An Image Curvature Microscope

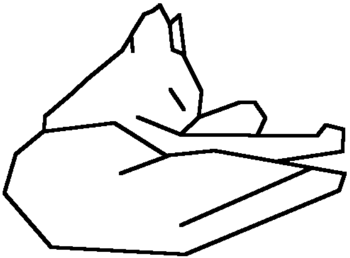
Jean-Michel MOREL
Joint work with Adina CIOMAGA and Pascal MONASSE

Centre de Mathématiques et de Leurs Applications,
Ecole Normale Supérieure de Cachan

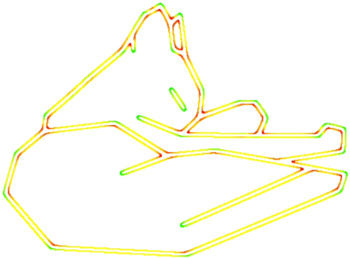
Séminaire Jean Serra - 70 ans
April 2, 2010

Overview

Role of Curvature in Visual Perception



(a). Attneave's cat



(b). Curvature map.

Image interpolations

Digital images can be modeled as

- piecewise constant functions. (*block interpolation*)

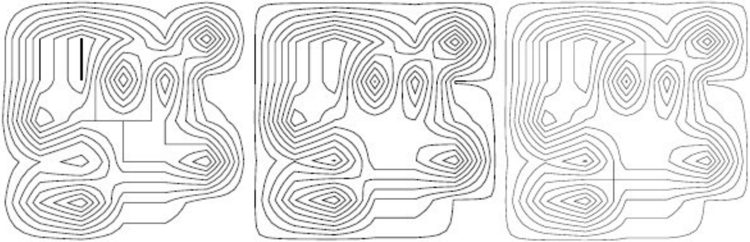
$$u = u_d * \chi_{[-\frac{1}{2},\frac{1}{2}]}$$

- continuous functions, taking the given values at the centers of the pixels and being affine on the corresponding edges (*bilinear interpolation*)

$$u = u_d * \chi_{[-1,1]}(1 - |\cdot|)$$

- higher order spline interpolations.

Bilinear tree of level lines



Bilinear interpolation in a dual pixel can locally be written as

$$u(x, y) = axy + bx + xy + d$$

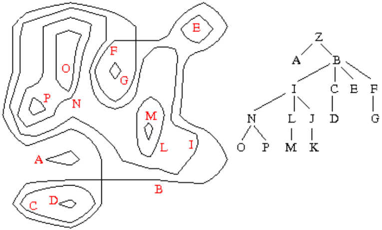
where the parameters a, b, c, d are given by the values taken at four adjacent pixels. Level lines then are then concatenations of pieces of hyperbole and straight lines.

Bilinear tree of level lines

- One can decompose an image into its level lines at quantized levels.

$$\mathcal{T} = \{\Sigma^{\lambda,i}\}_{\lambda\in\Lambda, i\in F_\lambda};$$

- The set is ordered in a tree structure, induced by the geometrical inclusion.



Algorithm (Monasse Guichard '98)

A fast algorithm, the Fast Level Set Transform (FLST) performs the decomposition of an image into a tree of shapes (subsequently, in a tree of level lines).

Image reconstruction

Algorithm (C., Monasse, Morel, '09)

Construct an image from its topographic map

- *walk the tree in pre-order (parent before children)*
- *fill the interior of the current level line* $\Sigma = \{P_k(x_k, y_k)\}_{1 \leq k \leq N}$ *with its level* λ*:*
  - *find intersections of the boundary with all horizontal lines of equation* $y = i$ *and write the abscissas in an ordered set*
  - *a pixel* (j, i) *is inside the polygon if and only if* j *is within an interval* $[x^i_{2k+1}, x^i_{2k+2}]$.

