Basic Image Processing Algorithms

PPKE-ITK, 2016

Lecture 11.

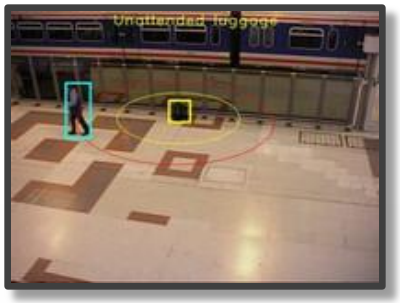
Video Processing

Video Processing

Video Processing

Motion Estimation

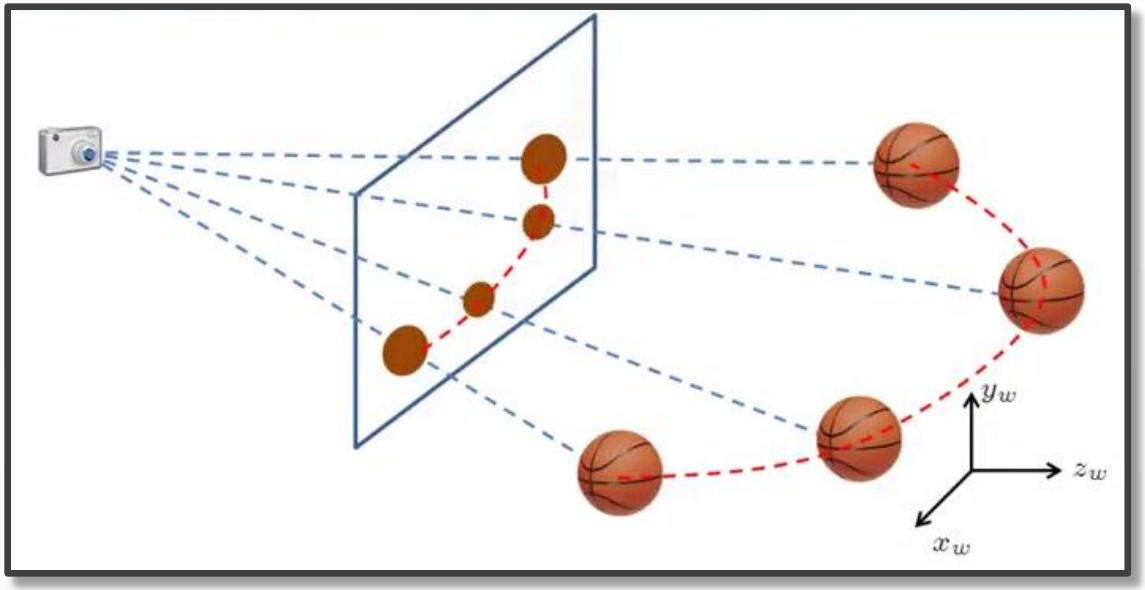
- The main different between a still image and a dynamic video is the motion.
- Estimating the motion on the video is a fundamental step for many algorithms:
  - Object tracking
  - Video surveillance
  - Human-Computer Interaction (HCI)
  - Video compression
  - Spatio-temporal filtering
  - Temporal Interpolation



Sources: http://www.indect-project.eu/benefits-for-the-security-of-citizens-selected-tools-and-applications
https://www.youtube.com/watch?v=3TUVTqbco90&list=PL_O5aD2shhGDzB0vPGLQak2Dk7-3cdGn5&index=1
http://mi.eng.cam.ac.uk/~cipolla/research.htm

3D vs. 2D Motion

- In general, we are interested in the motion in the 3D scene, but we can only work with it's 2D projection on the image plane:



- The interpretation of the 2D image can be ambiguous:
  - E.g. does the size of the object really changed or is it just further away from the camera?

Source of the image: Fundamentals of Digital Image and Video Processing lectures by Aggelos K. Katsaggelos

True and Apparent Motion

- We want to know how an object, an image region or a single pixel moved on the image plane from one frame to the next:

- But…
  - the change of the pixel does not necessarily means that motion occurred in the real 3D world
  - a real world change might not result a change in the pixel value.

True and Apparent Motion

- The change of the pixel does not necessarily means that motion occurred in the real 3D world:

True and Apparent Motion

- A real world change might not result a change in the pixel value:

Motion Estimation Methods

- Direct Methods: they compute the optical flow between two consecutive frames.
  - (Optical Flow: „… is the pattern of apparent motion of objects, surfaces, and edges in a visual scene caused by the relative motion between an observer (an eye or a camera) and the scene.” Wiki)
  - Phase Correlation
  - Block Matching
  - Spatio-Temporal Gradient
- Indirect methods:
  - Feature Matching:
    - locates feature points on both images, finds the pair of each feature point on the other image and calculates the motion based on the displacement of the feature points.
    - Different features can be used: SIFT, SURF, Harris,…

Phase Correlation

- Can be used to estimate global motion, for image registration.
- It is based on the translation property of the Fourier transform:

$$x(n_1 - m_1, n_2 - m_2) \leftrightarrow X(\omega_1, \omega_2)e^{-j\omega_1 m_1 - j\omega_2 m_2}$$

Noisy Original Image

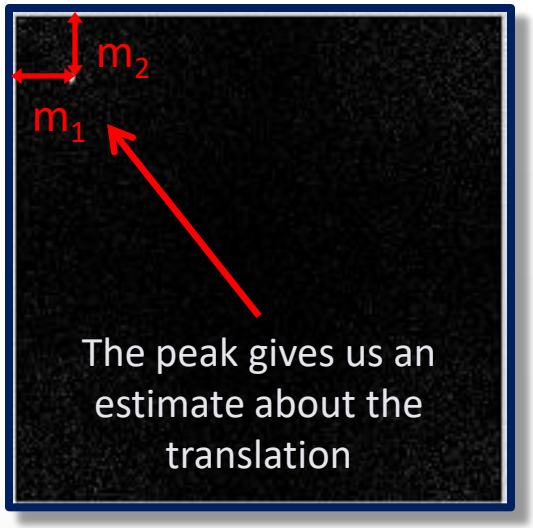
$x(n_1, n_2)$

$X(\omega_1, \omega_2)$

Noisy Translated Image

$x(n_1 - m_1, n_2 - m_2)$

$X(\omega_1, \omega_2)e^{-j\omega_1 m_1 - j\omega_2 m_2}$



Phase Correlation

Phase Correlation

- The assumption is that between two consecutive frames the image was shifted by (m_1, m_2):

$$x_t(n_1, n_2) \qquad x_{t+1}(n_1, n_2) = x_t(n_1 - m_1, n_2 - m_2)$$

$$\Downarrow \quad \textbf{Fourier Transform} \quad \Downarrow$$

$$X_t(k_1, k_2) \qquad X_t(k_1, k_2)e^{-j\frac{2\pi}{N_1}m_1k_1 - j\frac{2\pi}{N_2}m_2k_2}$$

