Chapter 3
High-level Applications

Variational Methods in Imaging

March 2023

Yvain QUÉAU
GREYC-CNRS
ENSICAEN - Université de Caen Normandie

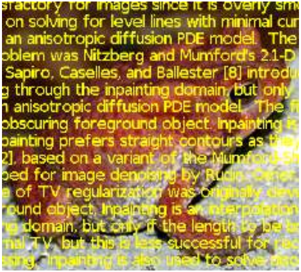
Overview

1. **Poisson Image Editing**
2. **Image Segmentation**
3. **Optical Flow Estimation**

Résumé des épisodes précédents

A principled way to process images, considering them as real valued functions $u : \mathbb{R}^2 \to \mathbb{R}$

Variational image restoration:
$\min_u E(u) = \int \mathcal{L}(x, u(x), \nabla u(x))dx$, with $\mathcal{L}$ an application-dependent Lagrangian function (usually, $\mathcal{L}$ is a pixel-wise data term + regularization term)



Optimality condition = Euler-Lagrange equations

$$\nabla_u E = 0 \iff \frac{\partial \mathcal{L}}{\partial u} - \text{div} \frac{\partial \mathcal{L}}{\partial \nabla u} = 0$$

which can be seen as the stable state of a diffusion equation

Overview

1. **Poisson Image Editing**
2. Image Segmentation
3. Optical Flow Estimation

Image Editing


Naive editing


Poisson editing

How to smoothly transport an image into another?

$\Rightarrow$ Insert source *gradient* into destination, and *integrate*

[Perez et al., Poisson image editing, Siggraph 2003]

Image Editing

The following slides are taken from a seminar talk by Tim Weyrich

Poisson Image Editing

Patric Perez, Michel Gangnet, and Andrew Black
(SIGGRAPH 2003)

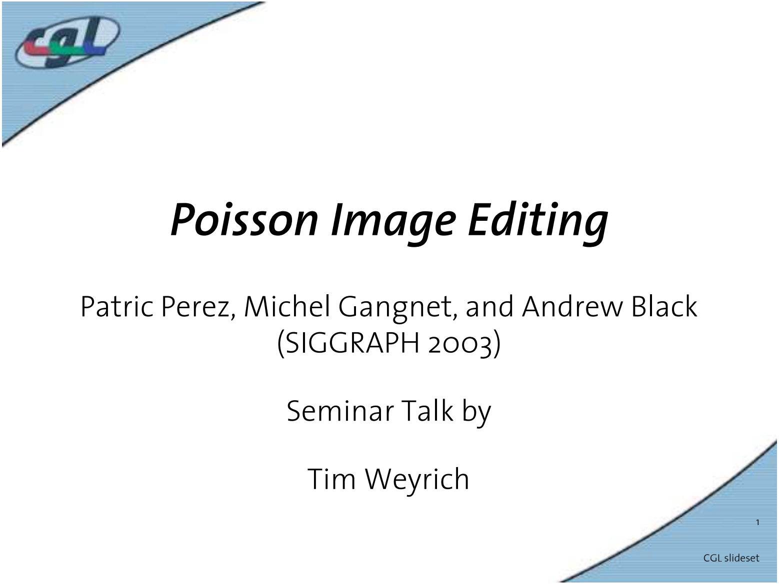
Seminar Talk by

Tim Weyrich

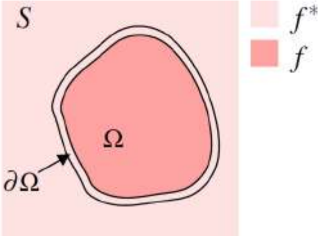
1

CGL slideset

Image Editing



Interpolation Problem



- f^*: known image values
- f : unknown values over region Ω
- Assuming scalar image values

3

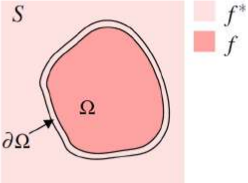
CGL slideset

Image Editing

Simple Interpolation

- Maximize smoothness

$$\min_{f} \int_{\Omega} \|\nabla f\|^2$$



- Boundary constraints

$$f|_{\partial\Omega} = f^*|_{\partial\Omega}$$

4

CGL slideset

Simple Interpolation

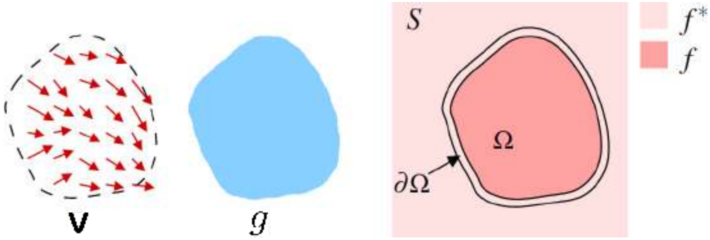
- Solution: *Laplace Equation* with Dirichlet boundary conditions

$$\nabla^2 f = 0, \qquad f|_{\partial\Omega} = f^*|_{\partial\Omega}$$

- Membrane solution
- Unsatisfactory due to over-blurring

Image Editing

Guided Interpolation



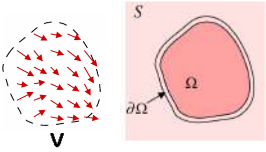
- **v**: guided field
- **v** may be gradient of a function g

Image Editing

Guided Interpolation

- Minimize difference of gradient fields

$$\min_f \int_\Omega \|\nabla f - \mathbf{v}\|^2$$



- Solution: *Poisson Equation* with Dirichlet boundary conditions

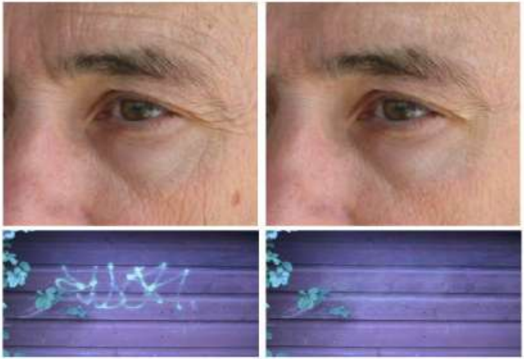
$$\nabla^2 f = \operatorname{div} \mathbf{v}, \qquad f|_{\partial\Omega} = f^*|_{\partial\Omega}$$

7

CGL slideset

Image Editing

Seamless Cloning Results

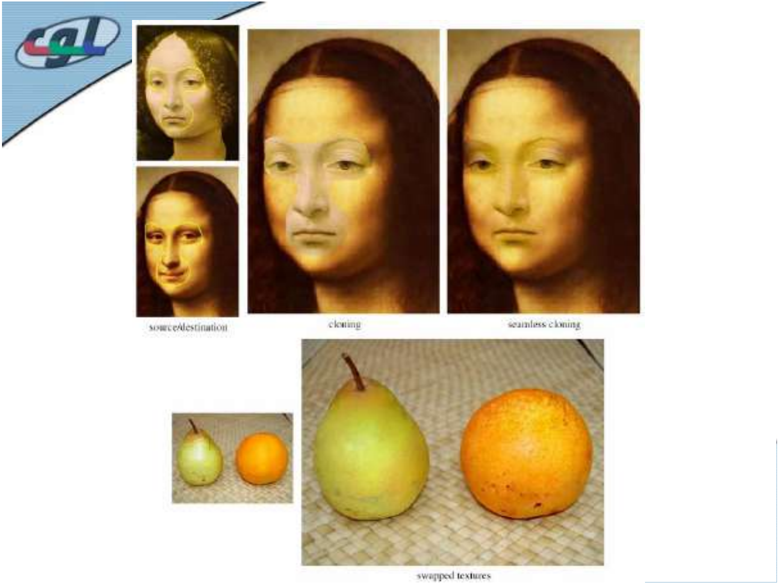


12

CGL slideset

Image Editing

Image Editing



Mixing Gradients

- Two Variants
  - **v** averaged from source and destination gradients $\Rightarrow$ transparency
  - Select stronger one from source and destination gradients:

$$\mathbf{v}(\mathbf{x}) = \begin{cases} \nabla f^*(\mathbf{x}) & \text{if } |\nabla f^*(\mathbf{x})| > |\nabla g(\mathbf{x})| \\ \nabla g(\mathbf{x}) & \text{otherwise} \end{cases}$$

