Video Compression - Cours 1

Nicolas Boutry[1]

↪ nicolas.boutry@lrde.epita.fr

[1] Laboratoire de Recherche et Développement de l'EPITA (LRDE), France

Octobre 2019

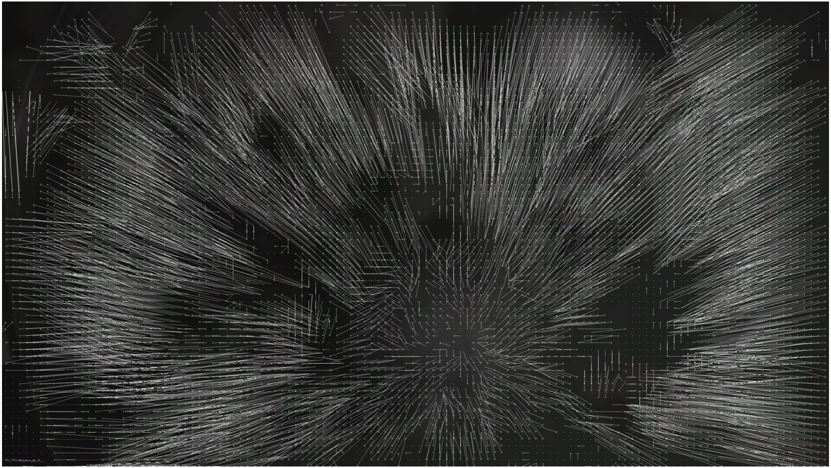
Outline

Outline

Summary I

- Course 1
  - Main concepts of the course
  - Optical Flow
  - Motion Estimation (pixel-wise / block-wise)
  - Notebook on ME (pixel-wise vs. block-based)
- Course 2
  - Motion Estimation (node-based / mesh-based)
  - Brief presentation of the other possible approaches
  - Notebook on ME (node-based vs. mesh-based)
- Course 3
  - Motion Compensation (MC)
  - Scalability and granularity
  - Notebook on MC and Video Compression

Motion estimation I

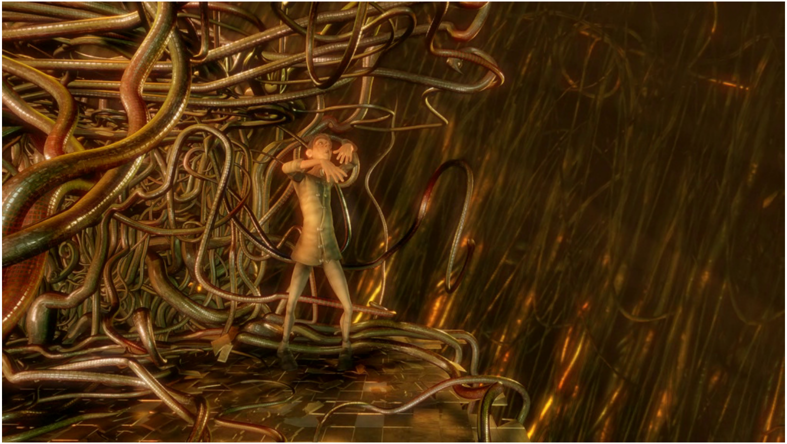


Motion estimation II

- Motion estimation is the process of determining motion vectors that describe the transformation from one 2D image to another.
- It is an ill-posed problem as the motion is in three dimensions but the images are a projection of the 3D scene onto a 2D plane.

Motion Compensation I

- Motion compensation (MC) is an algorithmic technique based on ME used to predict a frame in a video, given the previous and/or future frames.



Motion Compensation II

Figure: A frame ϕ_1 extracted from a video.