Image Reconstruction

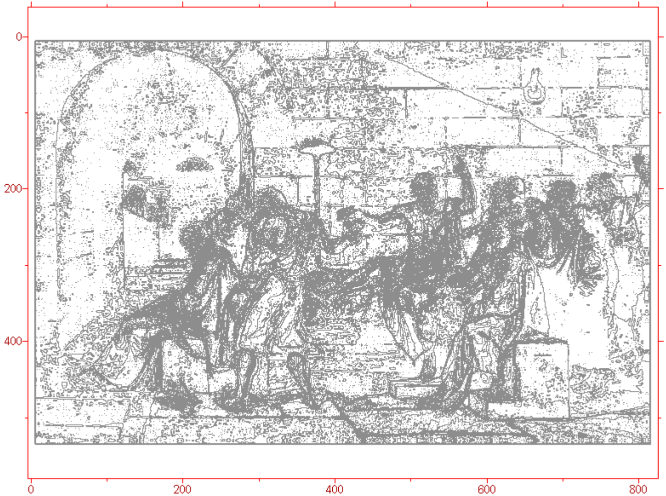


Image Reconstruction

Curvatures in digital images

The scalar curvature of a C^2 image at a nonsingular point $\mathbf{x_0}$ is defined by

$$\mathrm{curv}(u)(\mathbf{x_0}) = \frac{u_{xx}u_y^2 - 2u_{xy}u_xu_y + u_{yy}u_x^2}{(u_x^2 + u_y^2)^{3/2}}(\mathbf{x_0}). \qquad (1)$$

This scalar curvature at $\mathbf{x_0}$ is linked to the vectorial curvature $\kappa(\mathbf{x_0})$ of the level line passing by $\mathbf{x_0}$ via

$$\kappa(\mathbf{x_0}) = -\mathrm{curv}(u)(\mathbf{x_0}).\frac{Du}{|Du|}(\mathbf{x_0}). \qquad (2)$$

Thus, curvatures in digital images can be computed in two quite different ways.

Multiscale curvature

A previous smoothing is necessary, which introduces a new parameter, the smoothing *scale*. Hence the notion of *curvature scale space* which will be associated with curve or image evolutions.

Problem

Smoothing algorithms in the computer vision literature deal with either

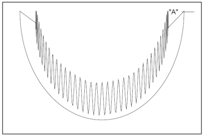
- *level lines: curve/affine shortenings*
- *level sets: threshold dynamics*
- *or with the image itself: FDSs and stack filters*

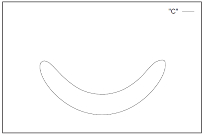
Curvature Flows

- Data: closed curve Γ_0
- Perform curvature driven flows

$$\Sigma_0 \mapsto \Sigma_t$$

$$\frac{\partial x}{\partial t} = |k|^{\sigma-1} k \overrightarrow{n}$$





Questions

- *well posedness; existence and regularity of solutions ;*
- *numerical approximation schemes;*
- *preserve morphological properties.*

Local heat equation ?

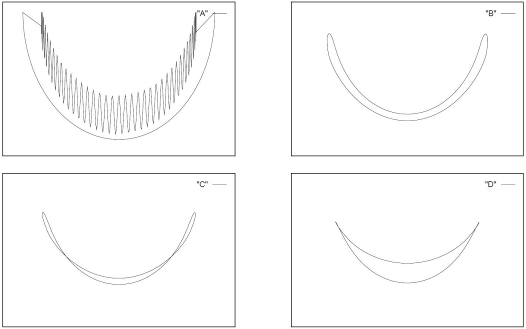


Figure: Curve evolution by the heat equation.The evolving curve can, however, develop self-crossings (as in C) or singularities (as in D).

Dynamic curve evolution: nonlocal heat equation

Algorithm (Mackworth Mockhtarian '92)

- *Convolve the curve x_n, parameterized by its length parameter $s_n \in [0, L_n]$, with a Gaussian G_h, where h is small.*

$$x_{n+1}(s_n) = G_h * x_n(s_n).$$

- *Reparametrize x_{n+1} by its length parameter $s_{n+1} \in [0, L_{n+1}]$.*

Dynamic curve evolution: nonlocal heat equation!