Discretization:

$$v_{pq} = \begin{cases} f_p^* - f_q^* & \text{if } |f_p^* - f_q^*| > |g_p - g_q| \\ g_p - g_q & \text{otherwise} \end{cases}$$

Image Editing

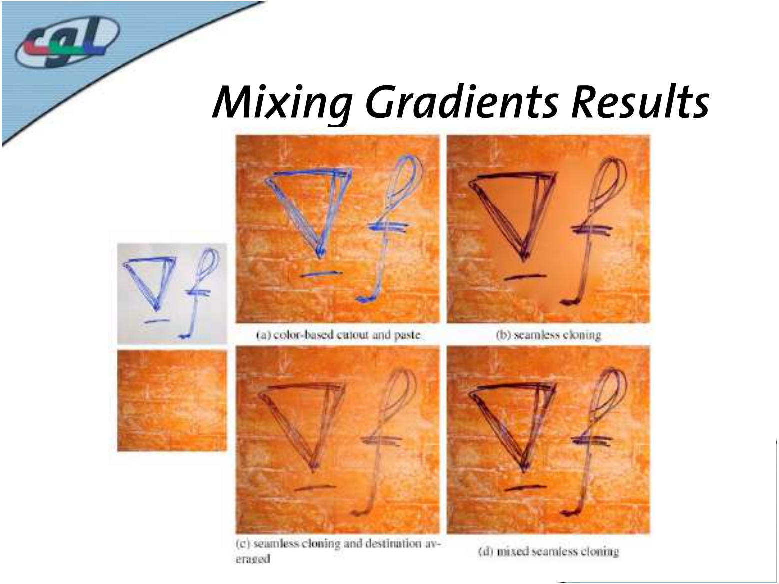


Image Editing

Mixing Gradients Results

source

destination

Image Editing

Mixing Gradients Results

source/destination | seamless cloning | mixed seamless cloning

18

CGL slideset

Image Editing

Texture Flattening

20

CGL slideset

Image Editing

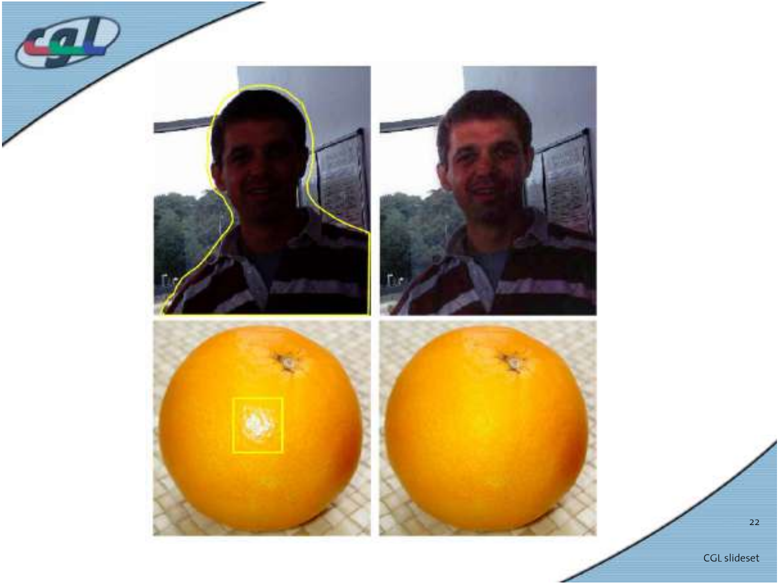
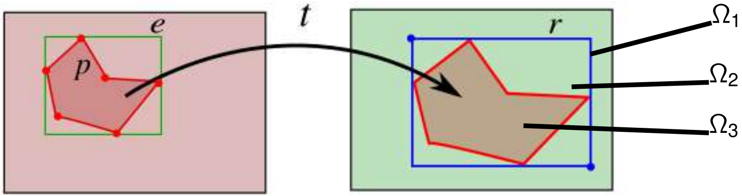


Image Editing

Image Editing

Implementation



Left: source image s to transport. Right: target image c, and rectangular sub-image $r \subset c$. Output u copies c outside the rectangle, and inside we must impose:

- $u = r$ on Ω_1
- $\nabla u \approx \nabla r$ on Ω_2
- $\nabla u \approx \nabla s$ on Ω_3

$$\min_u \int_{\Omega_1} (u(x) - r(x))^2 \, \mathrm{d}x + \int_{\Omega_2} |\nabla u(x) - \nabla r(x)|^2 \, \mathrm{d}x + \int_{\Omega_3} |\nabla u(x) - \nabla s(x)|^2 \, \mathrm{d}x$$

Euler-Lagrange: $Au = b$, with

- $(A, b) = (\mathrm{id}, r)$ on Ω_1
- $(A, b) = (\Delta, \mathrm{div}(v))$ on $\Omega \setminus \Omega_1$, where $v = \begin{cases} \nabla r & \text{on } \Omega_2 \\ \nabla s & \text{on } \Omega_3 \end{cases}$

Overview

Image Segmentation – A Difficult Problem

Original photograph by R. C. James

Image Segmentation

The goal of image segmentation is to partition the image plane into “meaningful” components.

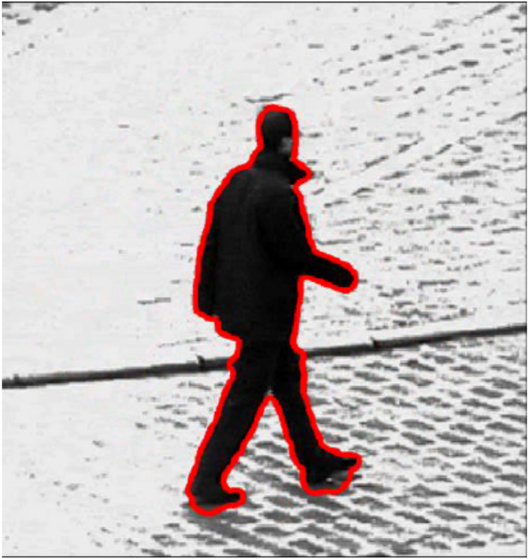
What is meaningful depends on the application. Typically one may want a segmentation where each region corresponds to a separate object or structure in the scene. In this sense, image segmentation is tightly coupled with figure-ground discrimination, image interpretation and semantic analysis.

Image segmentation is the most studied problem in image processing.

There exist many approaches. They typically differ in:

- which local properties are considered in the process (brightness, color, texture, motion,...).
- how the partitioning is computed (examples: region merging, region growing, watershed, graph cuts, level sets, convex relaxation techniques,...).

