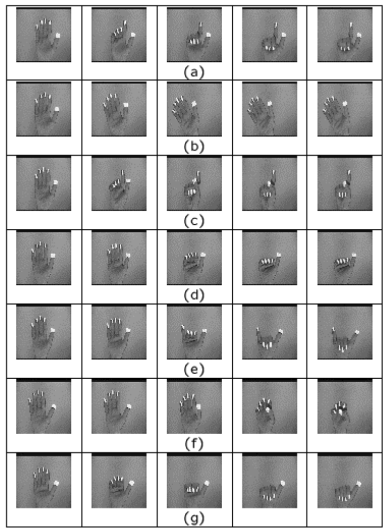
Hand Gesture Recognition, Aerobic exercises, Events

Lecture-15

Hand Gesture Recognition

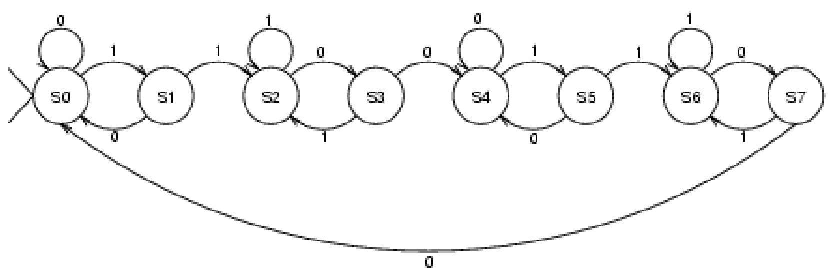
Seven Gestures



Gesture Phases

- Hand fixed in the start position.
- Fingers or hand move smoothly to gesture position.
- Hand fixed in gesture position.
- Fingers or hand return smoothly to start position.

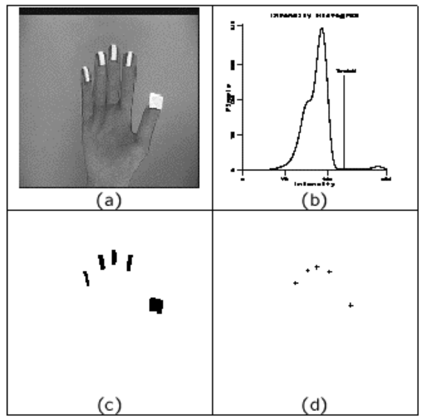
Finite State Machine



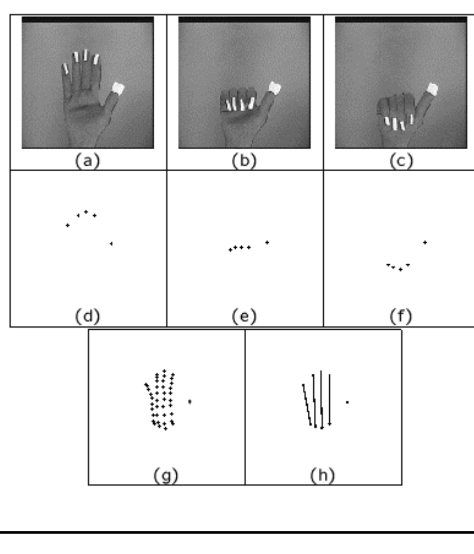
Main Steps

- Detect fingertips.
- Create fingertip trajectories using motion correspondence of fingertip points.
- Fit vectors and assign motion code to unknown gesture.
- Match

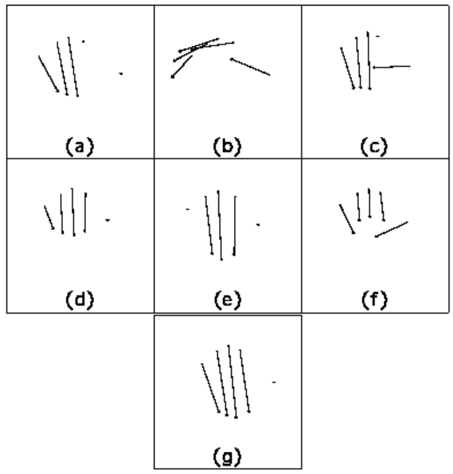
Detecting Fingertips



Vector Extraction



Vector Representation of Gestures



Results

Results

Run	Frames	L	R	U	D	T	G	S
1	200	√	√	√	√	√	√	√
2	250	√	√	√	√	√	√	√
3	250	√	√	√	X	√	√	√
4	250	√	√	√	√	√	√	√
5	300	√	√	√	√	√	√	√
6	300	√	√	√	√	√	√	√
7	300	√	√	√	√	√	√	√
8	300	√	√	√	√	√	√	√
9	300	√	√	√	√	*	*	*
10	300	√	√	√	√	√	√	√

L = Left, R = Right, U = Up, D = Down, T = Rotate, G = Grab, S = Stop, √ - Recognized, X - Not Recognized, * - Error in Sequence.