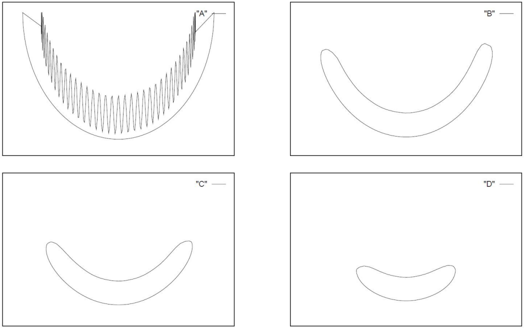


Figure: Curve evolution by the renormalized heat equation. The evolved curve is smooth for all times, eventually becomes convex and shrinks to a point.

Level set methods

Algorithm (Stack filter and threshold dynamics)

- *Decompose* u_0 *in its upper level sets and consider the characteristic function* $\chi_\lambda(\cdot)$ *of each upper level set* $X_\lambda u_0$*;*

$$u_0 \mapsto \{X_\lambda u_0\}_\lambda.$$

- *Solve mean curvature motion for* $\chi_\lambda(\cdot)$ *until the scale* t*.*

$$\psi_\lambda(t,\cdot) = FDS(\chi(\cdot))(t).$$

- *Get back the image by thresholding*

$$u(t,x) = \lambda, \forall x \text{ s.t. } \psi_\lambda(x) \geq 1/2.$$

Level set methods

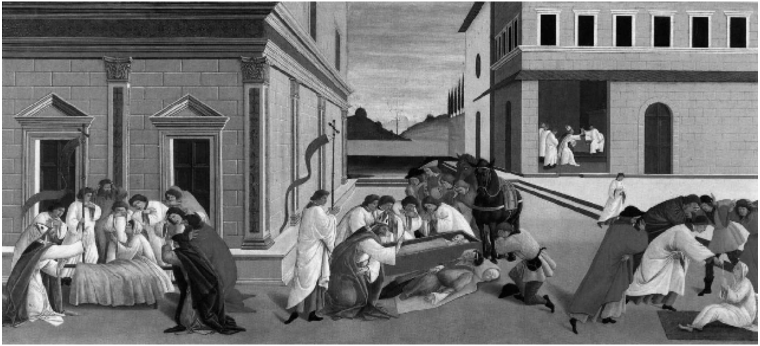


Figure: Level set method (BMO algorithm) for mean curvature evolution, at renormalized scale $l = 2$.

Level set methods

Figure: Level set method (BMO algorithm) for mean curvature evolution, at renormalized scale $l = 2$.

Level Lines Shortening

Subpixel algorithm based on the topological structure of the level lines

$$
\begin{array}{ccc}
u_0(\cdot) & \xrightarrow{\textit{level lines extraction}} & \{\Sigma_0^{\lambda,i}\}_{\lambda,i} \\
\textit{MCM/ACM} \downarrow & & \downarrow \textit{CS/AS} \\
u(\cdot, t) & \xleftarrow{\textit{reconstruction}} & \{\Sigma_t^{\lambda,i}\}_{\lambda,i}
\end{array}
$$

The scheme is monotonous and therefore ensures level lines order preserving.

Level lines Shortening

Algorithm (C., Monasse, Morel, '10)

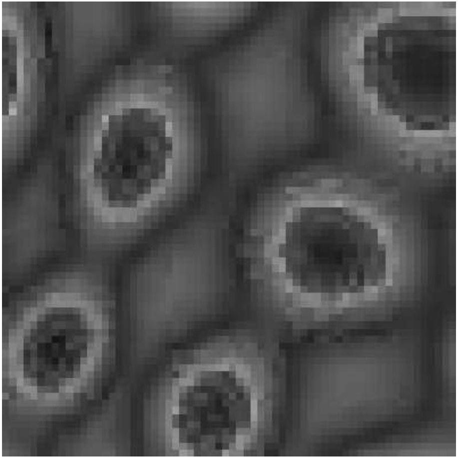
Perform the LLS evolution of u_0 at scale t:

