

IMAGE AND VIDEO INPAINTING

MVA 2023-2024

Yann Gousseau
Telecom Paris



Image Inpainting

Inpainting (disocclusion) : How to fill missing regions in images ?
Should be done in a **plausible** way.

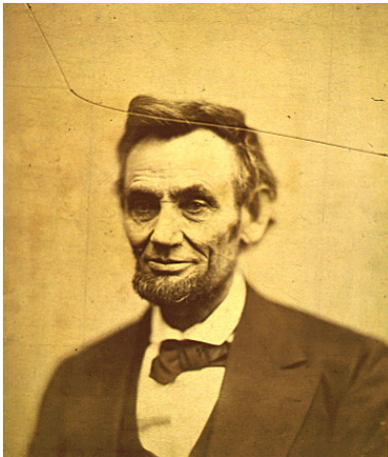


Image Inpainting

Inpainting (disocclusion) : How to fill missing regions in images ?
Should be done in a **plausible** way.



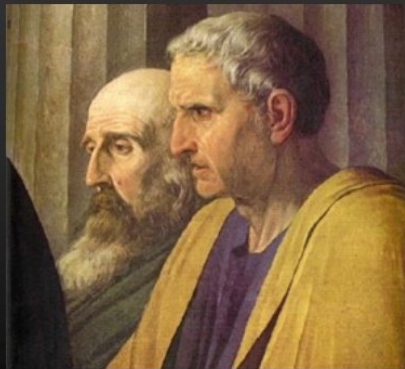
Image Inpainting

Inpainting (disocclusion) : How to fill missing regions in images ?
Should be done in a **plausible** way.



Image Inpainting

Inpainting (disocclusion) : How to fill missing regions in images ?
Should be done in a **plausible** way.



Detail of "Cornelia, Mother of the Gracchi" by J. Suvee (Louvre).

Image Inpainting

Inpainting (disocclusion) : How to fill missing regions in images ?
Should be done in a **plausible** way.



Image Inpainting

Inpainting (disocclusion) : How to fill missing regions in images ?
Should be done in a **plausible** way.



Image Inpainting

Inpainting (disocclusion) : How to fill missing regions in images ?
Should be done in a **plausible** way.

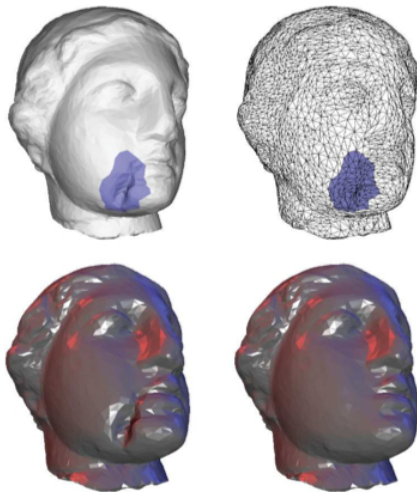


Image Inpainting

Inpainting (disocclusion) : How to fill missing regions in images ?
Should be done in a **plausible** way.



3D inpainting



(Bobenko, Schroder, 2005)

3D inpainting

3D model with a missing region

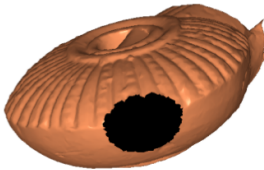


Completed model

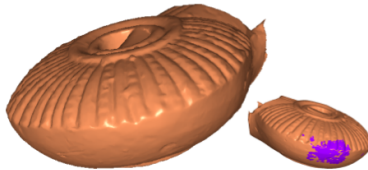


(Kawai, Sato, Yokoya, 2009)

3D inpainting



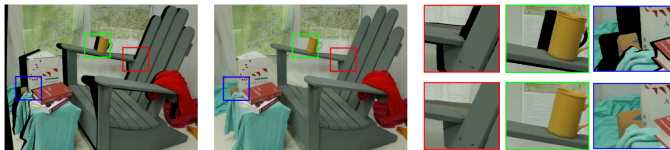
(a) hole



(b) our completion

(Harary et al., 2014)

Virtual view synthesis



(Buyssens et al., 2015)

Video inpainting



Inpainted video

Video inpainting



Original video

Historical example



The commissar vanishes (from www.newseum.org)

Applications

- Image and video editing
- Video post-production, visual effects
- Restoration of old materials (photographs and movies)
- Zoom, super-resolution, deinterlacing
- Multi-image restoration (moving objects)
- Etc.