Motion Compensation III

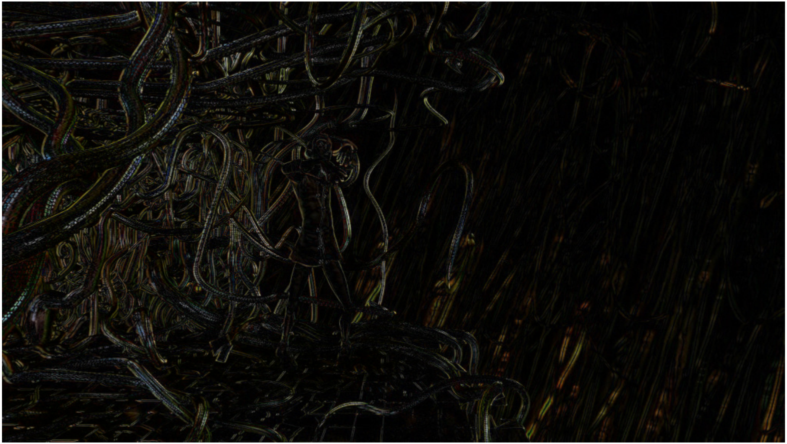


Figure: The difference between ϕ_1 and its following frame ϕ_2.

Motion Compensation IV

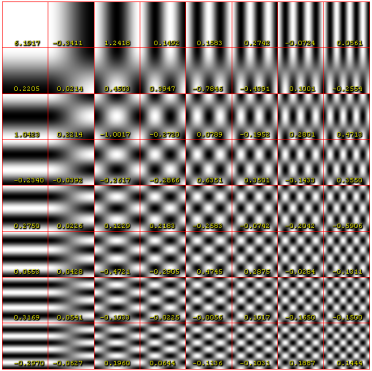
Figure: The difference between the MC-predicted frame $\hat{\phi}_2$ and ϕ_2 (much easier to compress !!!).

Motion Compensation V

- When we want to encode two consecutive frames ϕ_1 (reference) and ϕ_2 (current) using MC, we will then have to encode (for example):
  - The first frame ϕ_1 (using DCT),
  - The MVs predicting ϕ_2 from ϕ_1,
  - The prediction error $\varepsilon = \phi_2 - \hat{\phi}_2$
- Note: the reference picture may be previous in time or even from the future.
- Most video coding standards (H.26x, MPEGs) use motion-compensated DCT video coding (block motion compensation).

Motion Compensation VI

A brief recall about DCT-based transform:



Scalable video coding I

- Scalability refers to the capability of recovering physically meaningful image or video information from decoding only partial compressed bitstreams,
  - Quality scalability: finer to finer quantizations,
  - Spatial scalability: different spatial resolutions (Laplacian Pyramid, ...),
  - Temporal scalability (we can jump frames and add the missing ones progressively),
  - Frequency scalability (lower frequencies to higher frequencies),
  - Combination of basic schemes
  - Granularity (coarse vs fine ones)
- Object-based scalability (different resolutions for different objects)

Outline

2D motion vs. optical flow I

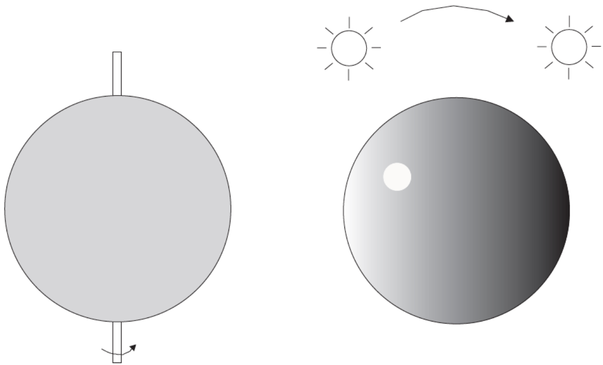


Figure 6.1. The optical flow is not always the same as the true motion field. In (a), a sphere is rotating under a constant ambient illumination, but the observed image does not change. In (b), a point light source is rotating around a stationary sphere, causing the highlight point on the sphere to rotate. Adapted from [17, Fig.12-2].

2D motion vs. optical flow II

- We can observe movements where there are not, and observe that there is no motion when there are!
- The observed or apparent 2D motion is referred to as optical flow in computer vision literature.
- In brief, the optical flow may not be the same as the true 2D motion.