- *Extract the tree of level lines* $\{\Sigma_0^{\lambda,i}\}_{\{i \in F_\lambda, \lambda\}}$;
- *Smooth each level line separately*

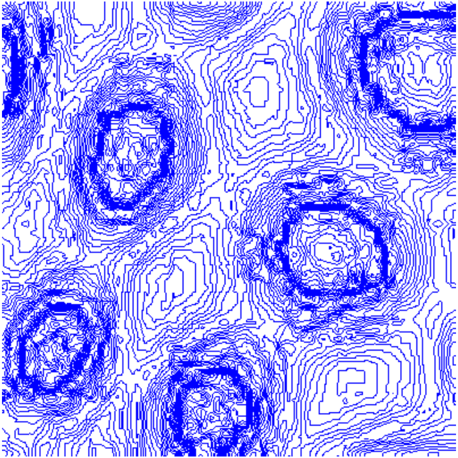
$$\Sigma_t^{\lambda,i} = \textit{ Curve Shortening Flow } (\Sigma_0^{\lambda,i})$$

- *Reconstruct the image by filling the interior laminas bounded by each level line* $\Sigma_t^{\lambda,i}$;

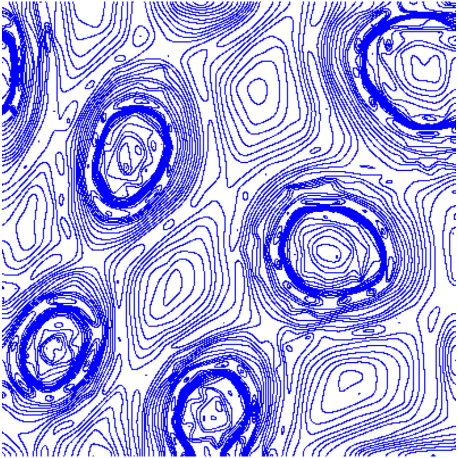
Level Lines Shortening



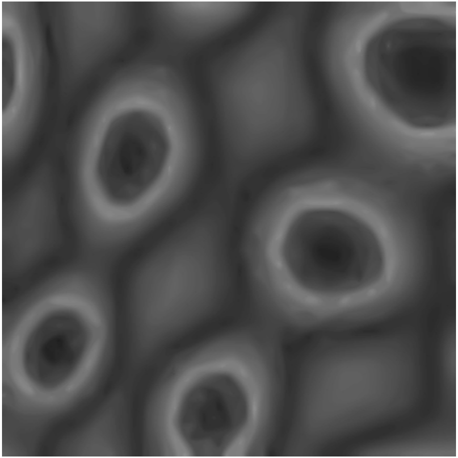
Level Lines Shortening



Level Lines Shortening



Level Lines Shortening



Level lines Shortening

Theorem

(C., Morel '10) Let $u_0 \in Lip(\Omega)$. Then $u(x,t) : \Omega \times [0,\infty) \to \mathbb{R}$ defined by the Level Lines Shortening evolution of u_0

$$u(x,t) = LLS(t)u_0(x), \forall x \in \mathbb{R}^2, \forall t \in [0,\infty)$$

is a viscosity solution for the mean curvature PDE, with the initial data u_0:

$$\begin{cases} u_t = (\delta_{ij} - \frac{u_{x_i} u_{x_j}}{|Du|^2}) u_{x_i x_j}, & in\ \mathbb{R}^2 \times [0,\infty) \\ u(\cdot, 0) = u_0, & on\ \mathbb{R}^2. \end{cases} \quad (3)$$