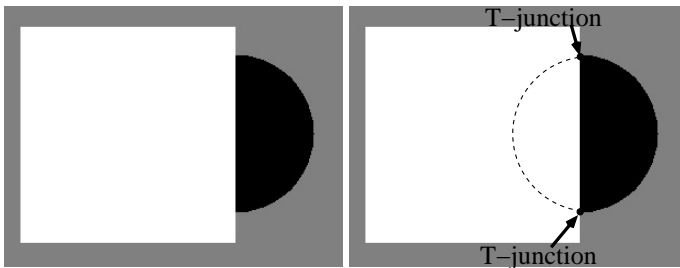
Human visual system and occlusions

Objects are (mostly) opaque \rightarrow most objects are only partially visible !



Our visual system is able to infer missing parts by **amodal completion**.

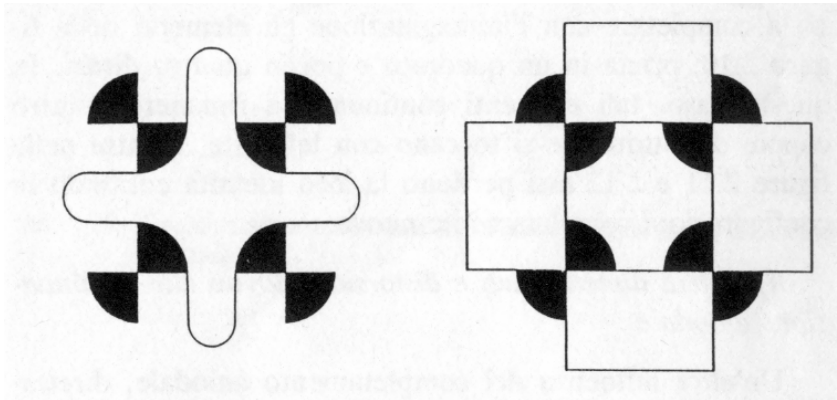
Amodal completion



Curves are interpolated smoothly between **T-junctions**

G. Kanizsa, *Organization in Vision: Essays on Gestalt Perception*, Prager, 1979

Amodal completion



Curves are interpolated smoothly between T-junctions

Amodal completion

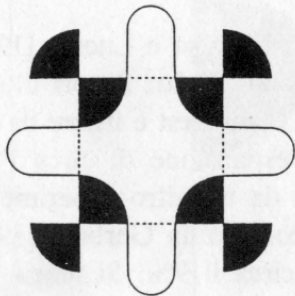


FIG. 2.13. Così si completa la figura 2.11.

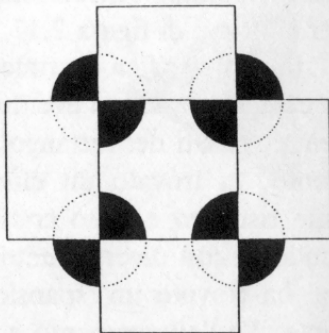


FIG. 2.14. Così si completa la figura 2.12.

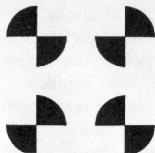


FIG. 2.10. Quattro coppie di settori neri. I vari completamenti mentalmente possibili non incidono sulla loro identità.

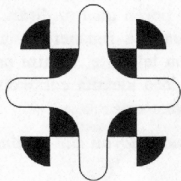


FIG. 2.11. Un quadrato con quattro appendici, come in figura 2.13.