Action Recognition Using Temporal Templates

Jim Davis and Aaron Bobick

Main Points

- Compute a sequence of difference pictures from a sequence of images.
- Compute Motion Energy Images (MEI) and Motion History Images (MHI) from difference pictures.
- Compute Hu moments of MEI and MHI.
- Perform recognition using Hu moments.

MEI and MHI

Motion-Energy Images (MEI)

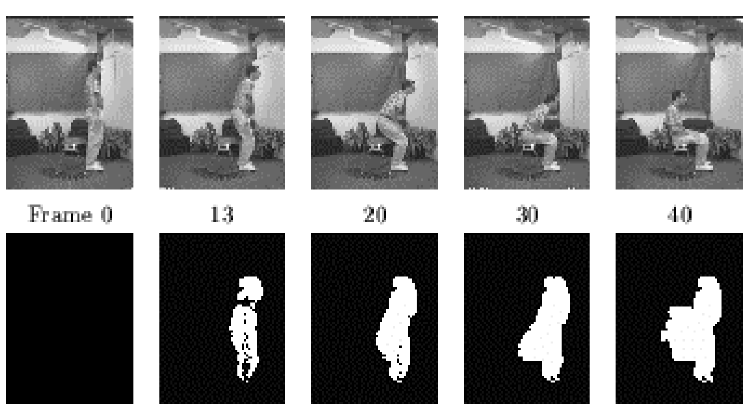
$$E_\tau(x,y,t) = \bigcup_{i=0}^{\tau-1} D(x,y,t-i)$$

Change Detected Images

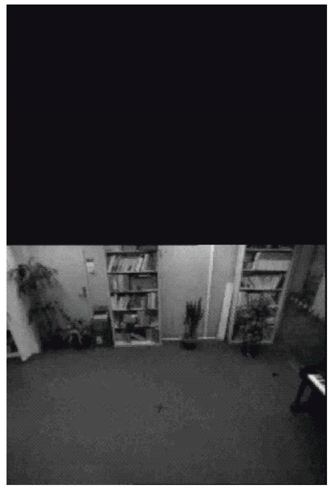
Motion History Images (MHI)

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

MEIs



Color MHI Demo



Summary

- Use seven Hu moments of MHI and MEI to recognize different exercises.
- Use seven views (-90 degrees to +90 degrees in increments of 30 degrees).
- For each exercise several samples are recorded using all seven views, and the mean and covariance matrices for the seven moments are computed as a model.
- During recognition, for an unknown exercise all seven moments are computed, and compared with all 18 exercises using Mahalanobis distance.
- The exercise with minimum distance is computed as the match.
- They present recognition results with one and two view sequences, as compared to seven view sequences used for model generation.

Moments

Binary image

General Moments

$$m_{pq} = \int\int x^p y^q \rho(x, y)dxdy$$

Central Moments (Translation Invariant)

$$\mu_{pq} = \int\int (x-\overline{x})^p (y-\overline{y})^q \rho(x,y)\; d(x-\overline{x})d(y-\overline{y})$$

$$\overline{x} = \frac{m_{10}}{m_{00}}, \overline{y} = \frac{m_{01}}{m_{00}}$$

centroid

Central Moments

$$\mu_{00} = m_{00} \equiv \mu$$
$$\mu_{01} = 0$$
$$\mu_{10} = 0$$
$$\mu_{20} = m_{20} - \mu\overline{x}^2$$
$$\mu_{11} = m_{11} - \mu\overline{x}\overline{y}$$
$$\mu_{02} = m_{02} - \mu\overline{y}^2$$
$$\mu_{30} = m_{30} - 3m_{20}\overline{x} + 2\mu\overline{x}^3$$
$$\mu_{21} = m_{21} - m_{20}\overline{y} - 2m_{11}\overline{x} + 2\mu\overline{x}^2 y$$
$$\mu_{12} = m_{12} - m_{02}\overline{x} - 2m_{11}\overline{y} + 2\mu\overline{x}y^2$$
$$\mu_{03} = m_{03} - 3m_{02}\overline{y} + 2\mu\overline{y}^3$$