Local comparison principle and regularity

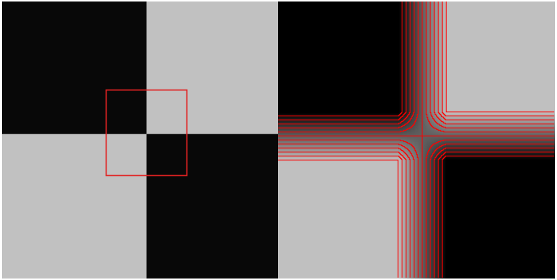


Figure: Original image and its level lines

Local comparison principle and regularity

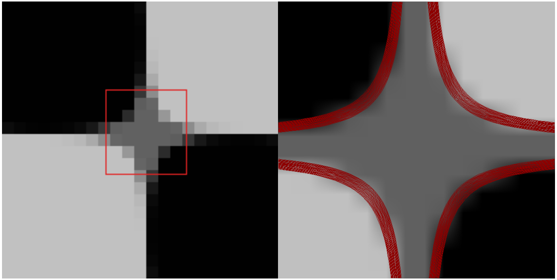


Figure: Level Lines Shortening

Local comparison principle and regularity

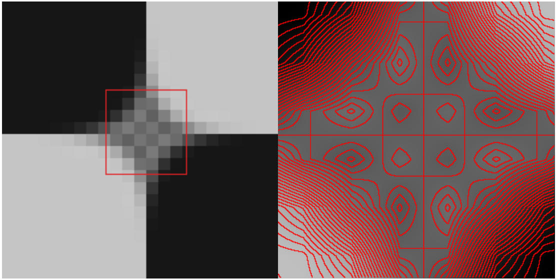


Figure: Finite difference scheme

Local comparison principle and regularity

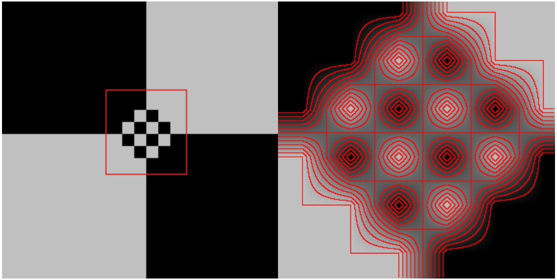


Figure: Stack filter and theshold dynamics

JPEG artifacts reduction on color images

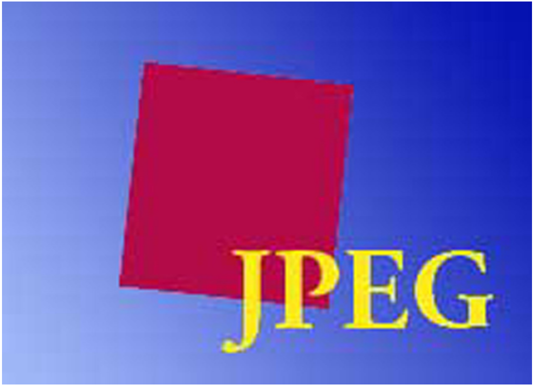


Figure: Original image, suffering of JPEG artifacts such as Gibbs oscillations, staircase noise along curving edges and checkerboarding.

JPEG artifacts reduction on color images

Figure: LLAS is applied separately to each RGB channel. Although diffusions occur at junctions, LLAS considerably reduces these artifacts.

Curvatures computed directly on level lines

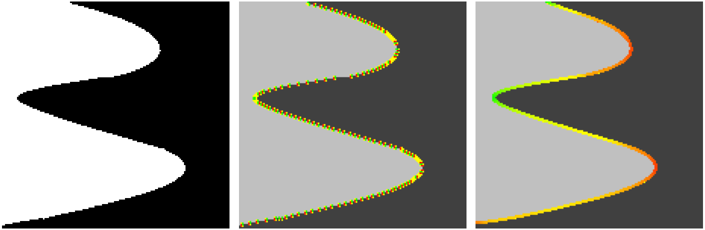


Figure: The curvature color display rule. Initial image, FDS and LLS.

Discrete curvature for a polygonal line.