- Calculation of the cross power spectrum:

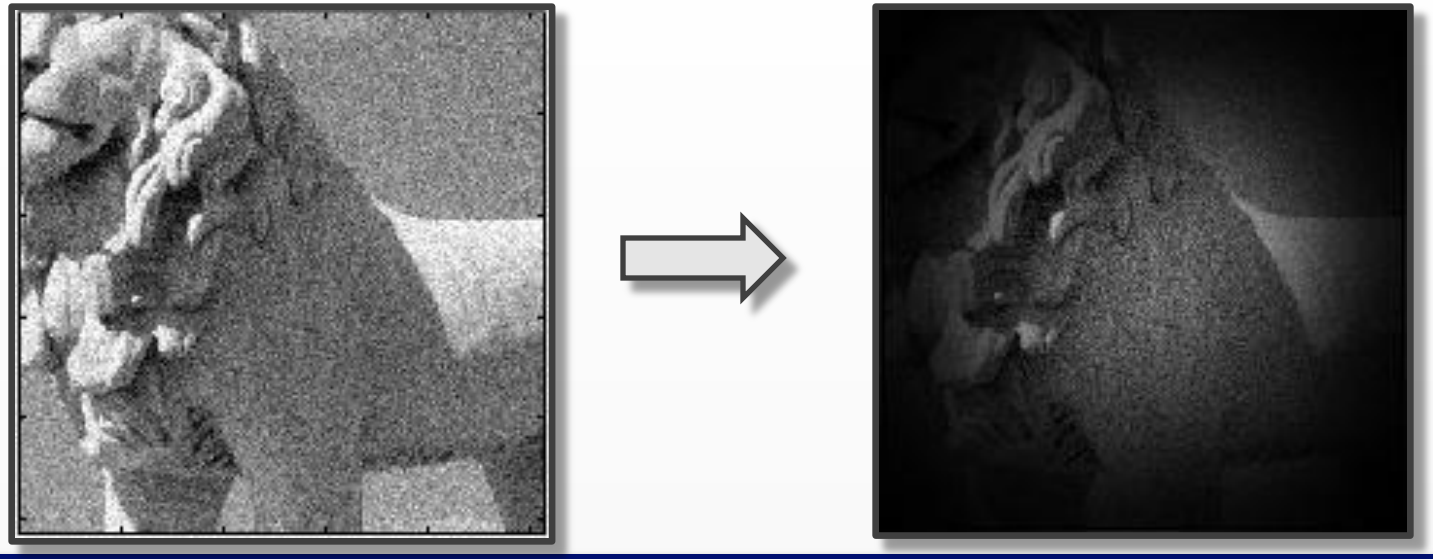
$$C(k_1, k_2) = \frac{X_t(k_1, k_2) \cdot X_{t+1}^*(k_1, k_2)}{\left|X_t(k_1, k_2) \cdot X_{t+1}^*(k_1, k_2)\right|} = \frac{\left|X_t(k_1, k_2)\right|^2 e^{-j\frac{2\pi}{N_1}m_1k_1 - j\frac{2\pi}{N_2}m_2k_2}}{\left|X_t(k_1, k_2)\right|^2}$$

- Transforming the ***C*** back to the spatial domain we get the normalized cross correlation:

$$c(n_1, n_2) = \delta(n_1 - m_1, n_2 - m_2)$$

Phase Correlation

- What happens at the border regions?
  - Since we use DFT the linear shifts becomes circular shifts
  - In most cases the images are related by linear shift not circular, so the border regions may corrupt our calculation.
  - Solution: use a 2D Hamming window (or something similar) to down weight the values close to the border:

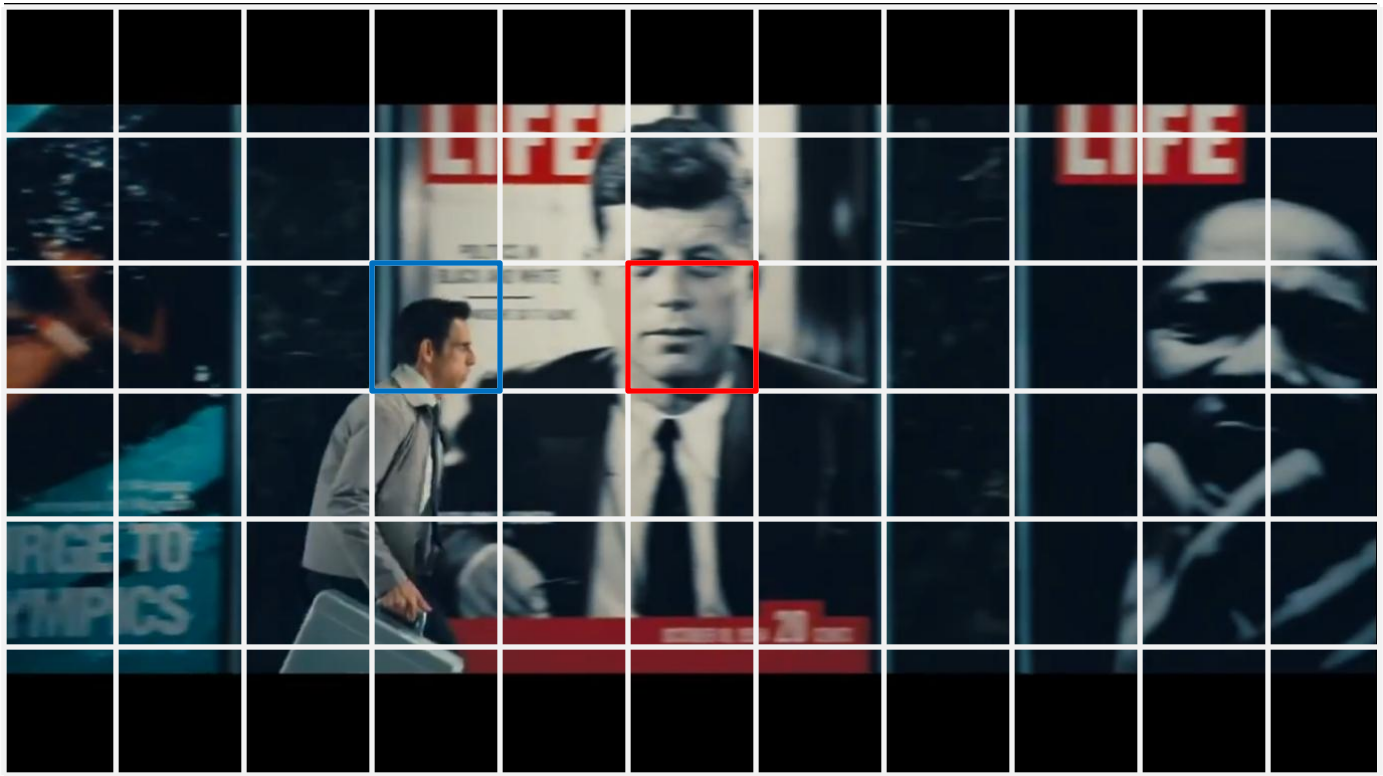


Block Matching

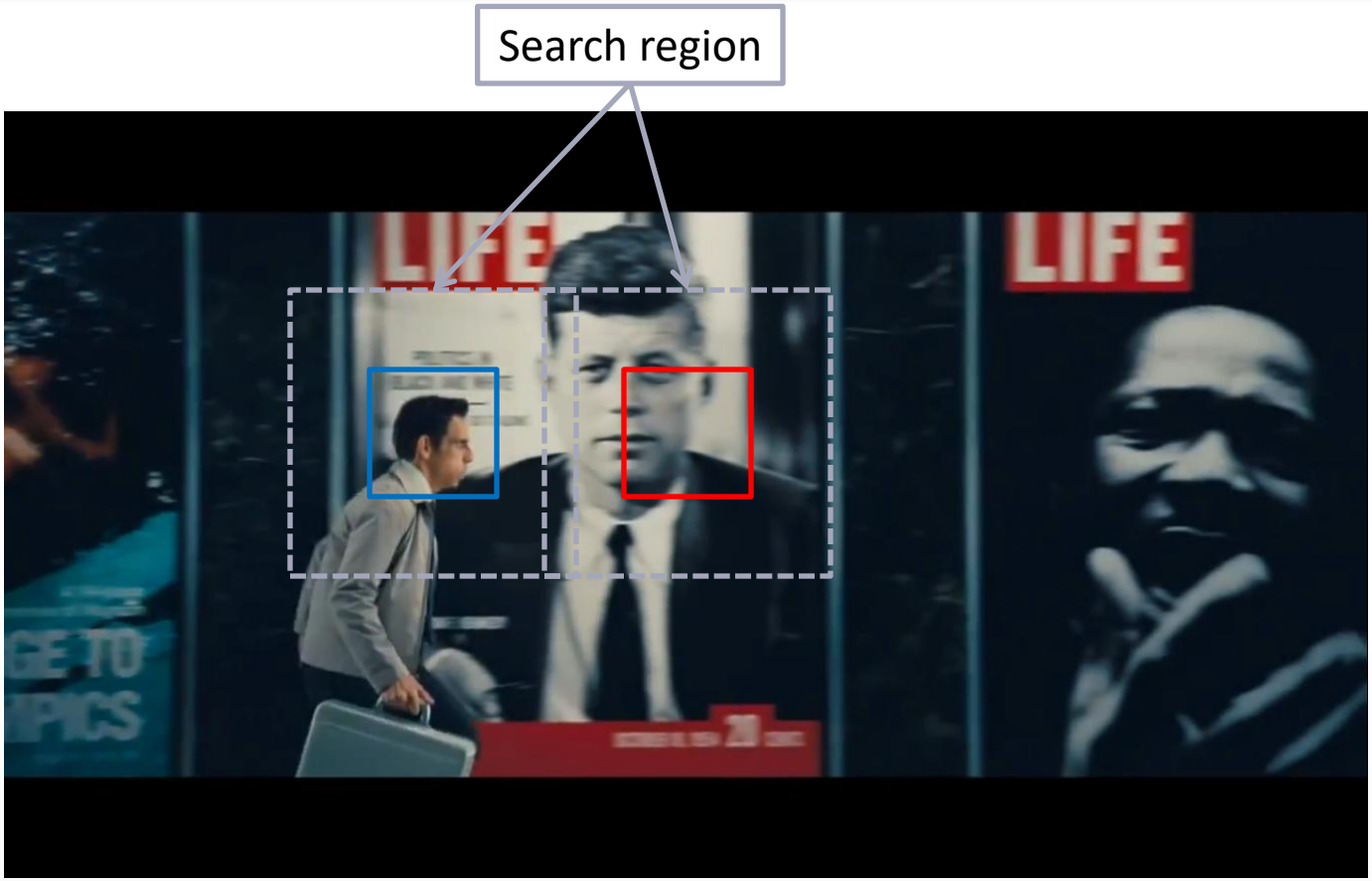
- An intuitive motion estimation algorithm, widely used in video compression.
- It is based on the following assumptions:
  - The objects are rigid
  - The illumination of the scene is constant
  - Objects are not entering or leaving the scene
  - The motion is parallel to the image plane
- It was originally introduced by Jain and Jain in 1981.
- To estimate the local motion between two frames (A and B)..
  1. Frame A is divided into (usually squared shaped) blocks.
  2. For each block of A, a search is made on frame B, to find its best matching pair.

J.R. Jain and A.K. Jain, "*Displacement measurement and its application in interframe image coding*", IEEE Trans. Commun., Vol. COM-29, No. 12, pp. 1799-1808, Dec. 1981.