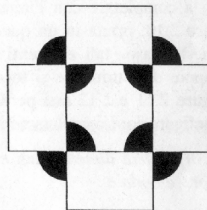
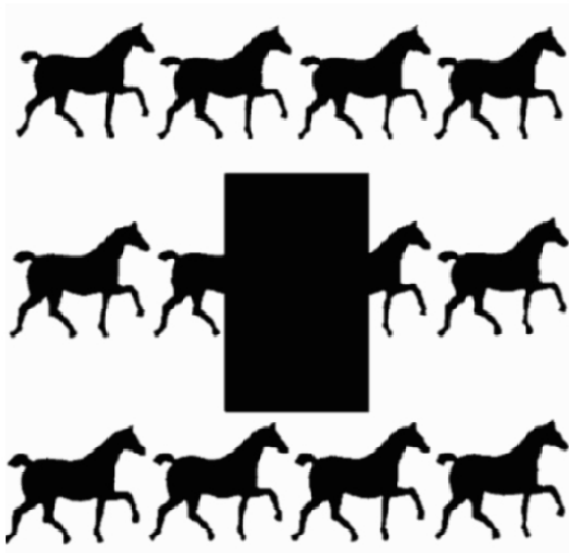


FIG. 2.12. Quattro dischi come in figura 2.14.



Amodal completion supersede common sense/previous knowledge !

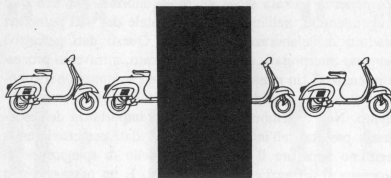
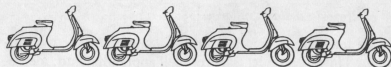


FIG. 2.29. Lo scooter «allungato».

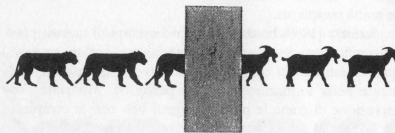
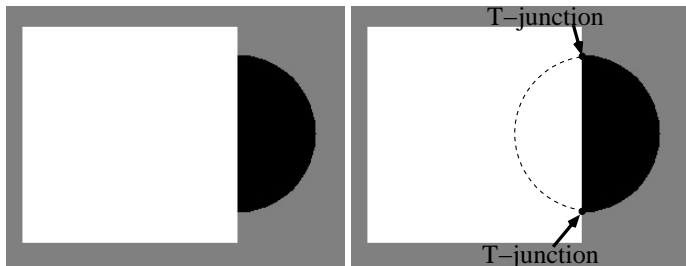


FIG. 2.30. Un ibrido creato dal completamento amodale.

First approach to inpainting (desocclusion) : Masnou-Morel 1998
Mimicks the human visual system



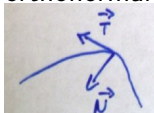
Virtual contour model = Euler elastica = $\operatorname{argmin} \int_0^{\mathcal{L}} (\alpha + \beta |\kappa|^2) ds$
with boundary constraints of order 1

Express differential geometry reminder

- $C : [a, b] \rightarrow \mathbb{R}^2$ is a **Jordan curve** if $C(p_1) \neq C(p_2)$ for $p_1 \neq p_2$.



- the choice of the parametrization C is of course not unique
- $L(a, p)$: length of C between a and p
parametrization is called **arc-length parametrization** if $\frac{dL}{dp} = 1$
- If C is twice differentiable and $C'(p) \neq 0$
the tangent vector is defined as $\vec{T} = \frac{C'(p)}{|C'(p)|}$
the normal vector \vec{N} is such that (\vec{T}, \vec{N}) is a direct orthonormal basis



- **curvature**

$\exists k$ such that

$$\frac{1}{|C'|} \frac{d\vec{T}}{dp} = k\vec{N}$$

and k is independent of the parametrization

$k\vec{N}$ is called the curvature vector

- For an arc-length parametrization :

$$\vec{T} = C'(s),$$

$$\frac{d\vec{T}}{ds} = k\vec{N} = C''(s)$$

(because $L(a, p) = \int_a^p |C'(u)| du$, so that $|C'(s)| = 1$.)