Optical flow equation and ambiguity in motion estimation I

- Consider a video sequence whose luminance is variable in time $\psi(x, y, t)$.
- Assume that an point (x, y) at time t is moved to $(x + d_x, y + d_y)$ at time $t + d_t$.
- Under the constant intensity assumption, the images of the same object point at different times have the same luminance value:

$$\psi(x + d_x, y + d_y, t + d_t) = \psi(x, y, t)$$

Optical flow equation and ambiguity in motion estimation II

- Using Taylor's expansion, when d_x, d_y and d_t are small, then we have:

$$\psi(x + d_x, y + d_y, t + d_t) = \psi(x, y, t) + \frac{\partial \psi}{\partial x} d_x + \frac{\partial \psi}{\partial y} d_y + \frac{\partial \psi}{\partial t} d_t$$

- We obtain that:

$$\frac{\partial \psi}{\partial x} d_x + \frac{\partial \psi}{\partial y} d_y + \frac{\partial \psi}{\partial t} d_t = 0,$$

which shows the relation between the motion vector (d_x, d_y) and d_t.

Optical flow equation and ambiguity in motion estimation III

- Let us define $v_x = d_x/d_t$ and $v_y = d_y/d_t$, then $\boldsymbol{v} := (v_x, v_y)$ is the velocity vector and we obtain that:

$$\frac{\partial \psi}{\partial x}\, v_x + \frac{\partial \psi}{\partial y}\, v_y + \frac{\partial \psi}{\partial t} = 0,$$

which can be written:

$$\nabla \psi^T\, \boldsymbol{v} + \frac{\partial \psi}{\partial t} = 0,$$

with $\nabla \psi = \left[\frac{\partial \psi}{\partial x}, \frac{\partial \psi}{\partial y}\right]^T$ is the spatial gradient vector of $\psi(x, y, t)$.

- From now on, we will write $\boldsymbol{x} := (x, y)$,

Optical flow equation and ambiguity in motion estimation IV

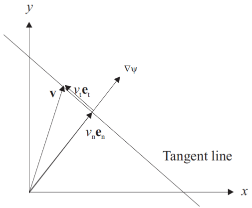


Figure 6.2. Decomposition of motion $\mathbf{v}$ into normal $(v_n \mathbf{e}_n)$ and tangent $v_t \mathbf{e}_t$ components. Given $\nabla\psi$ and $\frac{\partial \psi}{\partial t}$, any MV on the tangent line satisfies the optical flow equation.

- The flow vector $\boldsymbol{v}$ at any point x can be decomposed into two orthogonal components:

$$\boldsymbol{v} = v_n \, e_n + v_t \, e_t,$$

- As we can observe, when a straight edge moves in the plane, we can only detect the normal v_n of its motion vector !

Optical flow equation and ambiguity in motion estimation V

- Because $\nabla\psi = ||\nabla\psi||\ e_n$, the optical flow equation can be rewritten as:

$$v_n\ ||\nabla\psi|| + \frac{\partial\psi}{\partial t} = 0,$$

  where $||\nabla\psi||$ is the magnitude of the gradient vector.

- The consequences of these equations are:

1 (A) At any pixel x, one cannot determine the motion vector v based on $\nabla\psi$ and $\frac{\partial\psi}{\partial t}$ alone: there is only one equation for two unknowns (v_x and v_y, or v_n and v_t).

1 (B) In fact, the undetermined component is v_t.To solve both unknowns, one needs to impose additional constraints.

1 (C) The most common constraint is that the flow vectors should vary smoothly spatially (regularity).

2 (A) We can compute:

$$v_n = -\frac{\frac{\partial\psi}{\partial t}}{||\nabla\psi||},$$

whatever the value of v_t.

Optical flow equation and ambiguity in motion estimation VI

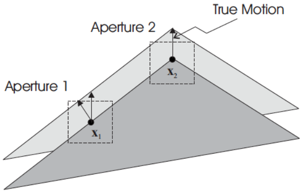


Figure 6.3. The aperture problem in motion estimation: To estimate the motion at $\mathbf{x}_1$ using aperture 1, it is impossible to determine whether the motion is upward or perpendicular to the edge, because there is only one spatial gradient direction in this aperture. On the other hand, the motion at $\mathbf{x}_2$ can be determined accurately, because the image has gradient in two different directions in aperture 2. Adapted from [39, Fig. 5.7].

2 (B) This ambiguity in estimating the motion vector is known as the aperture problem.

2 (C) The motion can be estimated uniquely only if the aperture contains at least two different gradient directions.

Optical flow equation and ambiguity in motion estimation VII

3 (A) In regions with constant brightness so that $\|\nabla\psi\| = 0$, the flow vector is indeterminate.