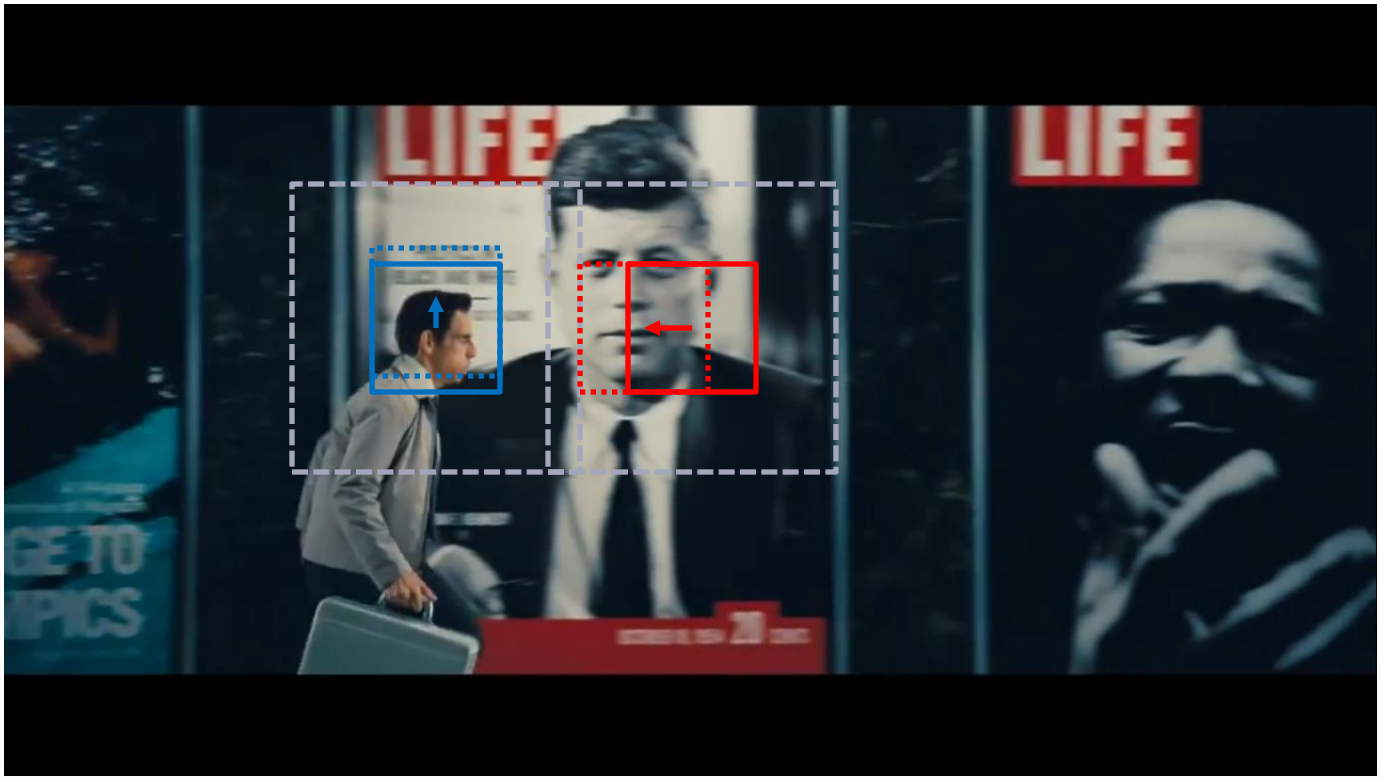
Block Matching



Block Matching

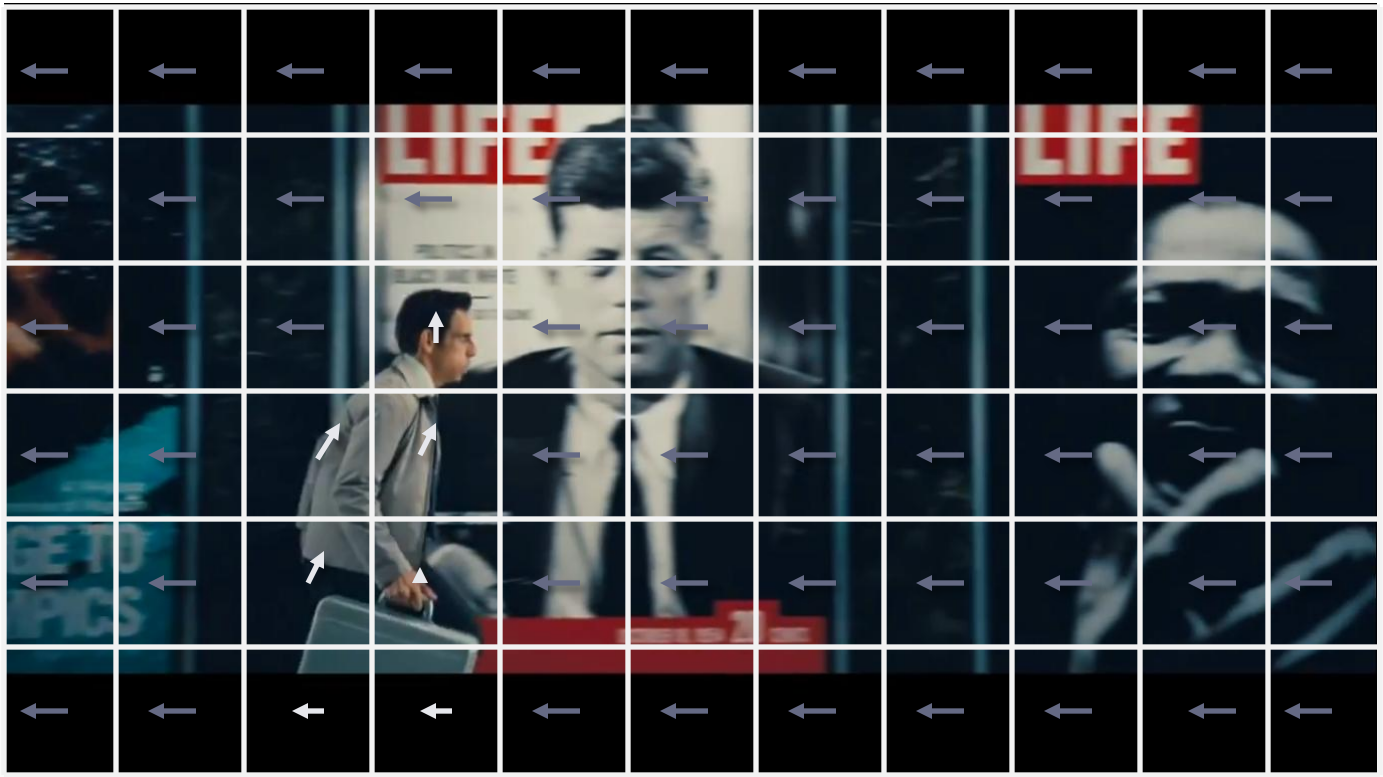


Block Matching



The best matching shift defines a motion vector for each block.