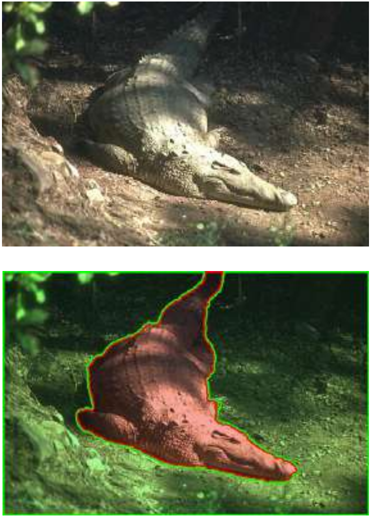
Segmentation: Brightness



Mumford, Shah ’89, Chan, Vese, TIP ’01

Segmentation: Color

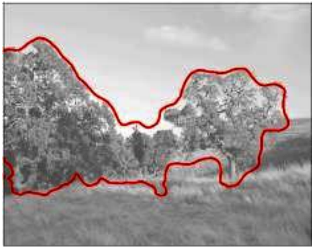
Keuchel et al. PAMI ’03



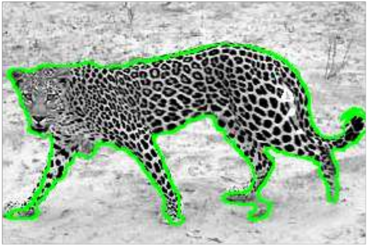
Nieuwenhuis, Cremers PAMI ’13

Segmentation: Texture

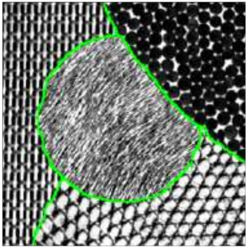
Brox, Weickert, ECCV ’04



Heiler, Schnörr, IJCV ’05

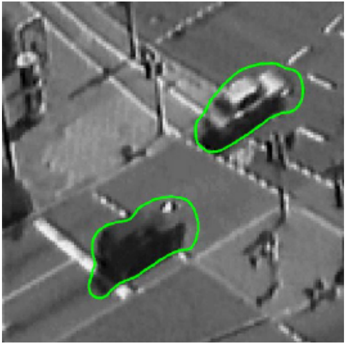


Awate et al., ECCV ’06



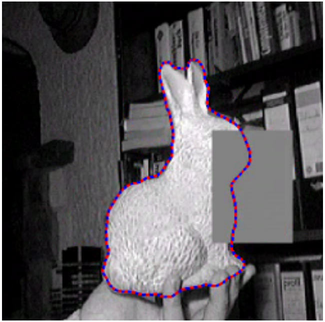
Awate et al., ECCV ’06

Segmentation: Motion



Cremers, Soatto, *Motion Competition*, IJCV 2005.

Segmentation: Brightness and Shape



Cremers et al., ECCV '02 Schoenemann, Cremers PAMI '09

Example of Heuristical Image Segmentation

Many strategies have been proposed to aggregate gradient-based edge information into a coherent segmentation.

For example the following:

1. Identify edges by thresholding the gradient norm,
2. thin out regions ($\rightarrow$ 1-dim. structures),
3. expand contour pieces ($\rightarrow$ close gaps),
4. identify connected components,
5. eliminate smaller regions,
6. thin out regions (again),
7. introduce new boundary pixels ($\rightarrow$ close gaps),
8. eliminate smaller regions.

W. A. Perkins, IEEE Trans. on PAMI 1980

Variational Image Segmentation

Many segmentation algorithms are based on two complementary concepts:

- Detecting discontinuities of the brighness function, or
- Grouping pixels of similar brightness (color, texture, etc.)

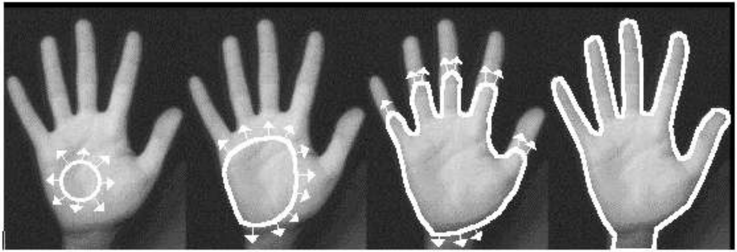
Most of historical approaches (Perkins, region growing and merging, K-means, superpixels, etc.) lack a clear optimization criterion: Edge regions are heuristically fused to connected lines (Perkins, Canny), or pixels are iteratively merged to regions (region merging, region growing).

Toward the end of the 1980s, the first variational formulations for image segmentation emerged, in particular:

- the Snakes (Kass, Witkin, Terzopoulos, “Snakes: Active contour models”, *Int. J. of Comp. Vision ’88*),
- the Mumford-Shah Functional (Mumford, Shah, “Optimal approximations by piecewise smooth functions and associated variational problems”, *J. Appl. Math. ’89*).

Snakes

Snakes (active contours) iteratively move an initial curve towards the edges in the images:



Formally, we are seeking a parametric curve $C : [0,1] \to \mathbb{R}^2$ which:

- matches "high" image gradients: $\min_u \int_{0,1} -|\nabla I(C(s))|^2 \mathrm{d}s \dots$
- is smooth:

$$+ \cdots \int_0^1 \left\{ \frac{\alpha}{2} \left| C_s(s) \right|^2 + \frac{\beta}{2} \left| C_{ss}(s) \right|^2 \right\} \mathrm{d}s$$

Snakes

In 1988, Kass, Witkin and Terzopoulos proposed to minimize the following functional:

$$E(C) = E_{ext}(C) + E_{int}(C)$$

with an external energy

$$E_{ext}(C) = -\int_0^1 |\nabla I(C(s))|^2 \, ds$$

and an internal energy

$$E_{int}(C) = \int_0^1 \left\{ \frac{\alpha}{2} |C_s(s)|^2 + \frac{\beta}{2} |C_{ss}(s)|^2 \right\} ds$$

Here, $I : \Omega \subset \mathbb{R}^2 \to \mathbb{R}$ denotes the input image, and $C : [0,1] \to \Omega$ denotes a parametric curve. C_s and C_{ss} denote the first and second derivative of the curve C with respect to its parameter s.

Snakes: External Energy

The external energy:

$$E_{ext}(C) = -\int_0^1 |\nabla I(C)|^2 \, ds$$

measures for a given curve C how well it coincides with the maxima of the brightness gradient $|\nabla I|$.

Thus rather than first searching for these maxima and then grouping them to a curve one defines a cost function which measures the "edge strength" along any conceivable curve.

Subsequently, the optimal curve $\hat{C}$ is determined by minimizing the total energy:

$$\hat{C} = \arg\min_C \; E(C)$$

Gradient descent on this energy induces an evolution of the curve toward locations of large image gradient.

Snakes: Internal Energy

The internal energy is a regularizer which induces some smoothness on the computed curves:

$$E_{int}(C) = \int_0^1 \left\{ \frac{\alpha}{2} \left| C_s(s) \right|^2 + \frac{\beta}{2} \left| C_{ss}(s) \right|^2 \right\} ds$$

It consists of two components, weighted by parameters $\alpha \geq 0$ and $\beta \geq 0$, which penalize the elastic length and the stiffness of the curve.

Minimizing the total energy

$$E(C) = E_{ext}(C) + E_{int}(C)$$

leads to curves which are short and stiff while passing through locations of large gradient.