Moments

Hu Moments: translation, scaling and rotation invariant

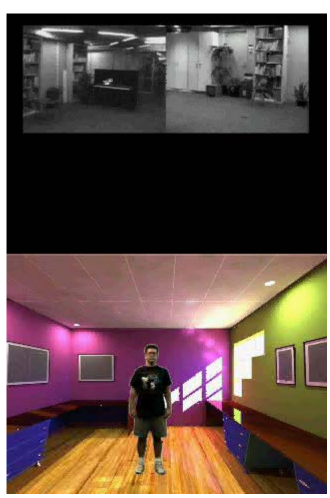
$$\upsilon_1 = \mu_{20} + \mu_{02}$$

$$\upsilon 2 = (\mu_{20} - \mu_{02})^2 + \mu_{11}{}^2$$

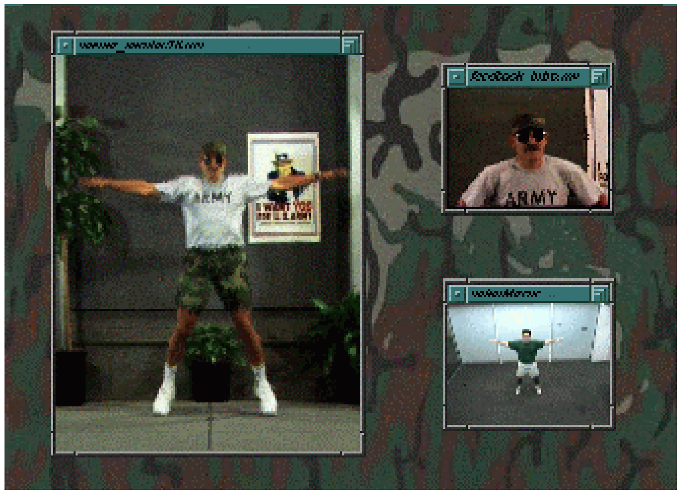
$$\upsilon_3 = (\mu_{30} - 3\mu_{12})^2 + (3\mu_{12} - \mu_{03})^2$$

$$\upsilon_4 = (\mu_{30} + \mu_{12})^2 + (\mu_{21} + \mu_{03})^2$$

$$\vdots$$



PAT (Personal Aerobic Trainer)



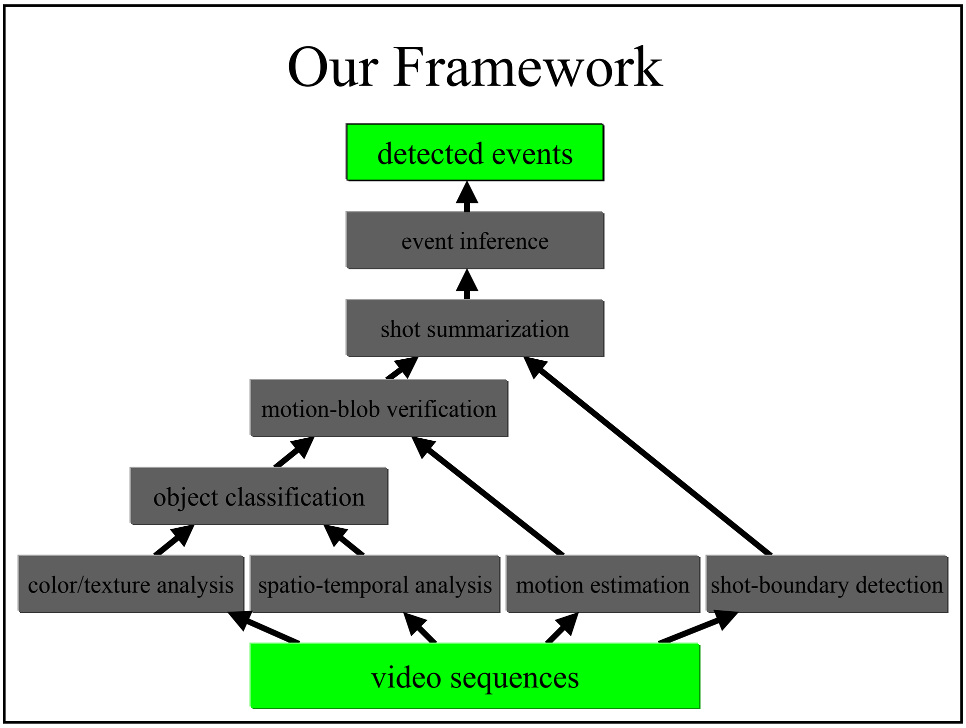
PAT (Personal Aerobic Trainer)