Block Matching



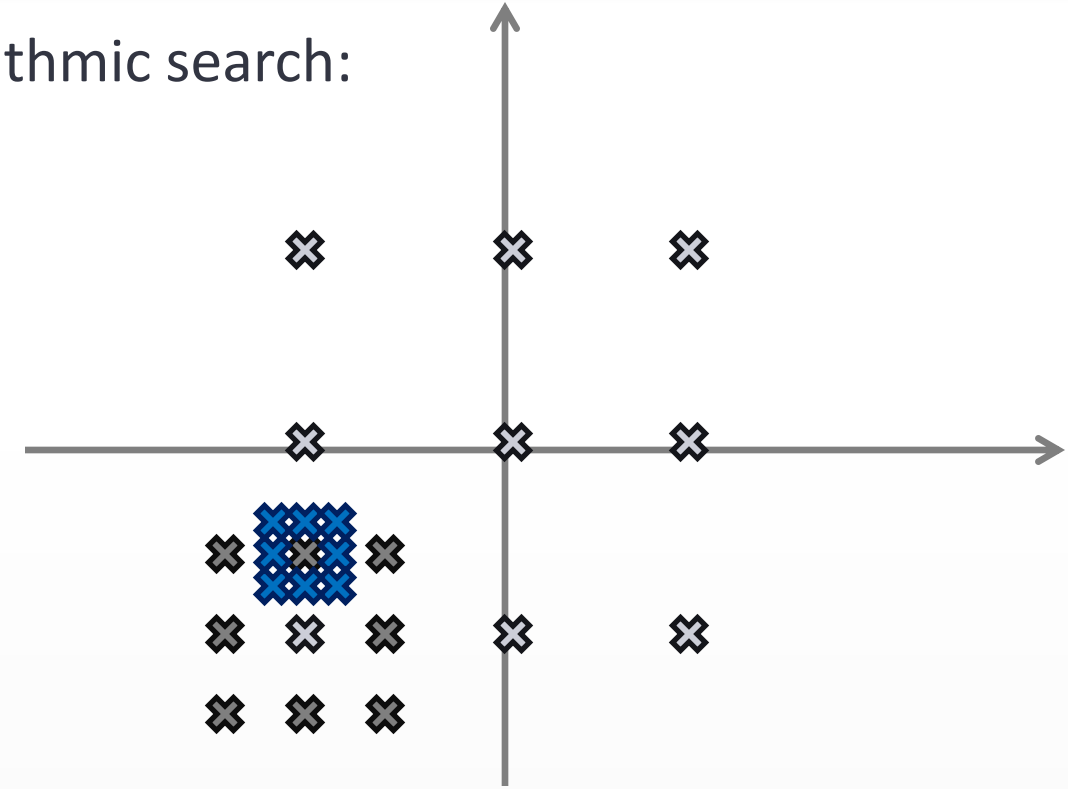
The best matching shift defines a motion vector for each block.

Block Matching

- Different measures can be used as matching criteria, to find the best fit for a block:
  - Mean Square Error
  - Mean Absolute Error
  - Correlation Function
- To get a dens field of motion vectors overlapping blocks can be used.
- To find the best match for a block with exhaustive search is computationally highly demanding.
- There are different ways to reduce the computational burden:
  - Reduced search window or block size
  - Block sub-sampling
  - Logarithmic search

Block Matching

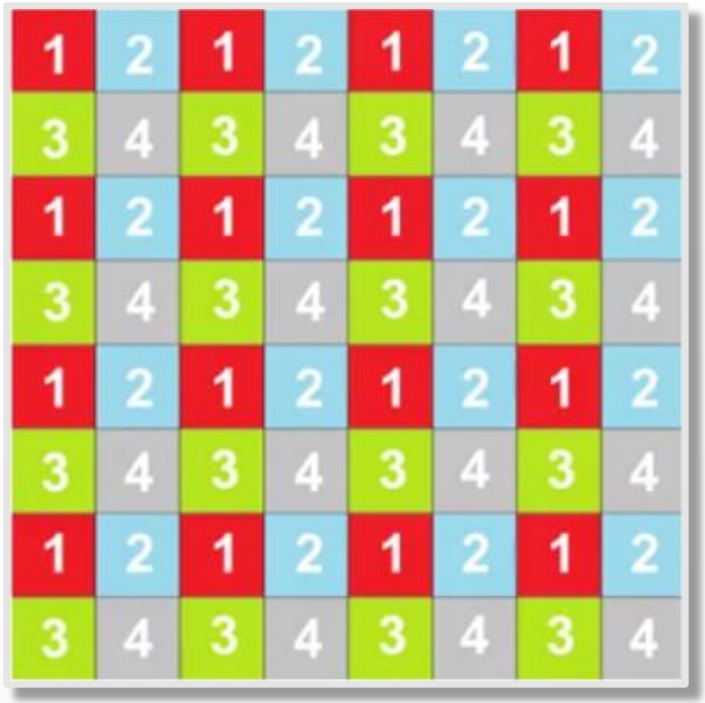
- A 2D logarithmic search:



- For reduced computational cost we risk to end up in a local minimum within the search region.