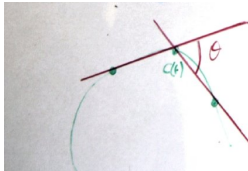
- writing $C(p) = (x(p), y(p))$, we get

$$k = \frac{y''x' - x''y'}{(x'^2 + y'^2)^{3/2}}$$

- the curvature satisfies $k(p) = r(p)^{-1}$, where $r(p)$ is the radius of the circle that best approximate the curve at $C(p)$ (osculating circle)



- In practice, one can approximate the curvature by the difference between two consecutive direction of the tangent vector (more robust than direct second order derivatives).



$$k \approx \frac{\theta}{\Delta s}$$

From curves to images: using the level set framework

Level lines of a (gray level) image are lines of **constant intensity** or, "equivalently", the boundaries of $\{x, u(x) \geq t\}$.

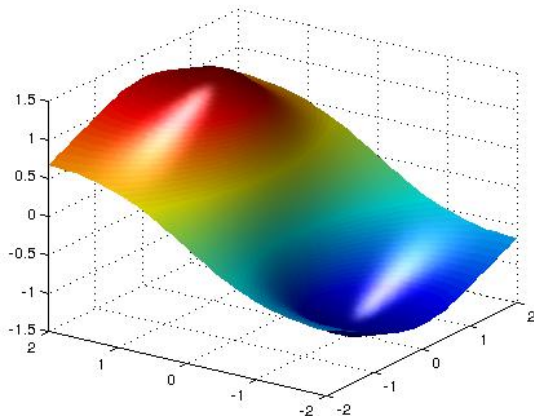


Figure: Graph of $f(x, y) = \frac{3y}{x^2 + y^2 + 1}$

From curves to images: using the level set framework

Level lines of a (gray level) image are lines of **constant intensity** or, "equivalently", the boundaries of $\{x, u(x) \geq t\}$.

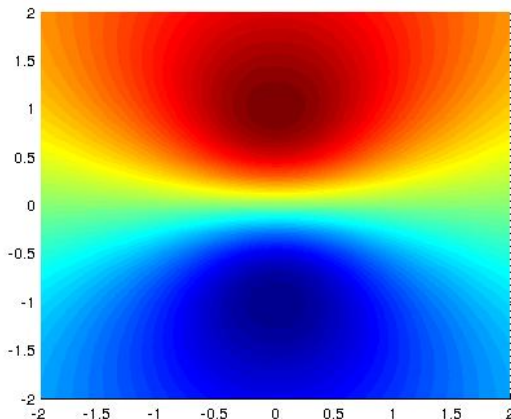


Figure: Graph of $f(x, y) = \frac{3y}{x^2 + y^2 + 1}$ (viewed from above)

From curves to images: using the level set framework

Level lines of a (gray level) image are lines of **constant intensity** or, "equivalently", the boundaries of $\{x, u(x) \geq t\}$.