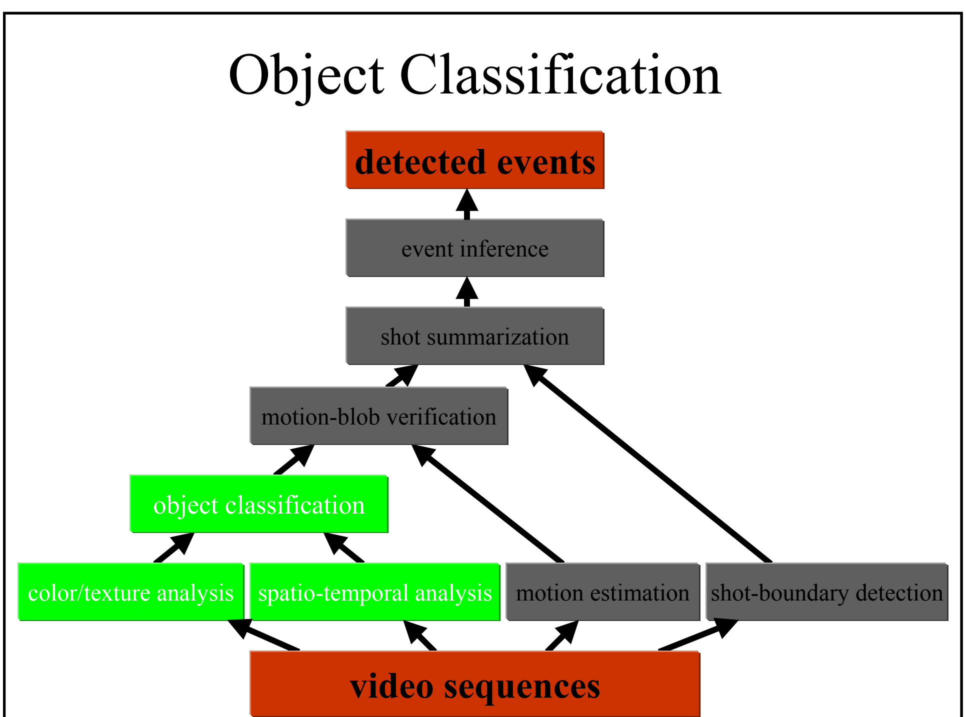
http://vismod.www.media.mit.edu/vismod/demos/actions/mhi_generation.mov

PAT (Personal Aerobic Trainer)

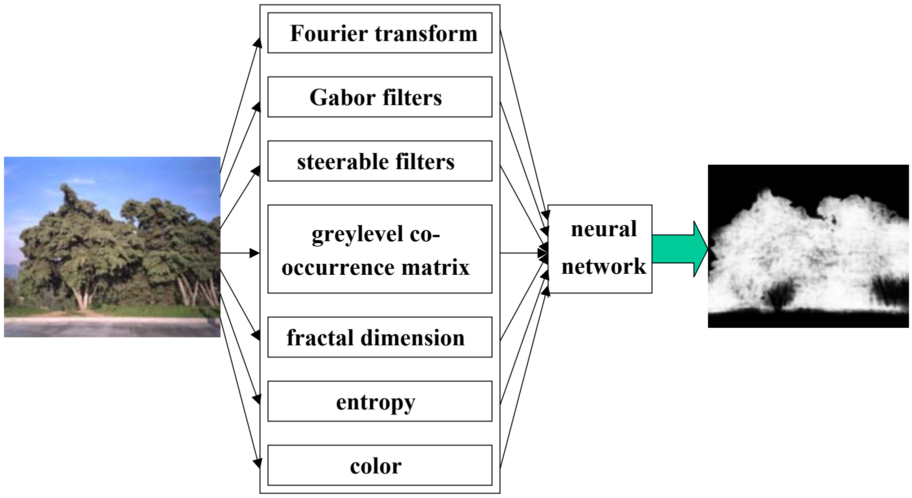
A Framework for the Design of Visual Event Detectors