Block Matching

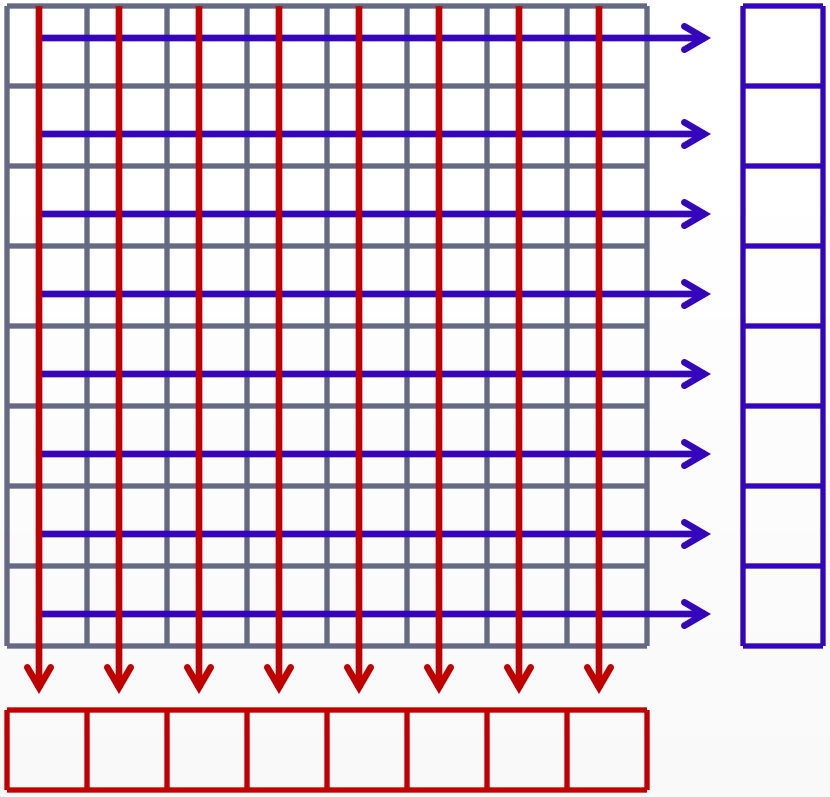
- Pixel Sub-Sampling:
  - Only ¼ of the pixels are used to calculate the matching criteria for a block:



  - Which 25% is used is changed from one location to an other.

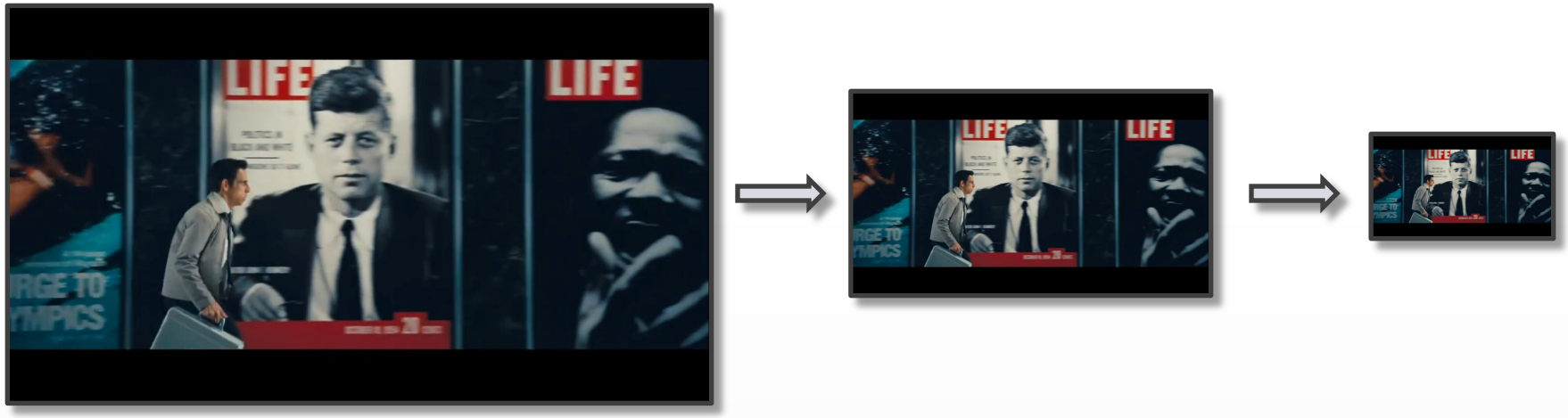
Block Matching

- Pixel Projection:
  - We compute the projection of the pixel values to the ***x*** and ***y*** axis and compare only the projected values:



Hierarchical Block Matching

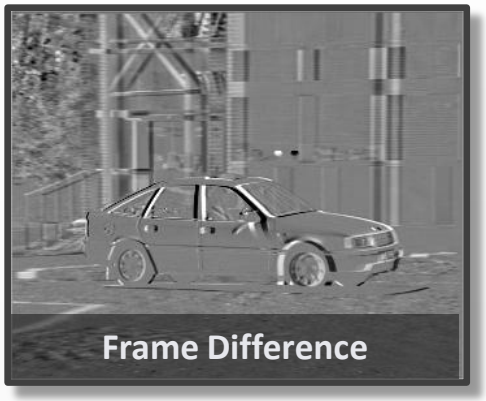
- Hierarchical Structure:
  - We reduce the size of the images:

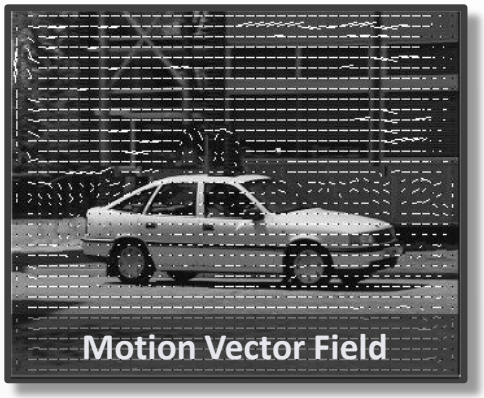


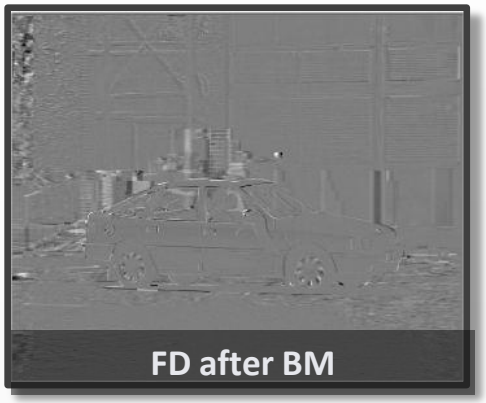
  - Do the motion estimation first at the lowest resolution, with relatively large search window.
  - Use the (properly up scaled) estimated vectors for initialization of the search at a higher level.
  - The final motion vectors are calculated at the original dimension.

Hierarchical Block Matching

- Examples created by VcDemo:







http://msp.ewi.tudelft.nl/content/image-and-video-compression-learning-tool-vcdemo

Optic Flow

- Constant brightness constraint: It is assumed that the brightness of an object remains the same from one frame to an other, hence all the changes in brightness are solely due to the motion in the scene.

reference frame

current frame

$$I(x, y, 0) = I(x + u, y + v, \tau)$$

where *u* and *v* are the displacement of the pixel in each dimension.