We recall that each level line is stored as a set of ordered points

$$\Sigma = \{P_i(x_i, y_i)\}_{i=0..n}, \text{ with } P_0 = P_n.$$

The discrete scalar curvature k_i computed at each vertex P_i is obtained as the inverse of the circumscribed radius R_i of the triangle $P_{i-1}P_iP_{i+1}$.

Lemma

The curvature at vertex P_i is given by

$$k_i = 2\frac{u_i^1 u_{i+1}^2 - u_i^2 u_{i+1}^1}{u_i u_{i+1} v_i}. \quad (4)$$

Subpixel curvature algorithm

Algorithm (C., Monasse, Morel, '10)

Compute the image curvature microscop

- *Extract the tree of level lines* $\{\Sigma_0^{\lambda,i}\}_{\{i \in F_\lambda, \lambda\}}$*;*
- *Perform uniform, fine sampling uniformly each level line;*
- *Smooth each level line separately*

$$\Sigma_t^{\lambda,i} = \textit{ Curve Shortening Flow } (\Sigma_0^{\lambda,i})$$

- *Compute the discrete curvatures at each vertex;*
- *Register at each dual pixel the average of all discrete curvatures computed in and create thus the curvature image.*

Signed and topological curvatures

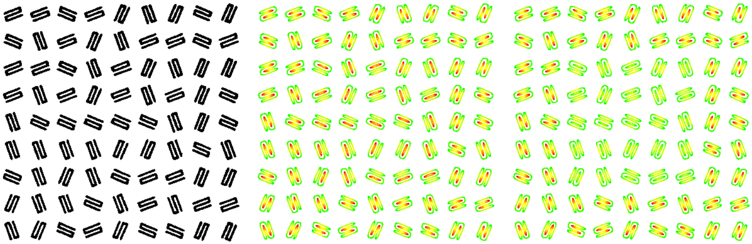


Figure: Original image, signed curvatures and topological curvatures

Curvature Microscope

Figure: Original image, 2X zoom and 4X zoom of the up-right corner. A zoom is necessary to observe the single curvatures.

Curvature Microscope

Figure: Curvature map computed on the original level lines with a quantization step $s = 36$.

Curvature Microscope

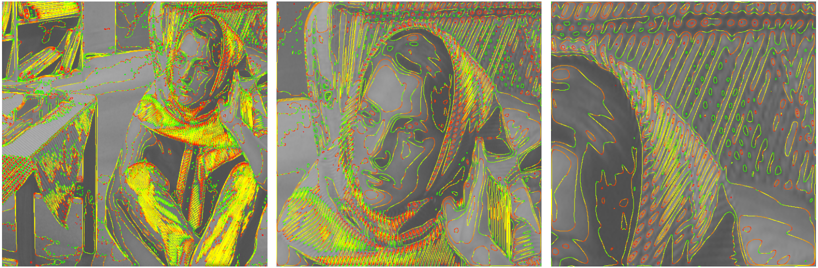


Figure: Curvature map computed on shortened level lines at normalized scales $l = 1$, $l = 2$, and $l = 4$.

A closer look at Attneave's cat

Figure: Zoom on the Attneave cat, its corresponding level lines and curvatures.

A closer look at Attneave's cat

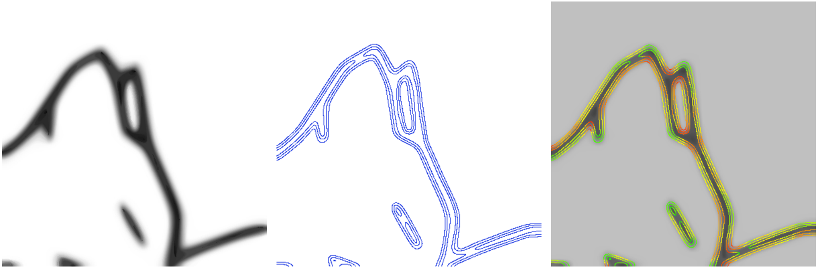


Figure: LLAS evolution, affine smoothed level lines and curvature map after filtering.