Niels Haering


Our Framework
detected events
event inference
shot summarization
motion-blob verification
object classification
color/texture analysis
spatio-temporal analysis
motion estimation
shot-boundary detection
video sequences

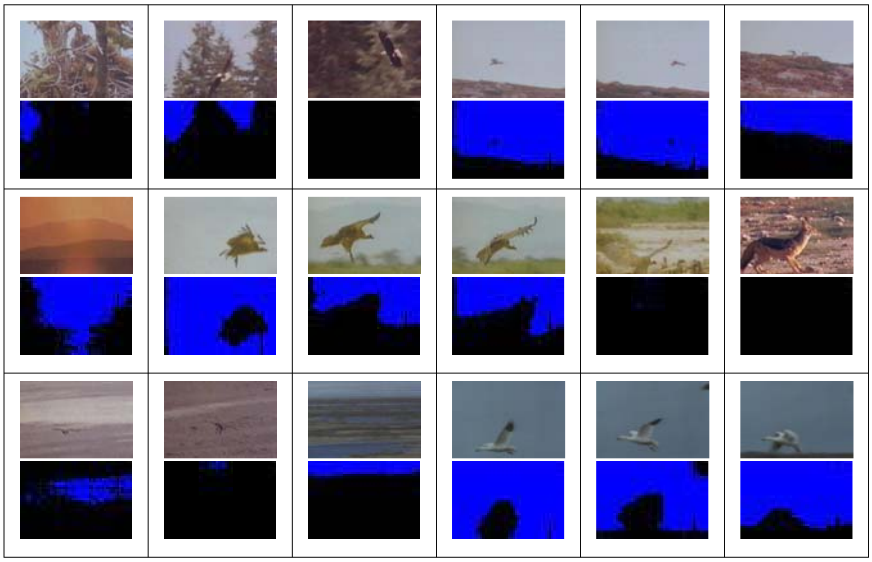

Object Classification
detected events
event inference
shot summarization
motion-blob verification
object classification
color/texture analysis
spatio-temporal analysis
motion estimation
shot-boundary detection
video sequences