Snakes: Gradient Descent

The Snakes energy

$$E(C) = -\int_0^1 |\nabla I(C)|^2 \, ds \; + \; \int_0^1 \left\{ \frac{\alpha}{2} \left| C_s(s) \right|^2 + \frac{\beta}{2} \left| C_{ss}(s) \right|^2 \right\} ds$$

is of the canonical form

$$E(C) = \int \mathcal{L}(C, C_s, C_{ss}) ds$$

The corresponding Euler-Lagrange equation is given by:

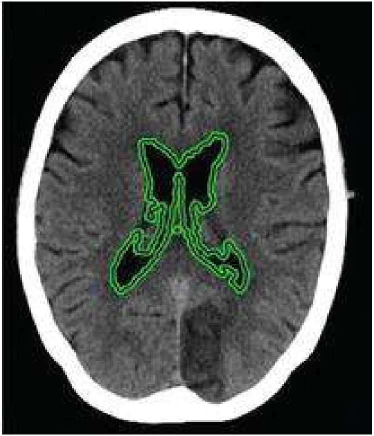
$$\frac{dE}{dC} = \frac{\partial \mathcal{L}}{\partial C} - \frac{d}{ds} \frac{\partial \mathcal{L}}{\partial C_s} + \frac{d^2}{ds^2} \frac{\partial \mathcal{L}}{\partial C_{ss}} = -\nabla |\nabla I(C)|^2 - \alpha C_{ss} + \beta C_{ssss} = 0.$$

Consequently, the gradient descent equation reads:

$$\frac{\partial C(s,t)}{dt} = -\frac{dE(C)}{dC} = \nabla |\nabla I(C)|^2 + \alpha C_{ss} - \beta C_{ssss}$$

Snakes

The optimization problem is nonconvex, thus needs a good initialization. Snakes are thus particularly useful when a human expert can easily draw a rough segmentation, which is then automatically refined. This happens a lot, e.g., in medical imaging:



The Mumford-Shah Approach

In 1989, Mumford and Shah proposed to compute a piecewise smooth approximation u of the input image $I : \Omega \subset \mathbb{R}^2 \to \mathbb{R}$ by minimizing the functional:

$$E(u, C) = \int_{\Omega} \left(I(x) - u(x)\right)^2 dx \; + \; \lambda \int_{\Omega \setminus C} |\nabla u(x)|^2 \, dx \; + \; \nu |C|,$$

jointly with respect to an approximation $u : \Omega \to \mathbb{R}$ and a one-dimensional discontinuity set $C \subset \Omega$. The three terms have the following meaning:

- The data term assures that u is a faithful approximation of the input I.
- The smoothness term, weighted by $\lambda > 0$, assures that u is smooth everywhere except for the discontinuity set.
- A further regularizer, weighted by $\nu > 0$, assures that this discontinuity set has minimal length $|C|$.

The Piecewise Constant Mumford-Shah

For increasing values of the weight λ, the approximation u is forced to be smoother and smoother outside of C. In the limit $\lambda \to \infty$ we obtain a piecewise constant approximation of the image I:

$$E(u, C) = \int_{\Omega} \big(I(x) - u(x)\big)^2 \, dx \; + \; \nu |C|,$$

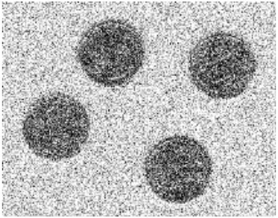
where $u(x)$ is constant in each of the regions separated by the boundary C. If we denote these regions by $\{\Omega_1, \ldots, \Omega_n\}$ and the constants by u_i, this can be rewritten as:

$$E(\{u_1, \ldots, u_n\}, C) = \sum_{i=1}^{n} \int_{\Omega_i} \big(I(x) - u_i\big)^2 \, dx \; + \; \nu |C|,$$

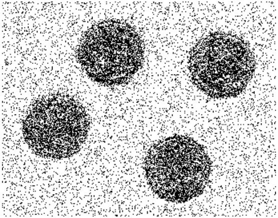
For the case of two regions, a spatially discrete formulation of this energy is known as the Ising model (Lenz 1920, Ising 1925, Heisenberg 1928). It models the phenomenon of ferromagnetism and is among the most studied models in statistical physics.

Solution via Graph Cuts

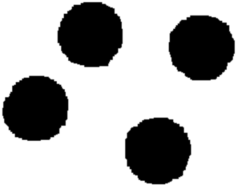
Original f



noisy: $I = f + \eta$



thresholding: $I > \theta$



$\arg\min E(u)$

Minimization of the discrete two-region model using graph cuts

Numerical Solving

The paper of Mumford and Shah is focused on aspects of existence and uniqueness of solutions and the study of properties of solutions. It does not propose a numerical implementation for finding minimizers.

Unfortunately, the Mumford-Shah functional in its original formulation is not in a canonical form, since the variable of interest (the boundary C) appears in the integrand.

There exist a number of ways to solve the Mumford-Shah problem, for example:

- *Ambrosio and Tortorelli, "Approximation of functionals depending on jumps by elliptic functionals via Γ-convergence", Comm. Pure Appl. Math. 90:* an approximation of the piecewise smooth model using quadratic functionals.
- *Cremers et al., "Diffusion snakes: Introducing statistical shape knowledge into the Mumford-Shah functional", IJCV 2002:* Implementation of the piecewise smooth and piecewise constant models using closed parametric spline curves (hybrid of the Mumford-Shah and the Snakes).

Ambrosio-Tortorelli Approximation

Ambrosio and Tortorelli showed in 1990 that the piecewise-smooth Mumford-Shah model

$$E(u,C) = \iint\limits_{\Omega} \left(I(x) - u(x)\right)^2 dx + \lambda \iint\limits_{\Omega \setminus C} |\nabla u(x)|^2 \, dx + \nu \int_0^1 |C'(s)| \, ds$$

can be approximated well by

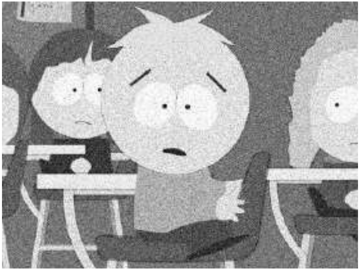
$$E_\epsilon(u,w) = \iint\limits_{\Omega} \left(I(x) - u(x)\right)^2 dx + \lambda \iint\limits_{\Omega} w(x)^2 |\nabla u(x)|^2 \, dx$$
$$+ \nu \iint\limits_{\Omega} \epsilon |\nabla w(x)|^2 + \frac{1}{4\epsilon} \left[w(x) - 1\right]^2 dx$$

in the sense that $E_\epsilon \xrightarrow{\Gamma} E$ when $\epsilon \to 0$, with $w : \Omega \to \mathbb{R}$ converging towards the indicator function of the the "smooth" part $\Omega \setminus C$.

Alternating optimization then boils down to a sequence of simple linear least-squares problems!