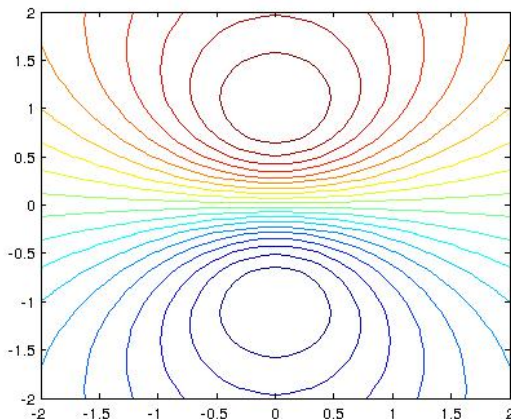


Figure: Some level lines

From curves to images: using the level set framework

Set of lines : "the topographic map" of the image

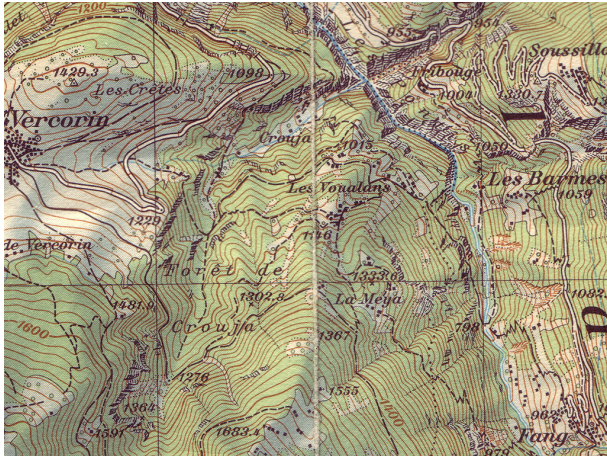


Figure: A topographic map

From curves to images: using the level set framework

Set of lines : "the topographic map" of the image



Adaptation to inpainting: using the level set framework (Masnou-Morel 1998)

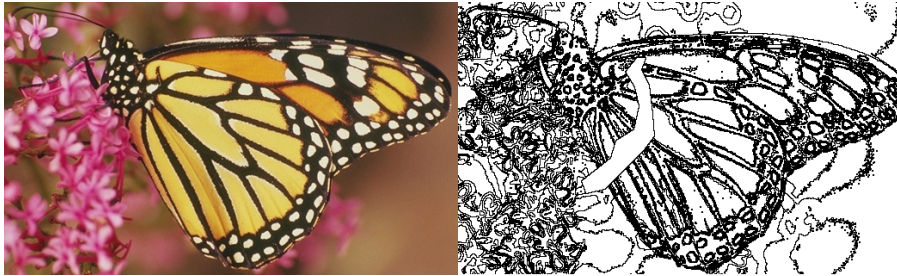


$$\text{Level sets } X_t^u = \{y : u(y) \geq t\} \iff u(x) = \sup \left\{ t : x \in X_t^u \right\}$$

Level lines = Boundaries of level sets

Level lines reconstruction \iff Image restoration

Adaptation to inpainting: using the level set framework (Masnou-Morel 1998)

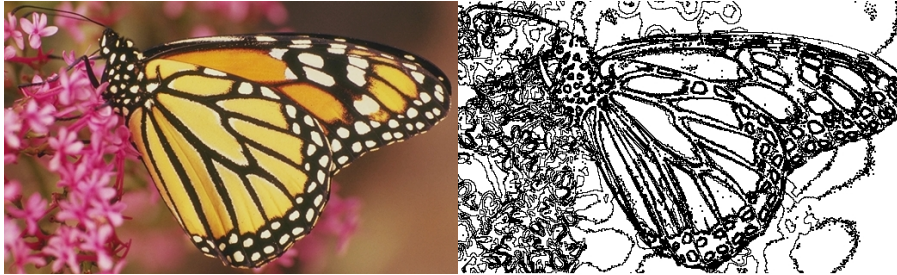


Level sets $X_t^u = \{y : u(y) \geq t\} \iff u(x) = \sup \left\{ t : x \in X_t^u \right\}$

Level lines = Boundaries of level sets

Level lines reconstruction \iff Image restoration

Adaptation to inpainting: using the level set framework (Masnou-Morel 1998)



$$\text{Level sets } X_t^u = \{y : u(y) \geq t\} \iff u(x) = \sup \left\{ t : x \in X_t^u \right\}$$

Level lines = Boundaries of level sets

Level lines reconstruction \iff Image restoration

Adaptation to inpainting: using the level set framework (Masnou-Morel 1998)



$$\int_{-m}^{+M} \left(\sum_{\text{Paired junctions}} \int (\alpha + \beta |\kappa_{X_t^u}|^p) d\mathcal{H}^1 \right) dt \iff \int |\nabla u| \left(\alpha + \beta \left| \operatorname{div} \frac{\nabla u}{|\nabla u|} \right|^p \right) dx$$

Minimization over collections of curves



Constructive approach

Classical minimization



Global approach

Many works have followed :

- **variational/PDE approaches**

(Masnou-Morel 1998, Chan-Shen 2001, Caselles et al. 2001, Bertalmio et al 2001, Tschumperlé-Deriche 2004, Bornemann and März 2007, Schönlieb - Bertozzi 2011, Chizhov et al. 2021, etc.)