Object Classification

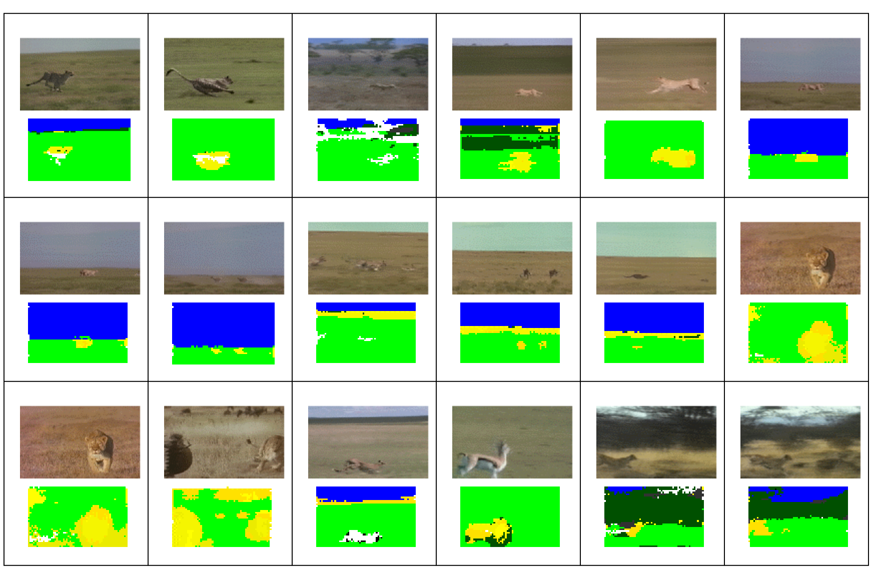


Deciduous Trees

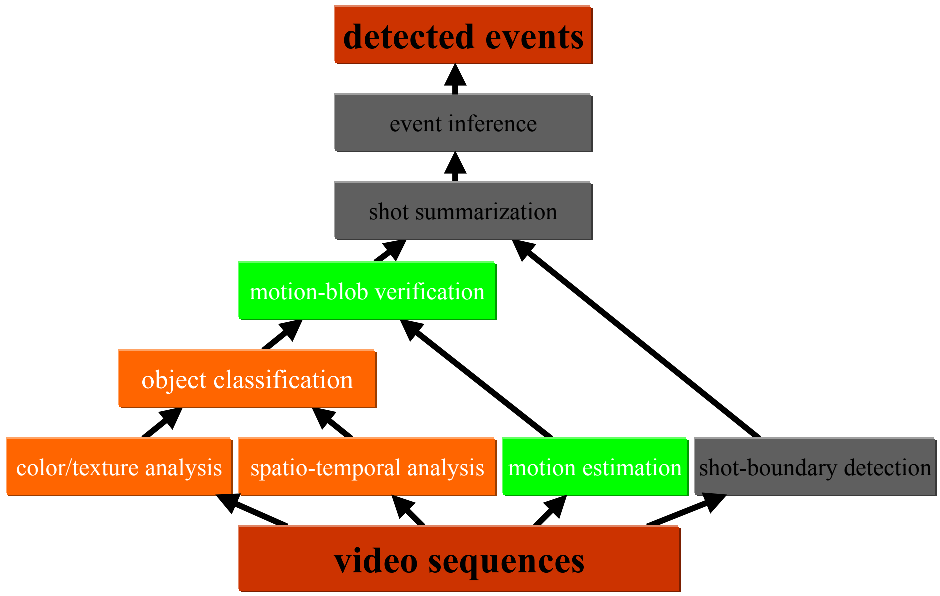
Sky



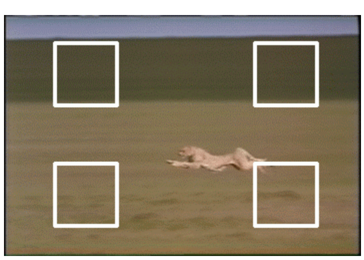
Animals, Sky, Grass, Trees, Rock

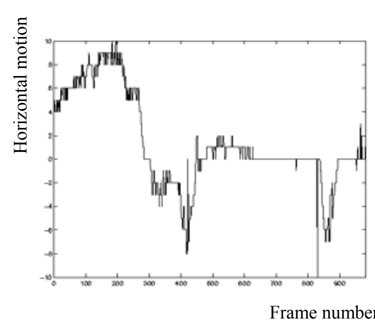


Motion-blob Verification



Motion Estimation

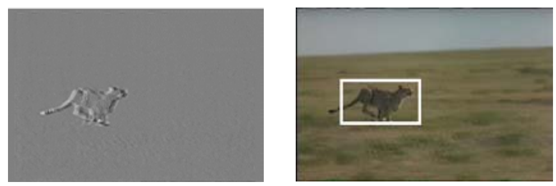




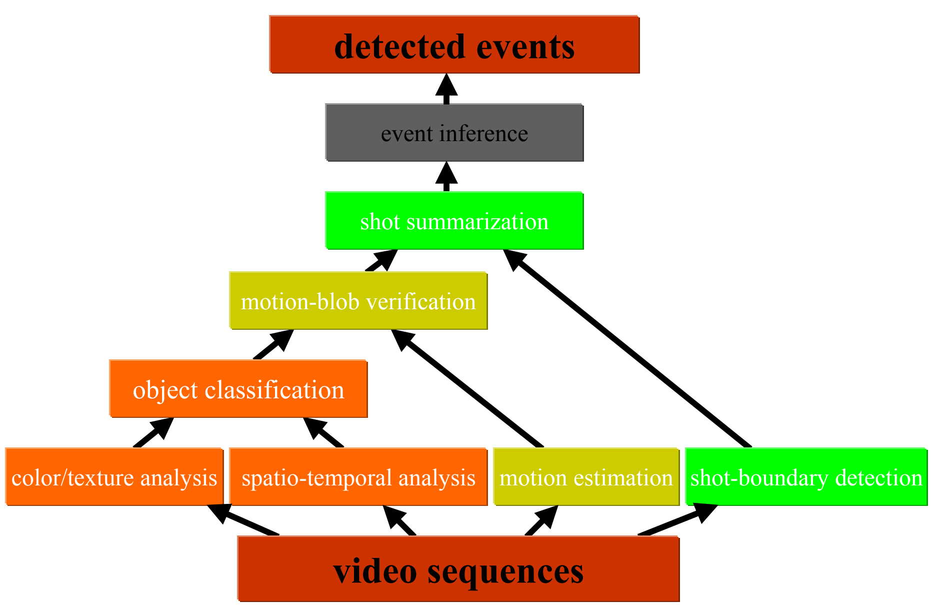
- three parameter system: x-, y-translation, and zoom,
- 4 motion estimates based on pyramid,
- 4 motion estimates based on previous best match,
- “texture” measure prevents ambiguous matches