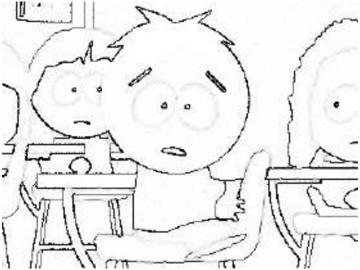
Ambrosio-Tortorelli Approximation

Results with $\lambda = 10$, $\nu = 10^{-7}$ and $\epsilon = 10^{-6}$:



I

u



w

C

Remarks on variational image segmentation

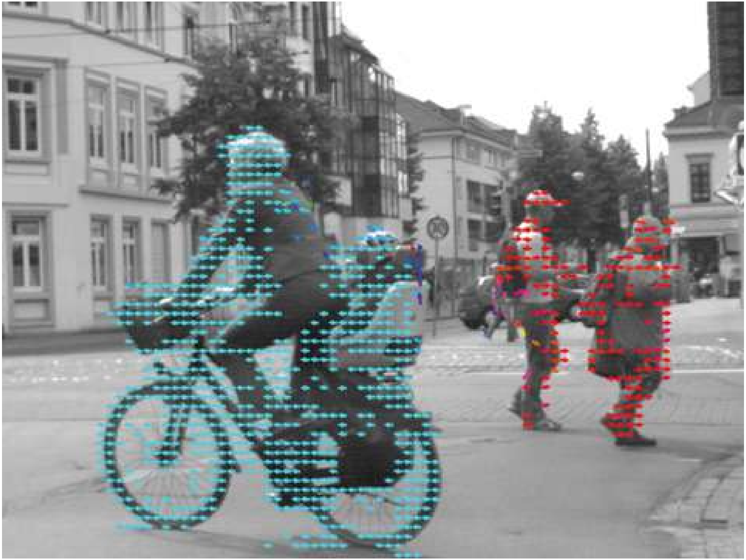
Nowadays, black-box deep learning frameworks are often more efficient, and they represent the state-of-the-art for (semantic) image segmentation

See e.g., Detectron2 and variants

Overview

Motion Estimation

Goal: estimate the apparent motion from image sequences



Motion Estimation

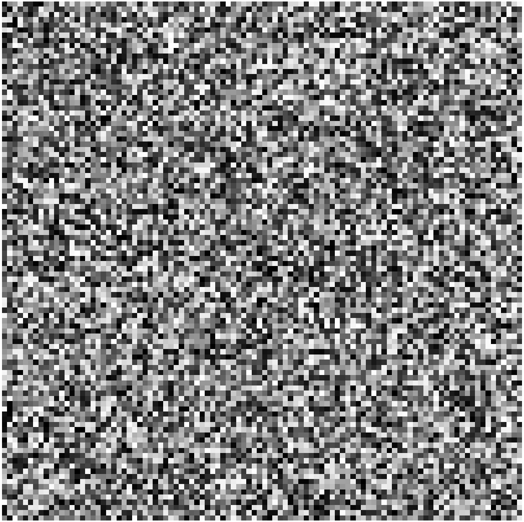
- The estimation of motion fields from image sequences is among the central problems in computer vision.
- With increasing amount of image sequence data – more and more video-capable cameras, higher frame rates, videos on the internet – image sequence analysis is becoming increasingly important.
- Compared to still images, video contains an enormous amount of information about the world surrounding us in the sense that structures can often be distinguished based on their temporal evolution.
- Some applications of motion estimation are already integrated in camera software – panorama generation from several images, video stabilization to remove camera shake, etc.
- Mathematically the problem of motion estimation from images is an ill-posed problem, which means that the question is not sufficiently specified to assure a unique solution.

The Correspondence Problem

Algorithmically, the key challenge in motion estimation is to solve the correspondence problem. Given two images, determine for each point in either image the corresponding partner in the other image. Many computer vision problems are inherently such correspondence problems:

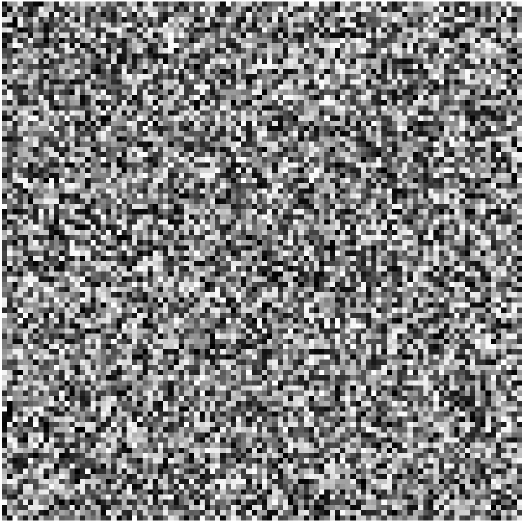
- Disparity estimation from stereo images: Determine a one-dimensional displacement for each pixel to determine the corresponding pixel in the other image. This displacement is inversely proportional to the depth of the respective point.
- Multimodal registration: Given two medical images of an organ acquired with different sensors – for example CT (Computer Tomography) and MRI (Magnetic Resonance Imaging), or CT and PET (Positron-Emission Tomography) – compute an optimal alignment of these images.
- Shape Matching: Given two object shapes (contours in 2D or surfaces in 3D), determine a correspondence between pairs of points from either shape.

Motion and Grouping



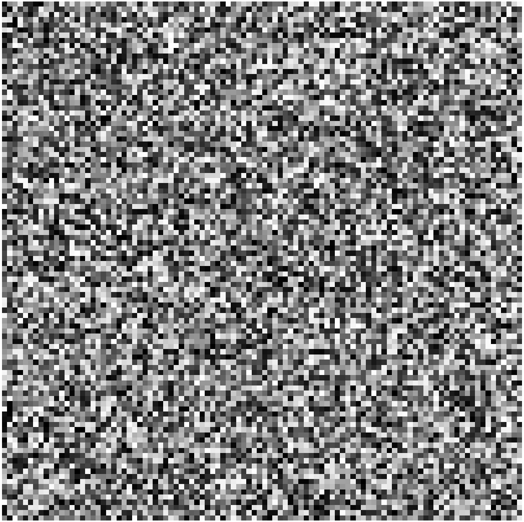
Moving regions of random brightness values

Motion and Grouping



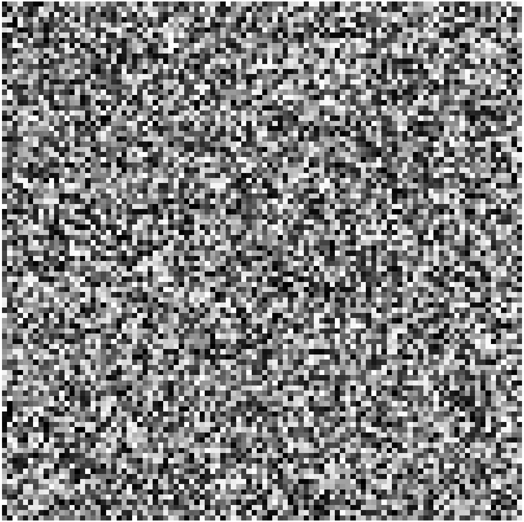
Moving regions of random brightness values

Motion and Grouping



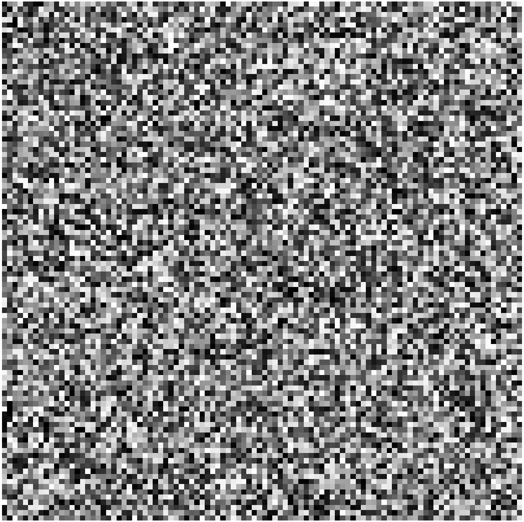
Moving regions of random brightness values