3 (B) This is because there is no perceived brightness changes when the underlying surface has a flat pattern.

3 (C) The estimation of motion is reliable only in regions with brightness variation, i.e., regions with edges or non-flat textures.

Outline

General methodologies

General methodologies I

- We consider the ME between two given frames, $\psi(x, y, t_1)$ and $\psi(x, y, t_2)$.
- The MV at x between time t_1 and t_2 is defined as the displacement of this point from t_1 to t_2.
- We will call the frame at time t_1 the tracked/reference frame, and the frame at t_2 the anchor/current frame.
- Depending on the intended application, the anchor frame can be either before or after the tracked frame in time.

General methodologies II

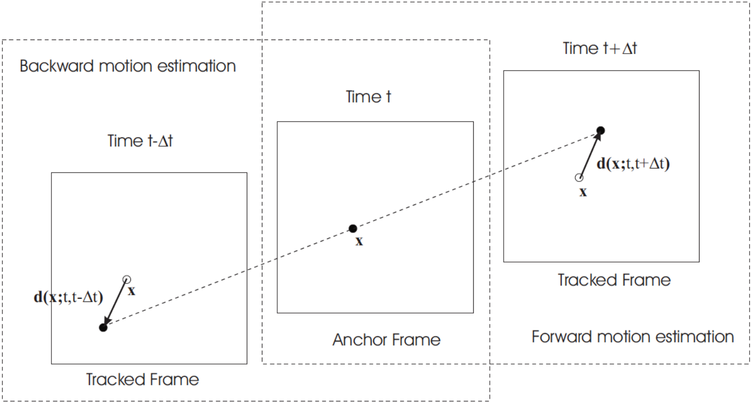


Figure 6.4. Forward and backward motion estimation. Adapted from [39, Fig. 5.5].

- The problem is referred as to as forward motion estimation when $t_1 < t_2$ and as backward motion estimation when $t_2 < t_1$.

General methodologies III

- For notation convenience, from now on, we use $\psi_1(\boldsymbol{x})$ and $\psi_2(\boldsymbol{x})$ to denote the reference and current frames respectively.
- In general, we can represent the MV as $d(x; \boldsymbol{a})$ where:

$$\boldsymbol{a} = [a_1, a_2, \ldots, a_n]^T$$

  is a vector containing all the motion parameters.
- Similarly, the mapping function can be denoted as $w(\boldsymbol{x}; \boldsymbol{a}) = \boldsymbol{x} + d(\boldsymbol{x}; \boldsymbol{a})$,
- The ME problem is to estimate the motion parameter vector $\boldsymbol{a}$.

General methodologies IV

- Methods that have been developed can be categorized into two groups: feature-based (matching) and intensity-based.
- We will only see the intensity-based approach.
- This approach applies the constant intensity assumption or the optical flow equation at every pixel.

Motion representation

Motion representation I

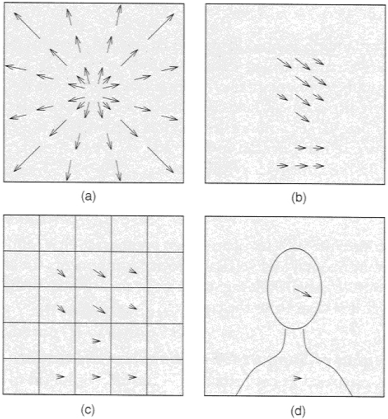


Figure 6.5. Different motion representations: (a) global, (b) pixel-based, (c) block-based, and (d) region-based. From [38, Fig. 3].

Motion representation II

- A key problem in ME is how to parameterize the motion field (translations, polynomial motions, rotations, etc.).
- However, usually, there are multiple objects in the scene that move differently,

Motion representation III

- When there are several objects in an image moving in different directions:
  - The most direct and unconstrained approach is to specify the motion vector at every pixel; this is the so called pixel-based representation.
  - It requires the estimation of a large numbers of unknowns (twice the number of pixels!)
  - The solution can often be physically incorrect!
  - Then we need physical constraints like regularity of the MV field.

Motion representation IV

- If only the camera is moving or the imaged scene contains a single moving object within a planar surface:
  - One could use a global motion representation to characterize the entire motion field.