- Use Taylor series expansion:
  - The Taylor series expansion of a function *f*, that is infinitely differentiable at *a*, is given with the following formula:

$$\sum_{n=0}^{\infty} \frac{f^{(n)}(a)}{n!}(x-a)^n$$

Optic Flow

- Use Taylor series expansion:
  - Take the Taylor series expansion of *I(x+u, y+v,τ)* at 0:

The higher order terms are discarded

$$I(x+u, y+v, \tau) = I(x, y, 0) + \frac{\partial I(x, y, 0)}{\partial x} u + \frac{\partial I(x, y, 0)}{\partial y} v + \frac{\partial I(x, y, 0)}{\partial t} \tau + \text{H.O.T.}$$

$$0 = \underbrace{\frac{\partial I(x, y, 0)}{\partial x}}_{I_x} u + \underbrace{\frac{\partial I(x, y, 0)}{\partial y}}_{I_y} v + \underbrace{\frac{\partial I(x, y, 0)}{\partial t}}_{I_t} \tau$$

$$0 = I_x u + I_y v + I_t \tau$$

Optic Flow

Divide everything with τ, we get velocities

$$0 = I_x u + I_y v + I_t \tau$$

The optic flow equation:

$$0 = I_x V_x + I_y V_y + I_t$$

- This equation is under determined: 2 unknowns, 1 equation.
- Assuming that neighboring pixels undergo the same motion we can increase the number of equations:

$$I_x(n_1)V_x + I_y(n_1)V_y = I_t(n_1)$$
$$I_x(n_2)V_x + I_y(n_2)V_y = I_t(n_2)$$
$$\vdots \qquad \vdots \qquad \vdots$$
$$I_x(n_N)V_x + I_y(n_N)V_y = I_t(n_N)$$

We consider $n_1 \ldots n_N$ pixels in the neighborhood

Optic Flow

- The equation system can be written in matrix vector form:

$$\begin{aligned} I_x(n_1)V_x + I_y(n_1)V_y &= I_t(n_1) \\ I_x(n_2)V_x + I_y(n_2)V_y &= I_t(n_2) \\ \vdots \quad\quad & \quad\quad \vdots \quad\quad \vdots \\ I_x(n_N)V_x + I_y(n_N)V_y &= I_t(n_N) \end{aligned}$$

$$\begin{bmatrix} I_x(n_1) & I_y(n_1) \\ I_x(n_2) & I_y(n_2) \\ \vdots & \vdots \\ I_x(n_N) & I_y(n_N) \end{bmatrix} \cdot \begin{bmatrix} V_x \\ V_y \end{bmatrix} = \begin{bmatrix} I_t(n_1) \\ I_t(n_2) \\ \vdots \\ I_t(n_N) \end{bmatrix}$$

$$\mathrm{A} \cdot \mathrm{x} = \mathrm{b}$$

- This is an inverse problem: we know **A** and **b** and look for **x**.
- Least Square or constrained least square solution can be used:

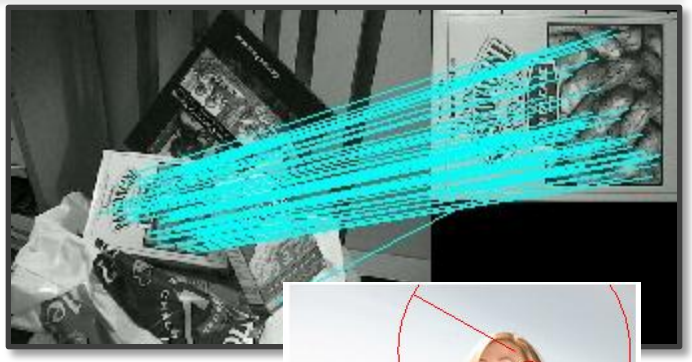
$$\mathrm{x} = \left(\mathrm{A}^T\mathrm{A}\right)^{+}\mathrm{A}^T\mathrm{b} \qquad \mathrm{x} = \left(\mathrm{A}^T\mathrm{A} + \lambda C^T C\right)^{-1}\mathrm{A}^T\mathrm{b}$$

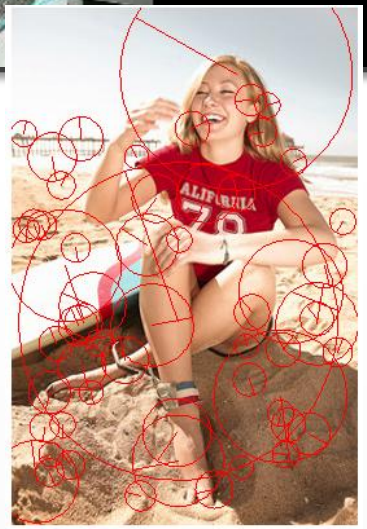
Feature Matching

- The algorithm is based on the matching of key points that has a well defined position on the image (e.g. corner).
- A descriptor is given to each point, that is preferably…
  - Shift and rotation invariant
  - Robust against the changes of illumination
  - Scale invariant
  - Low dimensional
  - Robust to noise
  - …
- From the matched point pairs the global motion of the camera can be estimated.

Feature Matching

- Feature point detectors:
  - Harris corners
  - SIFT (Scale-Invariant Feature Transform)
  - SURF (Speeded Up Robust Features)
- Feature Descriptors:
  - SIFT
  - SURF
  - HOG (Histogram of Oriented Gradients)
  - LBP (Local Binary Patterns)
  - ...





C. Harris and M. Stephens, "Proceedings of the 4th Alvey Vision Conference". pp. 147–151, 1988.

David G. Lowe, "Distinctive image features from scale-invariant keypoints," International Journal of Computer Vision, 60, 2 (2004), pp. 91-110

Herbert Bay, Andreas Ess, Tinne Tuytelaars, Luc Van Gool, "SURF: Speeded Up Robust Features", Computer Vision and Image Understanding (CVIU), Vol. 110, No. 3, pp. 346--359, 2008.

Navneet Dalal, Bill Triggs, "Histograms of Oriented Gradients for Human Detection" CVPR, June 2005.