Motion and Grouping



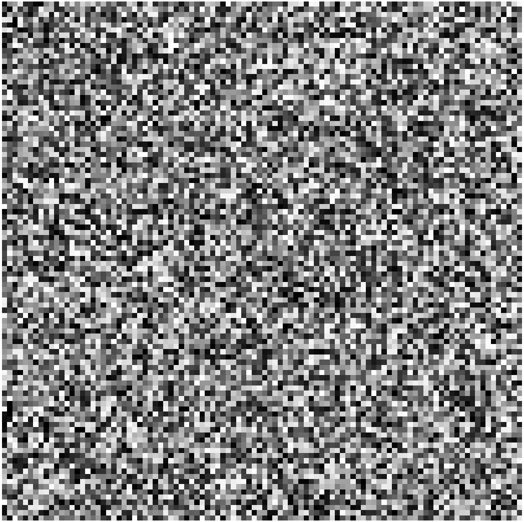
Moving regions of random brightness values

Motion and Grouping



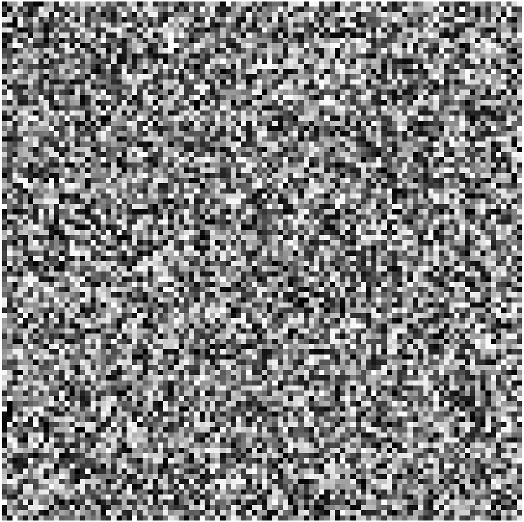
Moving regions of random brightness values

Motion and Grouping



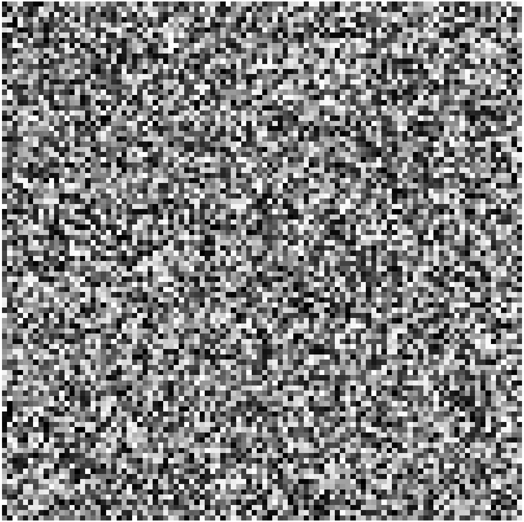
Moving regions of random brightness values

Motion and Grouping



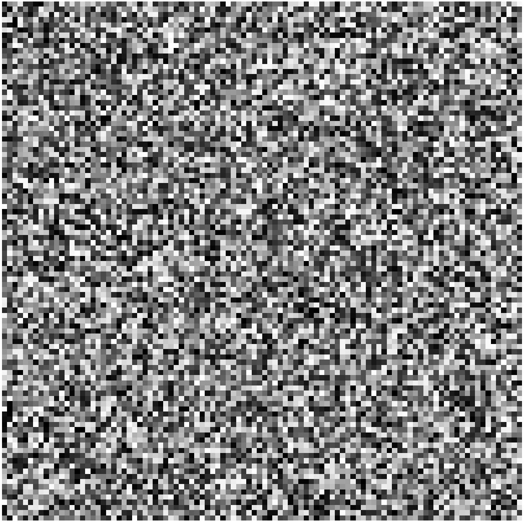
Moving regions of random brightness values

Motion and Grouping



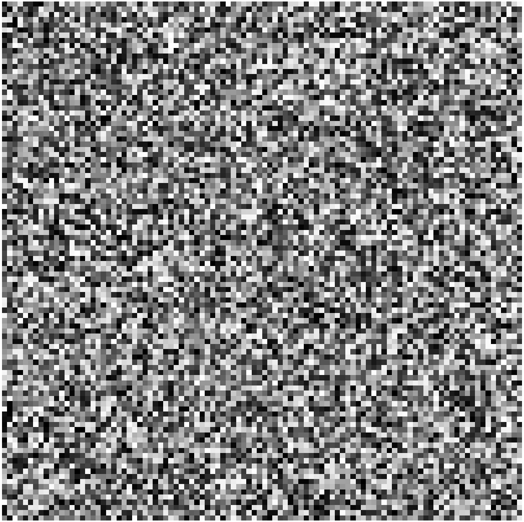
Moving regions of random brightness values

Motion and Grouping



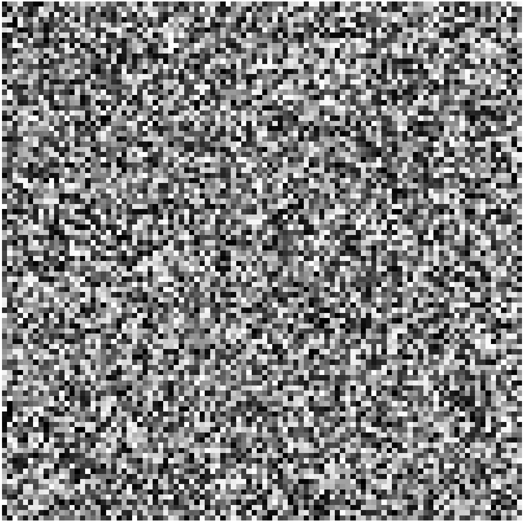
Moving regions of random brightness values

Motion and Grouping



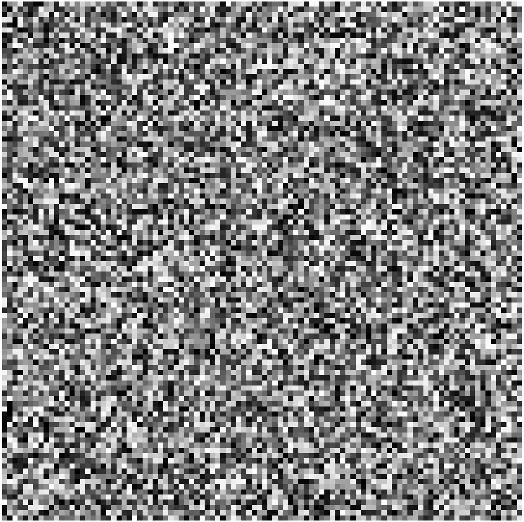
Moving regions of random brightness values

Motion and Grouping



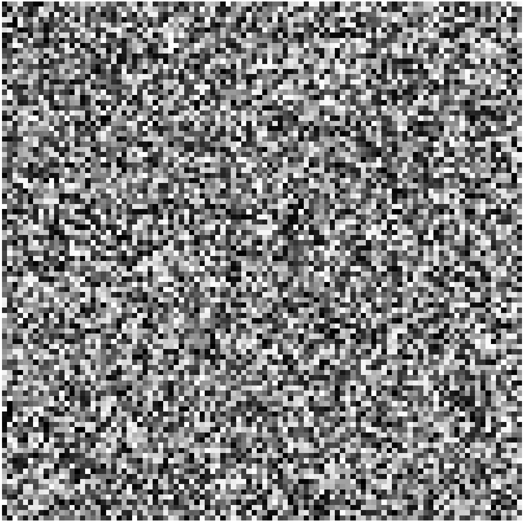
Moving regions of random brightness values