Graphics and aliasing

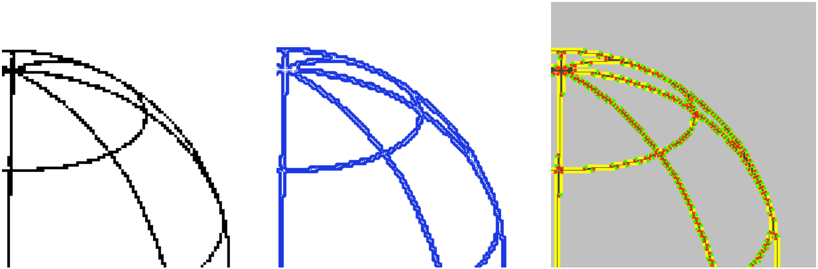


Figure: Original image, its corresponding level lines and curvatures.

Graphics and aliasing

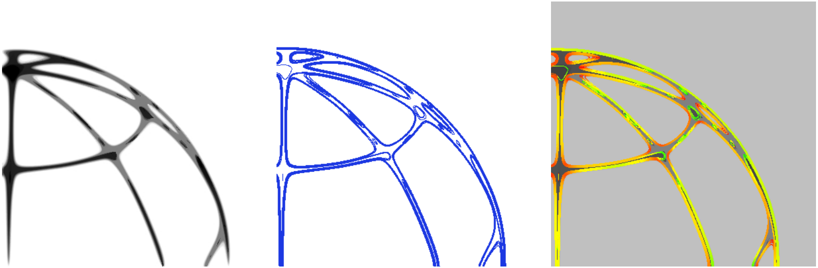


Figure: LLAS evolution, affine smoothed level lines and curvature map after filtering.

Bacteria morphologies

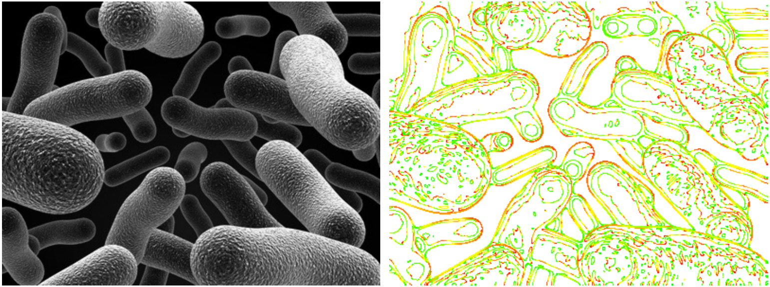


Figure: Original bacteria image and the corresponding curvature map.

Digital elevation models

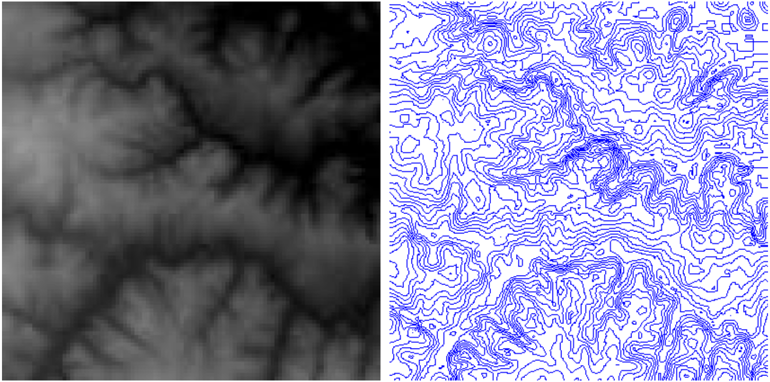


Figure: Digital elevation map and its corresponding level lines.