Ahonen, T., Hadid, A. and Pietikäinen, M., "Face Description with Local Binary Patterns: Application to Face Recognition". IEEE Trans. Pattern Analysis and Machine Intelligence 28(12):2037-2041, 2006

Object Tracking

- The objective of ***object tracking*** is to associate target objects in consecutive video frames.
- Can be applied in…
  - human-computer interaction
  - traffic monitoring
  - vehicle navigation
  - motion-based recognition
  - video indexing
  - automated surveillance
- The tracking task can be divided into two subtasks:
  - Build a ***model of the object*** you want to track
  - Use what you know about where the object was in the previous frame(s) to ***make predictions*** about the current frame ***to restrict the search.***
- Repeat the two subtasks and possibly update the model.

Source: http://cvpr.uni-muenster.de/teaching/ws11/ComputerVisionundMustererkennungWS11/script/CVME-11-Tracking.ppt

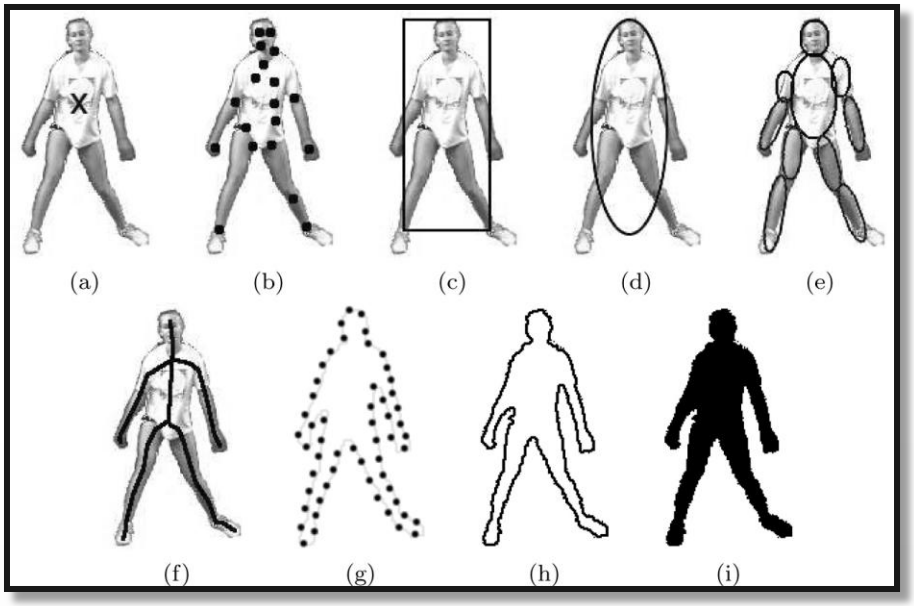
Object Tracking

- Tracking objects can be complex due to:
  - Loss of information caused by projection from 3D to 2D
  - Noise
  - Complex object shapes/motion
  - Non-rigid or articulated nature of objects
  - Partial and full occlusions of the object
  - Changes of the illumination
  - Real-time processing requirements
- Simplify tracking by making assumptions and the use of prior information:
  - The motion of the object is smooth with no abrupt changes
  - The object motion is assumed to be of constant velocity
  - Prior knowledge about the number and the size of objects, or the object appearance.

Source: http://cvpr.uni-muenster.de/teaching/ws11/ComputerVisionundMustererkennungWS11/script/CVME-11-Tracking.ppt

Object Tracking

- Modeling of the object:
  - Shape:
    - Point model
    - Simple geometrical forms
    - Contour/Silhouette
    - Articulated shape models
    - Skeletal models
  - Appearance:
    - Template (if the appearance is not changing)
    - Probabilistic representation of the object appearance (e.g. Histogram)
  - Different features can be used to describe the appearance:
    - Color, Edge, Motion, Texture
    - HOG, SIFT, LBP, ….



Source: http://cvpr.uni-muenster.de/teaching/ws11/ComputerVisionundMustererkennungWS11/script/CVME-11-Tracking.ppt

Object Tracking

- Tracking-Learning-Detection (TLD) by Zdenek Kalal

http://personal.ee.surrey.ac.uk/Personal/Z.Kalal/tld.html

Z. Kalal, K. Mikolajczyk, and J. Matas, “Tracking-Learning-Detection,” *Pattern Analysis and Machine Intelligence* 2011.

Main Sources and Further Reading

Fundamentals of Digital Image and Video Processing lectures by Aggelos K. Katsaggelos

Tracking:

- http://cvpr.uni-muenster.de/teaching/ws11/ComputerVisionundMustererkennungWS11/script/CVME-11-Tracking.ppt
- Z. Wu, A. Thangali, S. Sclaroff, and M. Betke. "Coupling Detection and Data Association for Multiple Object Tracking.'' In Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, Rhode Island, June, 2012.
  https://www.youtube.com/watch?v=3TUVTqbco90&list=PL_O5aD2shhGDzB0vPGLQak2Dk7-3cdGn5&index=1
- A. Yilmaz, O. Javed, and M. Shah: Object tracking: A survey. ACM Computing Surveys, Vol. 38, No. 4, 1-45, 2006
- Z. Kalal, K. Mikolajczyk, and J. Matas, “Tracking-Learning-Detection,” *Pattern Analysis and Machine Intelligence* 2011.

Human Computer Interaction:

- http://mi.eng.cam.ac.uk/~cipolla/research.htm
- http://www.nada.kth.se/cvap/gvmdi/

Feature Descriptors:

- C. Harris and M. Stephens, "Proceedings of the 4th Alvey Vision Conference". pp. 147–151, 1988.
- David G. Lowe, "Distinctive image features from scale-invariant keypoints," International Journal of Computer Vision, 60, 2 (2004), pp. 91-110
- Herbert Bay, Andreas Ess, Tinne Tuytelaars, Luc Van Gool, "SURF: Speeded Up Robust Features", Computer Vision and Image Understanding (CVIU), Vol. 110, No. 3, pp. 346--359, 2008.
- Navneet Dalal, Bill Triggs, "Histograms of Oriented Gradients for Human Detection„ International Conference on Computer Vision & Pattern Recognition - June 2005.
- Ahonen, T., Hadid, A. and Pietikäinen, M., Face Description with Local Binary Patterns: Application to Face Recognition. IEEE Trans. Pattern Analysis and Machine Intelligence 28(12):2037-2041, 2006
- M. Calonder, V. Lepetit, M. Ozuysal, T. Trzcinski, C. Strecha, and P. Fua, ”Computing a Local Binary Descriptor Very Fast,” IEEE Transactions on Pattern Analysis and Machine Intelligence 2012