Motion representation V

- In general:
  - It is more appropriate to divide an image frame into multiple regions so that the motion within each region can be characterized well by a parameterized model.
  - This is known as region-based motion representation
  - Very complicated in practice: do we estimate first the motion and then the regions? or the regions are then the motions? Or both in the same time?

Motion representation VI

- To reduce the complexity: we can decompose the image into small blocks (block-based representation).

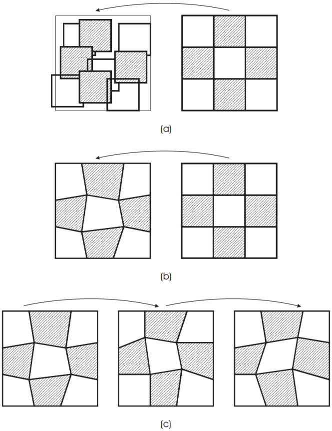


Figure 6.13. Comparison of block-based and mesh-based motion representations: (a) Block-based motion estimation between two frames, using a translational model within each block in the anchor frame; (b) Mesh-based motion estimation between two frames, using a regular mesh at the anchor frame; (d) Mesh-based motion tracking, using the tracked mesh for each new anchor frame.

Motion representation VII

- The simplest version models the motion in each block by a constant translation (all over the block),
- The estimation problem becomes that of finding one MV for each block.
- This method provides a good compromise between accuracy and complexity.

Motion representation VIII

- The block-based approach does not impose any constraint on the motion transition between adjacent blocks.
- The resulting motion is often discontinuous across block boundaries.

Motion representation IX

- One approach to overcome this problem is by using a mesh-based representation where we deform the image as a mesh in time.
- In this method, the image frame is partitioned into non-overlapping polygonal elements.
- This representation induces a motion field that is continuous everywhere.

Motion representation X

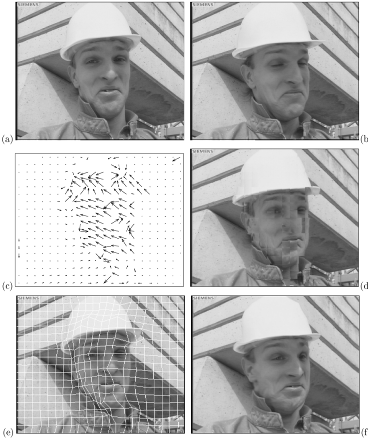


Figure 6.8. Example motion estimation results: (a) the tracked frame; (b) the anchor frame; (c-d) motion field and predicted image for the anchor frame (PSNR=29.86 dB) obtained by half-pel accuracy EBMA ; (e-f) motion field (represented by the deformed mesh overlaid on the tracked frame) and predicted image (PSNR=29.72 dB) obtained by the mesh-based motion estimation scheme in [43].

- It is not the ultimate solution: it can introduce warping effects!

Motion representation XI

Motion estimation criteria

Motion estimation criteria I

- We can simply minimize the error based on the Displaced Frame Difference (DFD):

$$E_{DFD}(a) = \sum_{\boldsymbol{x} \in \Lambda} \left| \psi_2(w(\boldsymbol{x}; \boldsymbol{a})) - \psi_1(\boldsymbol{x}) \right|^p ,$$

  where Λ is the domain of all pixels in ψ_1, and p is a positive number.

- When $p = 1$, the above error is called mean absolute difference (MAD), and when $p = 2$, the mean squared error (MSE).

- The error image $e(\boldsymbol{x}; \boldsymbol{a}) = \psi_2(w(\boldsymbol{x}; \boldsymbol{a})) - \psi_1(\boldsymbol{x})$ is usually called displaced frame difference (DFD) image.

- When a is optimal (case $p = 2$):

$$\frac{\partial E_{DFD}}{\partial \boldsymbol{a}} = 2 \sum_{\boldsymbol{x} \in \Lambda} (\psi_2(w(\boldsymbol{x}; \boldsymbol{a})) - \psi_1(\boldsymbol{x})) \frac{d(\boldsymbol{x})}{d\boldsymbol{a}} \nabla \psi_2(w(\boldsymbol{x}; \boldsymbol{a})) = 0,$$