Motion and Grouping



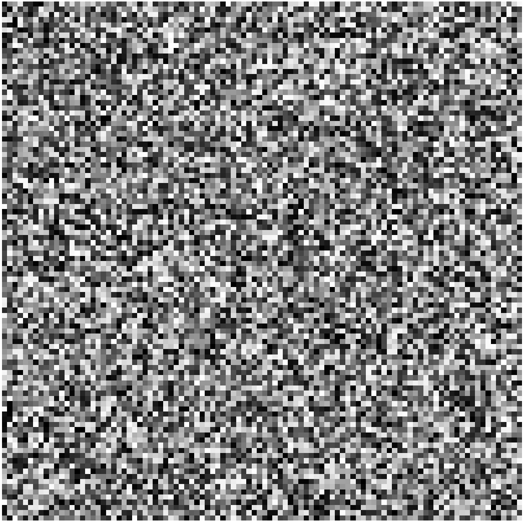
Moving regions of random brightness values

Motion and Grouping



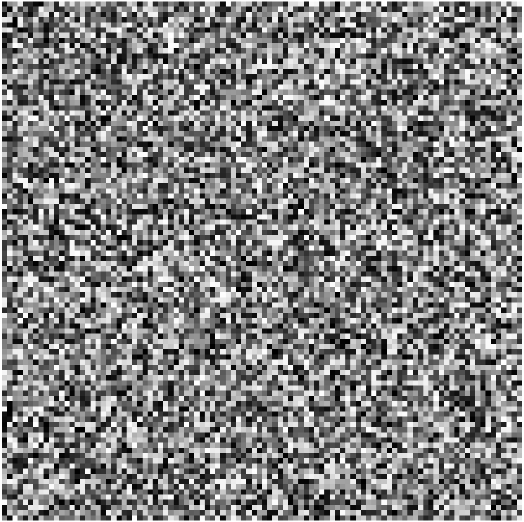
Moving regions of random brightness values

Motion and Grouping



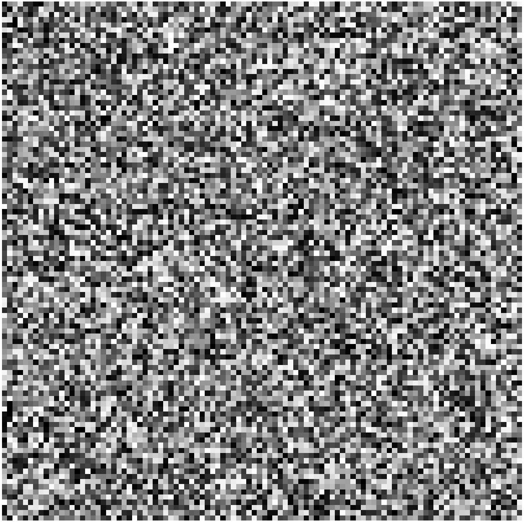
Moving regions of random brightness values

Motion and Grouping



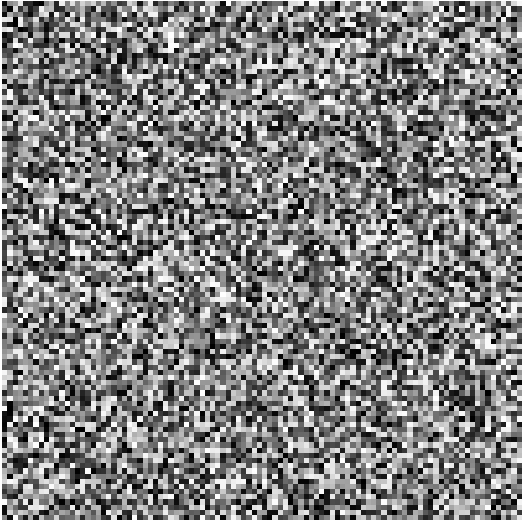
Moving regions of random brightness values

Motion and Grouping



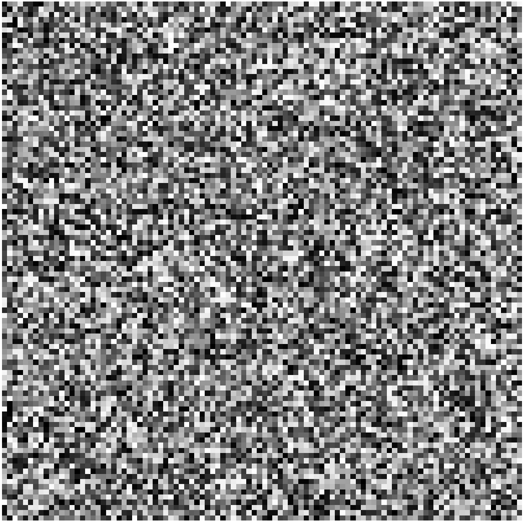
Moving regions of random brightness values

Motion and Grouping



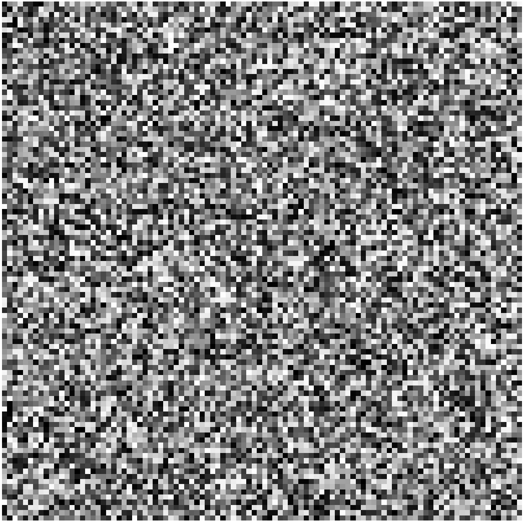
Moving regions of random brightness values

Motion and Grouping



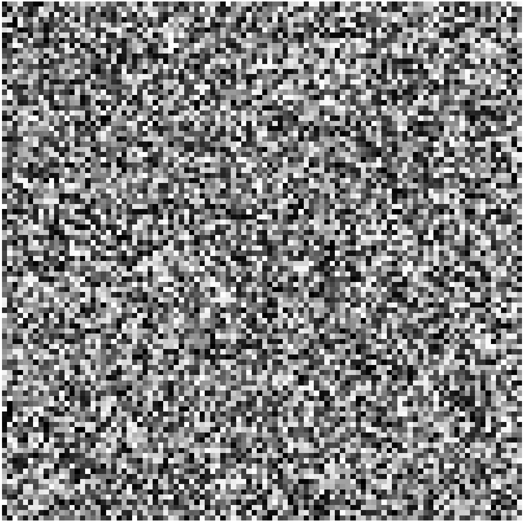
Moving regions of random brightness values

Motion and Grouping



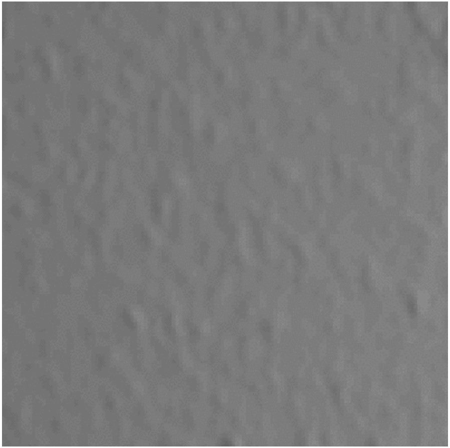
Moving regions of random brightness values