Motion-blob detection

Motion estimate
Δx = -7
Δy = 0
zoom = 1.0



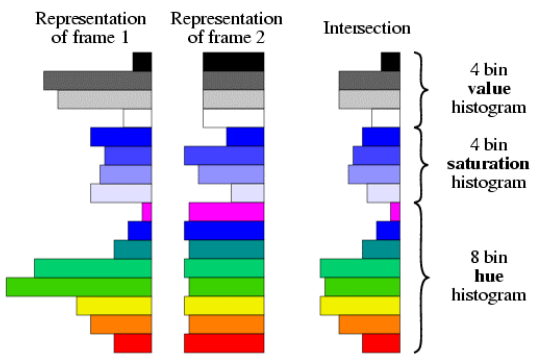
Shot Summarization



Shot Detection

Characteristics of shot boundaries:

- Change of camera/viewpoint
- Change of color characteristics



4 Bins for Value
4 Bins for Saturation
8 bins for hue

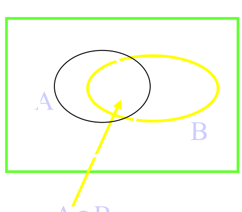
$$= 0.79$$

$$\cap = \frac{\sum_{n=0}^{15} \min(f_1(n), f_2(n))}{\min(\sum_{n=0}^{15}(f_1(n), \sum_{n=0}^{15} f_2(n))}$$

Shot Summaries

[redacted]

[redacted]