- **exemplar-based, patch-based**

(Efros-Leung 1999, Wei-Levoy 2000, Efros-Freeman 2001, Ashikmin et al 2001, Harrison 2001, Criminisi-Pérez-Toyama 2004, Pérez-Gangnet-Blake 2004, de Bonet 1997, Igehy-Pereira 1997, Komodakis 2007, Kawai et al. 2009, Arias et al. 2011, Liu-Caselles 2013, Wang 2013, Newson et al. 2014, Daisy et al. 2015, etc.)

Two main trends:

- greedy (sequential)
- global, patch-based optimization (parallel)

- **inpainting in transform domains**

(Elad et al. 2005, Chan et al. 2006, Fadili et al. 2007, Cai 2008)

- **Convolutional neural networks**

(Pathak et al. 2016, Iizuka et al. 2017, Yu et al. 2018, 2019, Liu et al. 2018, Nazeri et al. 2019, Yi et al. 2020, Saharia et al. 2021, Lugmayr et al. 2022, etc.)

Simplest approach : heat equation

Image I and hole (occlusion) Ω

$$\frac{\partial I}{\partial t} = \Delta I \quad \text{inside } \Omega$$

and

$$I = I_0 \quad \text{outside } \Omega$$

Information is propagated by averaging :



Blurred results

In a discrete setting :

$$\Delta(u)(i, j) \approx u(i+1, j) + u(i-1, j) + u(i, j+1) + u(i, j-1) - 4u(i, j)$$

$$u^{n+1}(i, j) - u^n(i, j) = \delta t \Delta u^n(i, j)$$

$$u^{n+1}(i, j) = (1 - 5\delta t)u^n(i, j) + \delta t (u^n(i+1, j) + u^n(i-1, j) + u^n(i, j+1) + u^n(i, j-1) + u^n(i, j)),$$

→ local smoothing

Bertalmío, Sapiro, Caselles, Ballester (2000)

- Introduce the term "inpainting"
- Evolution equation :

$$\frac{\partial u}{\partial t} = \nabla \Delta u \cdot \nabla^\perp u$$

(+anisotropic diffusion for stabilization)

- Idea : a measure of smoothness (Δu) is "transported" along the isophotes directed by $\nabla^\perp u$
by analogy with a transport equation $\frac{\partial u}{\partial t} = -\text{div}(u\vec{v})$,
where \vec{v} is the speed. If \vec{v} is constant, then

$$\frac{\partial u}{\partial t} = -\nabla(u) \cdot \vec{v}$$

- Efficient for small and non-textured occlusions
- Many variants and follow-up
(see e.g. *Partial Differential Equation Methods for Image Inpainting*, Schoenlieb, 2016)





From Bertalmio et al. 2000



From Bertalmio et al. 2000

Alternative: the denoising viewpoint

- Chan, Shen (2001) (Total variation)

$$\int_A |\nabla u| dx + \frac{\lambda}{2} \int_{\Omega} |u - u_0|^2 dx,$$

A being the image domain

- Chan, Kang, Shen (2002) & Esedoglu, Shen (2002)
(Mumford-Shah-Euler)

$$\int_{\Omega \setminus A} |u - u_0|^2 dx + \int_{\Omega \setminus K} |\nabla u|^2 dx + \int_K (\alpha + \beta k^2) ds.$$

- and many, many other contributions (higher-order inpainting, topological analysis, fractional-order inpainting, etc.) !