Motion and Grouping



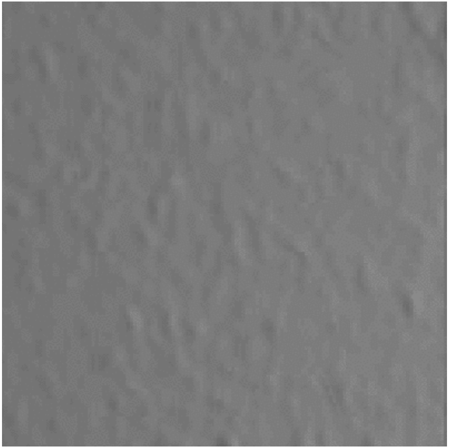
Moving regions of random brightness values

Motion and Grouping



Moving wallpaper regions

Motion and Grouping



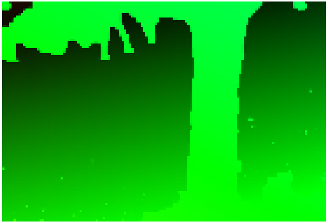
Moving wallpaper regions

Motion and 3D Structure

Several images of a static scene filmed by a moving camera.
Foreground objects move faster than background objects.

Motion and 3D Structure



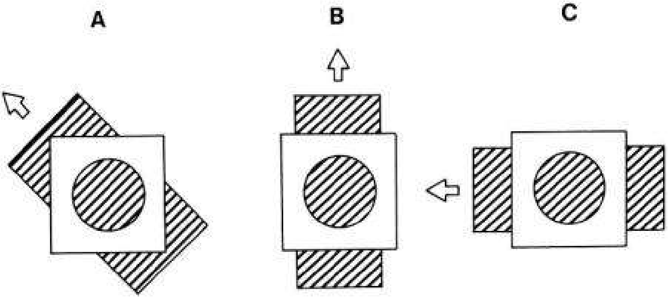
Schoenemann & Cremers,

Near Real-time Motion Segmentation, DAGM 2006.

Applications of Motion Estimation

- Grouping and Segmentation: Motion information allows to identify image regions as objects. This can also be done if semi-transparent structures overlap at a given location.
- Tracking: Using motion information, objects can be tracked in a video sequence.
- Depth estimation: Motion information allows to infer the distance of respective objects from the camera. In principle, one can recover the 3D geometry of the world from an image sequence.
- Time-to-Impact: In the context of driver assistance, motion information allows to make predictions when an obstacle will be hit. As a consequence, one can initiate evasion maneuvers or breaking.
- Video compression: Motion information allows to efficiently compress videos (MPEG encoding).

The Aperture Problem



In these three cases, the same motion will be observed (Blake and Sekuler, Perception, 2005): motion estimation is an ill-posed problem

The Brightness Constancy Assumption

Given an image sequence $I : \Omega \times [0, T] \to \mathbb{R}$, on the image plane $\Omega \subset \mathbb{R}^2$ and the time interval $[0, T]$, we wish to compute a motion field $v : \Omega \times [0, T] \to \mathbb{R}^2$, which assigns to each point $x \in \Omega$ at each time $t \in [0, T]$ a motion vector $v(x, t)$.

The classical assumption in motion estimation states that the brightness of a moving point remains constant over time:

$$I(x, t) = I(x + v(t), t + \delta t) \quad \forall t \in [0, T]$$

which we can linearize as follows for small displacements:

$$\boxed{\nabla I^\top v + \partial_t I = 0}$$

This equation is referred to as the differential brigthness constancy constraint or the optic flow constraint.

Additional Assumptions: Spatial Regularity

The optic flow constraint is necessary but not sufficient to uniquely determine a motion field. It only specifies the normal component of the velocity field.

In order to eliminate the additional degree of freedom, we therefore need to make additional assumptions.

Two pioneering approaches:

- *Lucas and Kanade 1981*: Assume that the velocity in an entire window around each point is constant. If the window is “sufficiently” large one obtains a unique solution. (over 12800 citations in Jan 2017).
- *Horn and Schunck 1981*: A variational approach to motion estimation based on the assumption of spatial smoothness of the the flow field $v(x, t)$. Extensions to temporal smoothness are straight-forward. (over 13100 citations in Jan 2017). This paper is often considered the first variational method in computer vision.

Horn and Schunck

The approach of *Horn and Schunck (1981)* is considered the first variational approach in computer vision (cf. Snakes: 1988, Mumford-Shah: 1989). In addition to the optic flow constraint for each point, one assumes spatial smoothness of the velocity field $v(x)$:

$$E(v) = \int_\Omega \left(\nabla I^\top v + I_t\right)^2 dx\,dy \; + \; \lambda \int_\Omega |\nabla v(x)|^2 \, dx\,dy.$$

Increasing smoothness of the flow field can be imposed by increasing the weight $\lambda > 0$ of the regularizer.:

$$|\nabla v(x)|^2 \equiv |\nabla v_1(x)|^2 + |\nabla v_2(x)|^2$$

Euler-Lagrange Equations

Let $v = (v_1, v_2)$ be the flow field with components v_1 and v_2 in x- and y-direction. The minimizer of the Horn and Schunck functional

$$E(v) = \frac{1}{2} \int_\Omega \left(I_x v_1 + I_y v_2 + I_t\right)^2 + \lambda \left(|\nabla v_1(x)|^2 + |\nabla v_2(x)|^2\right) \, dx \, dy.$$

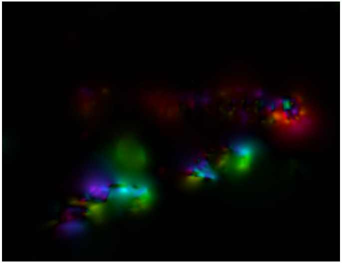
must fulfill the Euler-Lagrange equations:

$$\begin{cases} \dfrac{\partial E}{\partial v_1} = I_x \left(I_x v_1 + I_y v_2 + I_t\right) - \lambda \Delta v_1 = 0 \\ \dfrac{\partial E}{\partial v_2} = I_y \left(I_x v_1 + I_y v_2 + I_t\right) - \lambda \Delta v_2 = 0 \end{cases}$$

These equations are linear and can be solved with a Gauss-Seidel or Jacobi solver. The regularizer imposes smoothness of the computed flow field. It generates a fill in effect: Components of the velocity field which are not affected by the optic flow constraint are simply adopted from neighboring regions.

Horn/Schunck: Examples

One of two images



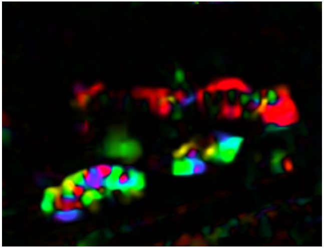
Color-coded flow field

Color encodes direction and magnitude

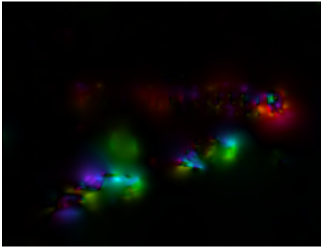
Author: Thomas Brox

Discontinuity-preserving extension

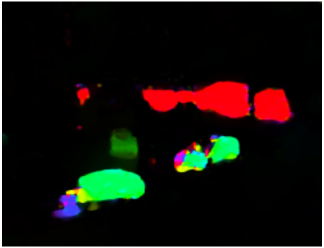
first of two images



Lucas & Kanade '81



Horn & Schunck '81



Brox et al. '04

Author: Thomas Brox