A = Ground Truth

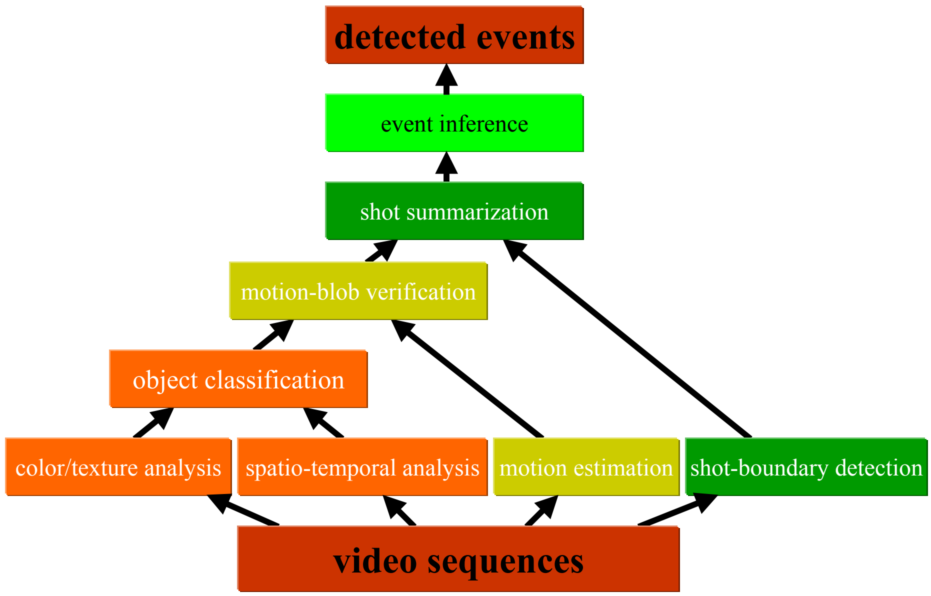
B = Result of Algorithm

A∩B = Correct detection

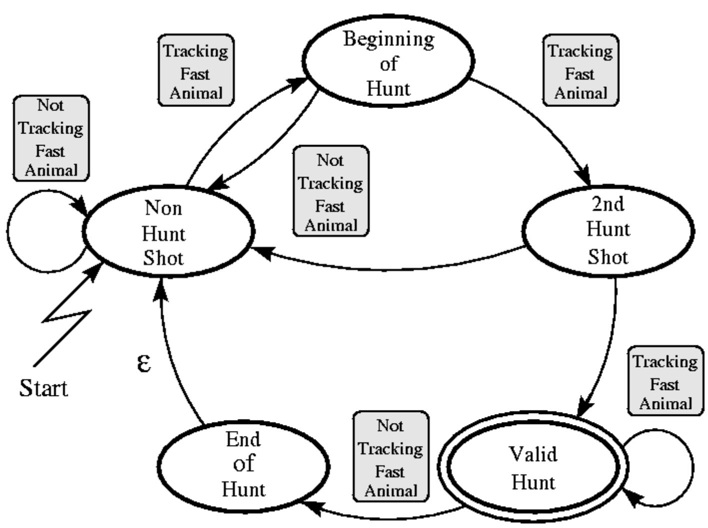
$$recall = \frac{A \cap B}{A}$$

$$precision = \frac{A \cap B}{B}$$

Event Inference



Hunt events



Hunts

Hunt

Non-hunt

Hunts

Non-
hunt

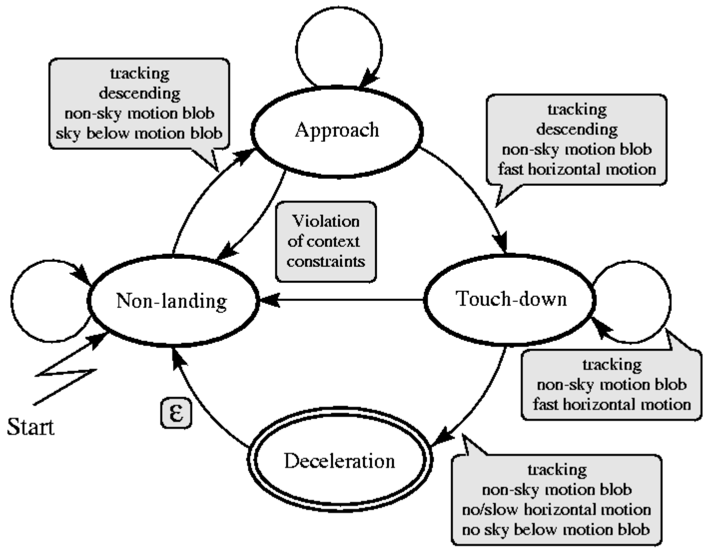
Hunt

Non-hunt

Event Detection

Landing Events

Landing Events



Landing Events

Non-landing

Approach

Touch-down

Deceleration

Non-landing

Landing Events

Non-landing

Approach

Touch-down

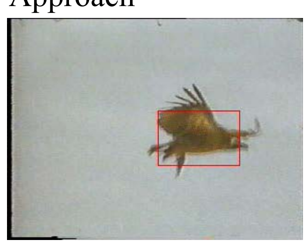
Deceleration

Non-landing

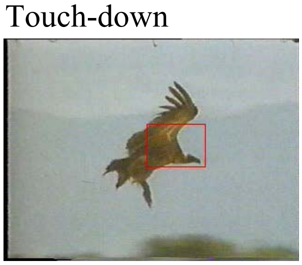
Landing Events

Non-landing

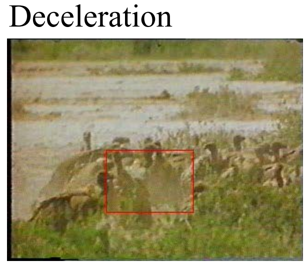
Approach



Touch-down



Deceleration



Non-landing

Conclusions

- Many natural objects are easily recognized by their color and texture signatures (shape is often not needed)
- Many events are easily detected and recognized by the classes of the comprising objects and their approximate motions
- The proposed visual event detection is robust to changes in scale, color, shape, occlusion, lighting conditions, view points and distances, and image compression