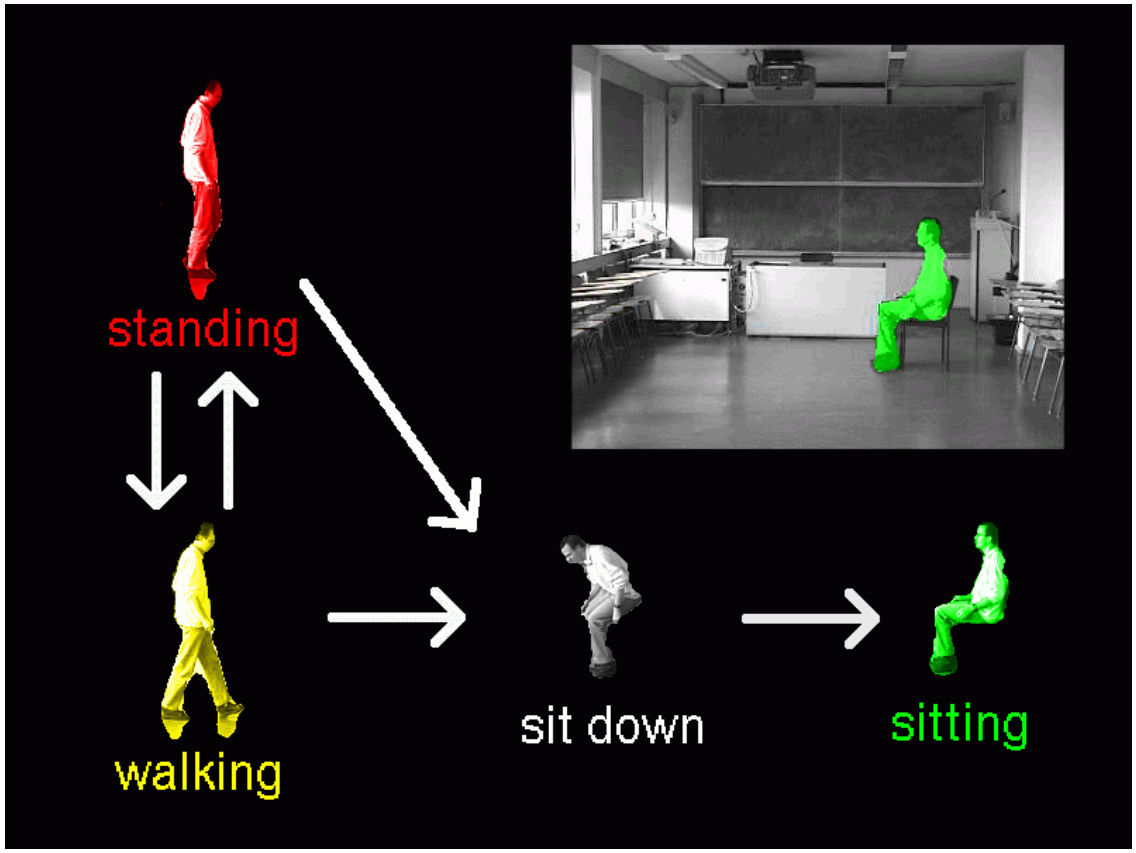




demo 4

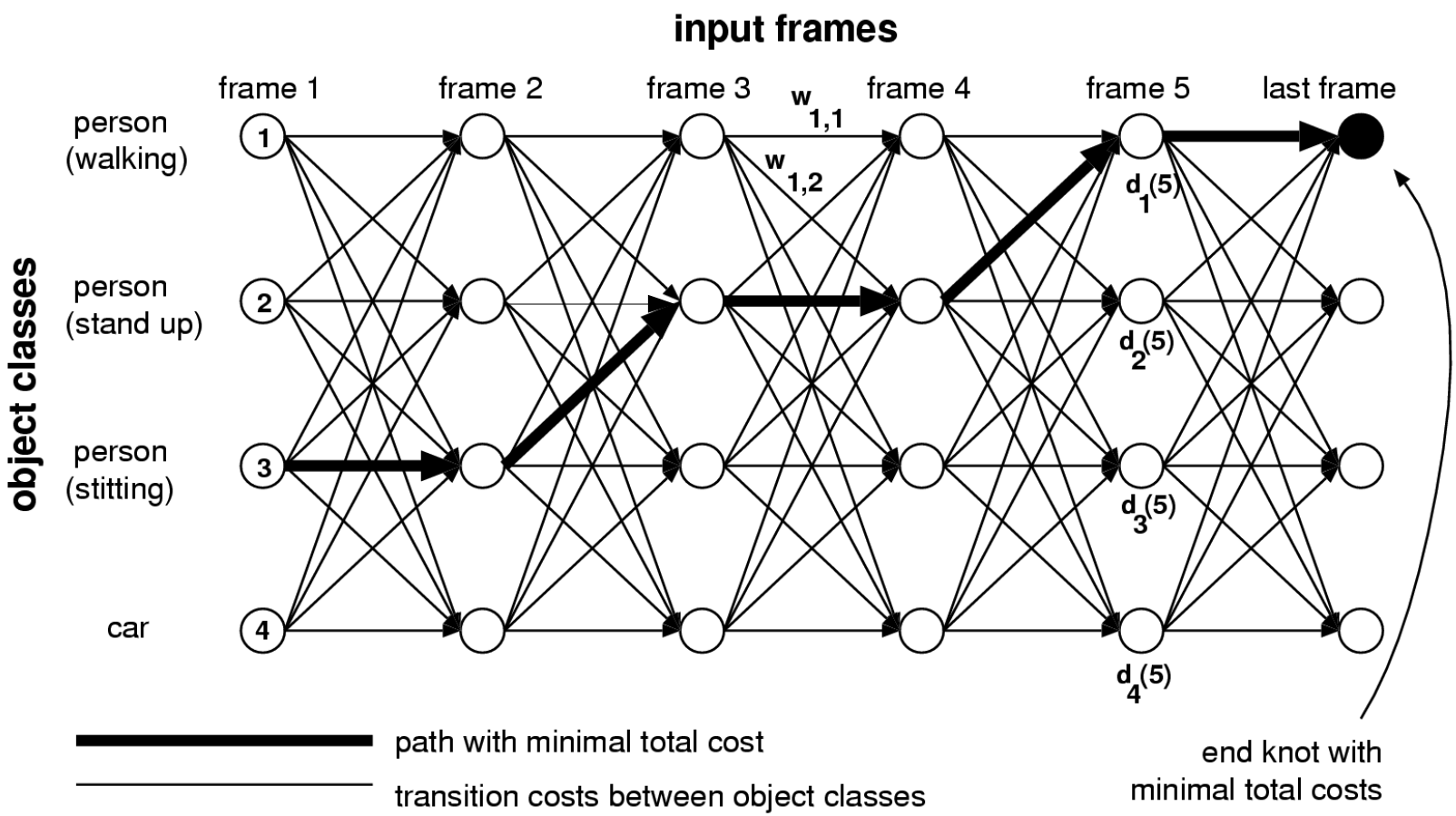
Example Behavior-Model

Typical object silhouettes and their temporal relationship:

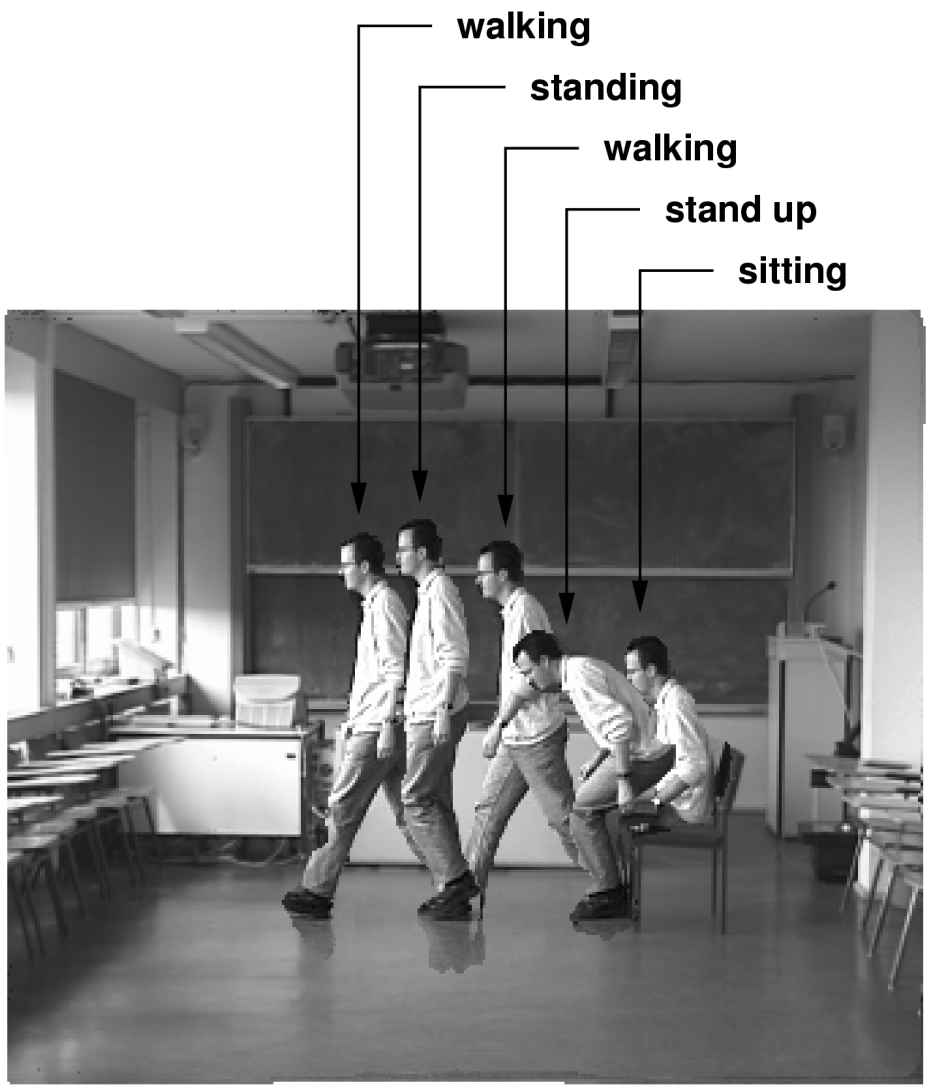


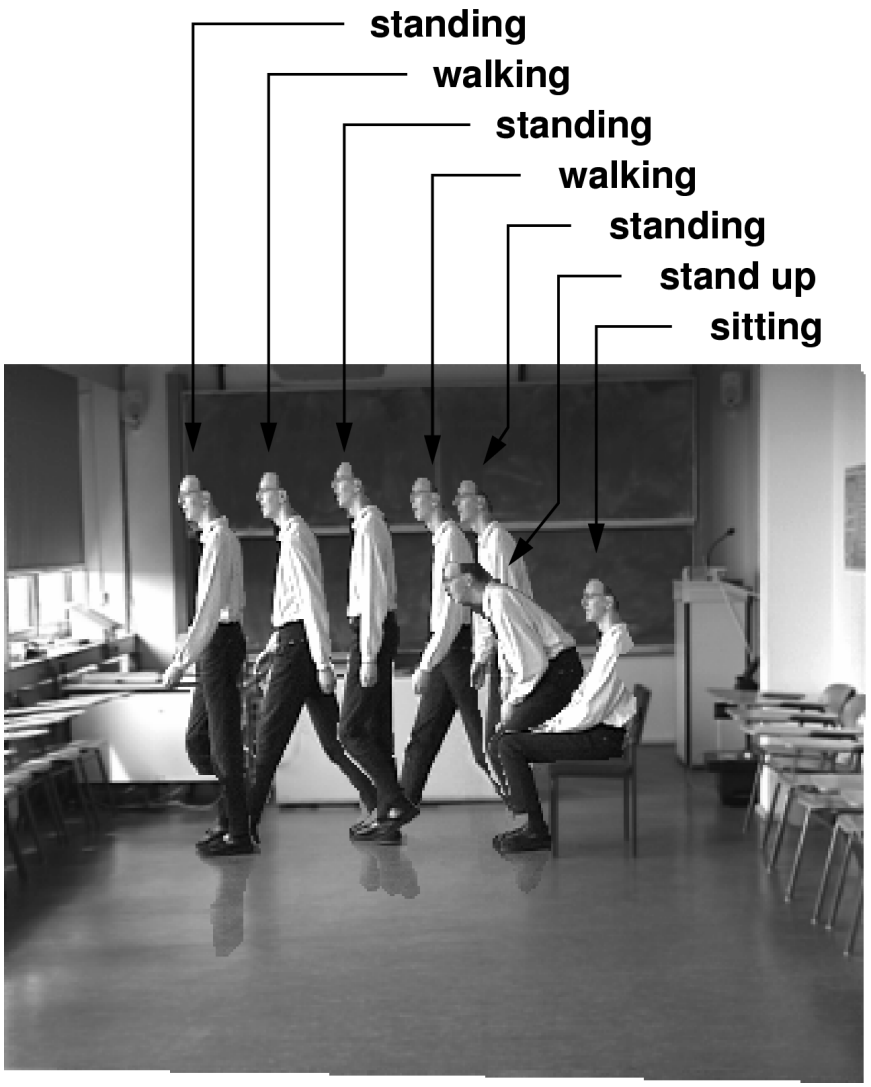
Behaviour Analysis from Video Sequence

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



Example Results





Conclusions

Automatic Video-Object Segmentation

- Global camera-motion compensation
- Background reconstruction
- Integration of object-models

Applications

- Increase of compression ratio (up to factor ~3)
- Video-scene compositing
- Object-behaviour extraction

Status

- Still active topic for research
- Practical solutions for limited application area

Further information

References:

D. Farin et al.: Segmentation and Classification of Moving Video Objects, in: Borko Furht, Oge Marques (eds), Handbook of Video Databases, CRC Press, 2003 (to appear)

D. Farin, P. H. N. de With, W. Effelsberg: A Segmentation System with Model assisted Completion of Video Objects, Visual Communications and Image Processing (VCIP), Lugano, Switzerland, July 2003 (to appear)

D. Farin, P. H. N. de With, W. Effelsberg: Robust Background Estimation for Complex Video Sequences, IEEE International Conference on Image Processing (ICIP), Barcelona, Spain 2003 (to appear)

Further references on

http://www.informatik.uni-mannheim.de/pi4