Motion estimation criteria II

- Or we can minimize the error relative to the optical flow:
- Let $\psi_1(x, y) = \psi(x, y, t)$ and $\psi_2(x, y) = \psi(x, y, t + d_t)$. If d_t is small, we can assume that:

$$\frac{\partial \psi}{\partial t} d_t = \psi_2(x) - \psi_1(x),$$

- Then the optical flow equation becomes:

$$\frac{\partial \psi_1}{\partial x} d_x + \frac{\partial \psi_1}{\partial y} d_y + (\psi_2 - \psi_1) = 0,$$

or equivalently:

$$\nabla \psi_1^T d + (\psi_2 - \psi_1) = 0.$$

Motion estimation criteria III

- It is equivalent to minimize:

$$E_{flow} = \sum_{x \in \Lambda} \left| \nabla \psi_1(x)^T d(x; a) + \psi_2(x) - \psi_1(x) \right|^p ,$$

- This solution verifies when $p = 2$:

$$\frac{\partial E_{flow}}{\partial a} = 2 \sum_{x \in \Lambda} (\nabla \psi_1(x)^T d(x; a) + \psi_2(x) - \psi_1(x)) \frac{\partial d(x; a)}{da} \nabla \psi_1(x)$$

- However, the optical flow equation is valued only when the motion is small
- In practice,the DFD error criterion, and find the minimal solution using the <u>gradient descent</u> or <u>exhaustive search method</u>.

Motion estimation criteria IV

- About constant intensity assumption: minimizing the DFD error or solving the optical flow equation does not always give physically meaningful motion estimate.
- This is partially because the constant intensity assumption is not always correct.
- Regularization: the ME problem is ill-defined, and then to obtain physically meaningful solutions, one need to impose additional constraints to regularize the problem.
- To this aim, we can add a penalty term in our equation to enforce the smoothness of our vector field (i.e. it must vary smoothly across space except at boundaries):

$$E_s = \sum_{x \in \Lambda} \sum_{y \in N_x} \|d(x; a) - d(y; a)\|^2$$

where N_x is the set of adjacent pixels of x,

Motion estimation criteria V

- Then we want to minimize:

$$E_{total} = E_{DFD} + w_s \; E_s$$

  with w the weighting coefficient.

- We have to regularize but not too much! (to avoid over-blurring).

Minimization methods

Minimization methods I

- Minimization methods are various, but we will see only exhaustive search and gradient-based search methods.
- Usually, for the exhaustive search, the MAD is used for reasons of computational complexity, whereas for the gradient-based search, the ME is used for its mathematical tractability (we can compute the derivation !).
- The exhaustive search method guarantees reaching the global minimum.
- However, such search is feasible only if the number of unknown parameters is small, and each parameter takes only a finite set of discrete values.
- To reduce the search time, various fast algorithms can be developed, which achieve sub-optimal solutions.
- The most common gradient descent methods include the steepest gradient descent and the Newton-Raphson method.
- A gradient descent method can handle unknown parameters in a high dimensional continuous space.

Minimization methods II

- However, it can only guarantee the convergence to a local minimum.
- The error functions used in video processing are usually not convex and can have many local minima that are far from the global minimum.
- Therefore, it is important to obtain a good initial solution through the use of a prior knowledge, or by adding a penalty term to make the error function convex.

Minimization methods III

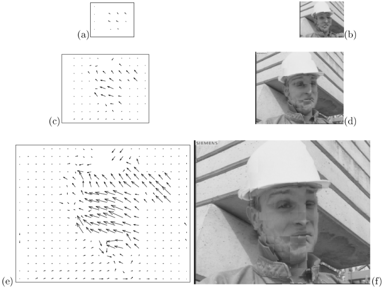


Figure 6.21. Example motion estimation results by HBMA for the two images shown in Fig. 6.8: (a-b) the motion field and predicted image at level 1; (c-d) the motion field and predicted image at level 2; (e-f) the motion field and predicted image at the final level (PSNR=29.32); A three-level HBMA algorithm is used. The block size is 16×16 at all levels. The search range is 4 at all levels with integer-pel accuracy. The result in the final level is further refined by a half-pel accuracy search in the range of ± 1.

- One important search strategy is to use a multi-resolution representation of the motion field and conduct the search in a hierarchical manner.

Minimization methods IV

- The basic idea is too first search the motion parameters in a coarse resolution, propagate this solution into a finer resolution, and then refine the solution in the finer resolution.
- It can combat both the slowness of exhaustive search methods and the non-optimality of gradient-based methods.