Digital elevation models

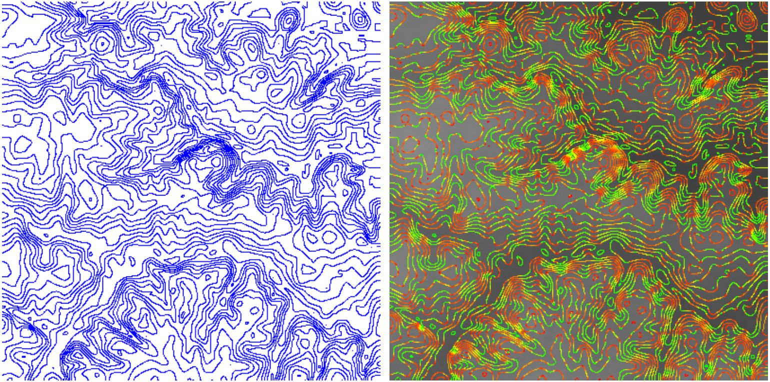


Figure: The affine smoothed level lines and their curvature map.

Paitings *sfumato technique*

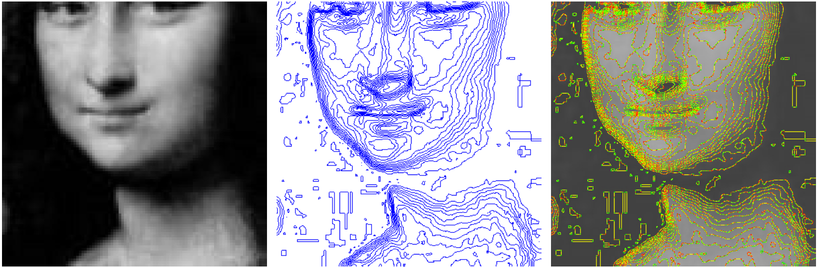


Figure: Extraction with zoom of *Mona Lisa* photograph, its corresponding level lines and curvatures.

Paitings *sfumato technique*

Figure: LLAS evolution, affine smoothed level lines and curvature map after filtering.

Text processing

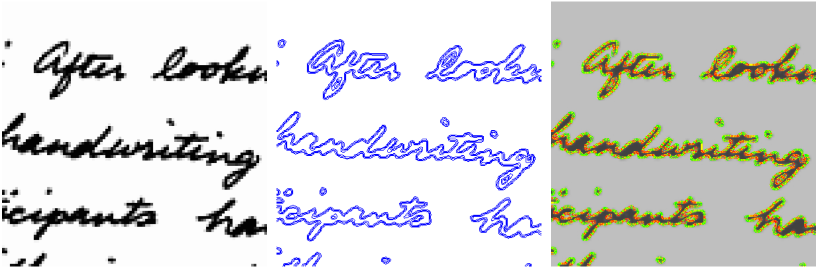


Figure: Original handwriting, corresponding level lines and curvatures.

Text processing

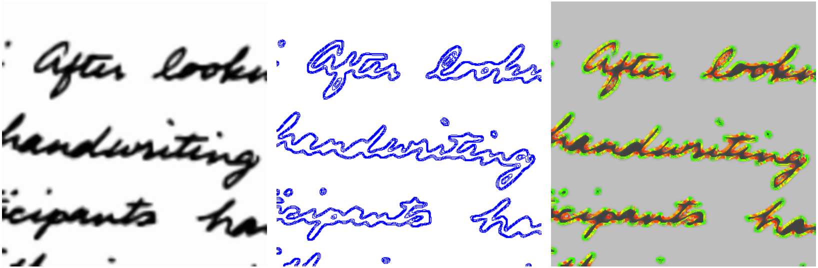


Figure: LLAS evolution, affine smoothed level lines and curvature map after filtering.

Fingerprints restoration and discrimination

Figure: Original fingerprint, Level Lines Affine Shortening and its Curvature map.

Conclusion

- *The first outcome of the Level lines Shortening algorithm is the evolved image, which presents some sort of denoising, simplification, and desaliasing;*
- *The main outcome is an accurate curvature estimate on all level lines;*
- *A powerful visualization tool, due to the fact that all level lines are polygons with real coordinates allows to zoom in the image at an arbitrary resolution;*
- *It runs online at www.ipol.im/pub/algo/cmmm_image_curvature_microscope/.*