Regularization using motion smoothness constraint

Regularization using motion smoothness constraint I

- Since pixel-based motion representation is ill-defined, we need regularization techniques.
- Horn and Schunck propose to estimate the motion vectors by minimizing the following energy function:

$$E = \sum_{x \in \Lambda} \left(\frac{\partial \psi}{\partial x} v_x + \frac{\partial \psi}{\partial y} v_y + \frac{\partial \psi}{\partial t} \right)^2 + w_s \left(\|\nabla v_x\|^2 + \|\nabla v_y\|^2 \right)$$

- Interpretation: it is a combination of the flow-based criterion (we want it to be as much as possible close to 0) and a motion smoothness criterion (the more the motion field is smooth, the more it will tend to 0).
- Trick : in order to avoid over-smoothing of the motion field, Nagel suggest an oriented-smoothness constraint (smoothness along the object boundaries, but not across the boundaries).

Block matching algorithm (BMA)

Block matching algorithm (BMA) I

- One way to impose some smoothness constraints on the estimated motion field is to divide the image domain into non-overlapping small regions, called block
- We assume that the motion within each block can be characterized by a simple parametric model (constant, affine, or bilinear).
- If the block is sufficiently small, then this model can be quite accurate.
- We will use B_m to represent the m-th image block, M the number of blocks, and $\mathcal{M} = \{1, 2, \ldots, M\}$.
- The partition into blocks should satisfy:

$$\bigcup_{m \in \mathcal{M}} B_m = \Lambda,$$

and

$$B_m \bigcap B_n = \emptyset, \ m \neq n$$

Block matching algorithm (BMA) II

- Theoretically, a block can have any polygonal shape, but in practice, the square shape is used almost exclusively.
- In the simplest case, the motion in each block is assumed to be constant (block-wise translational model).
- This kind of algorithm is referred as Block Matching Algorithm (BMA).

The exhaustive search block matching algorithm (EBMA)

The exhaustive search block matching algorithm (EBMA) I

- Given an image block in the reference frame B_m, the motion estimation problem at hand is to determine a matching block B'_m in the current frame so that the error between these two blocks is minimized.
- The displacement vector d_m between the spatial positions of these two blocks (the center or a selected corner) is the MV of this block.
- Under the block-wise translational model,

$$w(x; a) = x + d_m, \; x \in B_m,$$

- Then the error can be written:

$$E(d_m, \forall m \in \mathcal{M}) = \sum_{m \in \mathcal{M}} \sum_{x \in B_m} |\psi_2(x + d_m) - \psi_1(x)|^p$$

The exhaustive search block matching algorithm (EBMA) II

- We can estimate the MV for each block individually:

$$E_m(d_m) = \sum_{x \in B_m} |\psi_2(x + d_m) - \psi_1(x)|^p$$

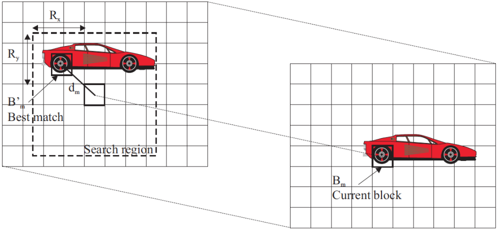


Figure 6.6. The search procedure of the exhaustive block matching algorithm.

- One way to determine the d_m that minimizes the above error is by using Exhaustive block matching algorithm (EBMA) (see the figure above).

The exhaustive search block matching algorithm (EBMA) III

- The EBMA determines the optimal d_m for a given block B_m in the reference frame (error minimization).
- To reduce the computational load, the MAD error ($p = 1$) is often used.
- In 2002, the computation of the EBMA needed VLSI/ASIC hardware architectures since no software were not able to do so in real-time.

References

Indrajit Chakrabarti, Kota Naga Srinivasarao Batta, S. K. C. (2015).
Motion Estimation for Video Coding.
Springer.

Jie Chen, Ut-Va Koc, K. R. L. (2002).
Design of Digital Video Coding Systems.
Signal Processing and Communication Series, Marcel Dekker, Inc.

Yao Wang, Jorn Ostermann, Y.-Q. Z. (2002).
Video Processing and Communications.
Prentice Hall.