Variational/PDE methods

Advantages : fast, mathematical interpretation (strong geometrical property)

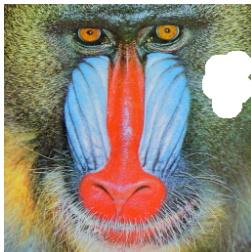
Main limitation : no texture



15×15 patches are removed (from Masnou et al. 2011)

Variational/PDE methods

Advantages : fast, mathematical interpretation (strong geometrical property) Main limitation : no texture



Large inpainting using the Total Variation

Nonlocal methods : from texture synthesis to inpainting

- The patch-based texture synthesis method of Efros and Leung (that we saw in the texture synthesis lecture) can be straightforwardly applied to the inpainting problem
- Many patch-based methods followed from the 2000's

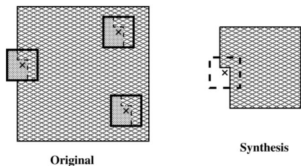
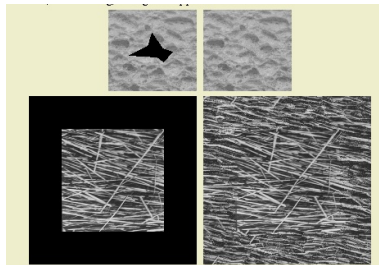


FIG. 1. *The nonparametric resampling algorithm.*



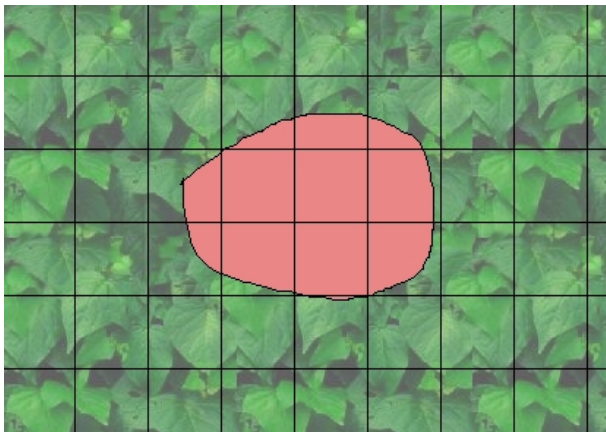
III. Levina-Bickel 2006 - Efros-Leung 1999

Application to inpainting

Many papers, and many methods :

- Drori et al. 2003 (multiscale sampling)
- Criminisi et al. 2004, Pérez et al 2014 (greedy approach, priority order for the filling-in) → next slides
- Sun et al. 2005 (user-assisted method to help the recovery of geometric structures)
- Wexler et al. 2005, Newson et al 2017 (global patch-based energy, heuristic for the minimisation) → second part of the lecture
- Komodakis et al. 2007 (variational and patch-based strategy, minimization with belief propagation)
- Cao et al. 2011 (patch-based strategy with automatic geometrical guide)
- Arias et al. 2011 (variational framework for non-local patch-based inpainting)
- Liu-Caselles 2013 (multi-scale graph-cut)
- and a lot more...

→ synthèse par "patches" (Efros-Freeman 2000,
Prez-Gangnet-Blake 04)



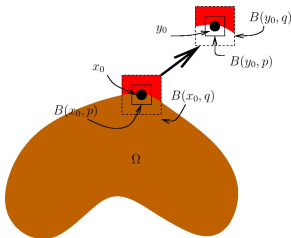
Principe (Pérez-Gangnet-Blake 04)

Soit Ω la région à reconstruire, $0 < p < q$ deux paramètres.

Soit $B(x, p)$ le patch centré sur x de rayon p et

$C(x, p, q) = B(x, q) \setminus B(x, p)$.

- 1) soit $x_0 \in \partial\Omega$ ayant un nombre de voisins maximum dans Ω^c .
- 2) soit $y_0 \in \Omega^c$ qui minimise la norme L^2 entre $C(x_0, p, q) \setminus \Omega$ et $C(y_0, p, q) \setminus (\Omega + y_0 - x_0)$.
- 3) pour chaque $x \in B(x_0, p) \cap \Omega$ soit $I(x) = I(x + y_0 - x_0)$.
- 4) remplacer Ω par $\Omega \setminus B(x_0, p)$ et itérer.



Nombreuses variantes de cet algorithme (choisir au hasard un patch proche, injecter de l'invariance en considérant des rotations des patches, etc.)

Les résultats dépendent de

① **paramètres p, q**

en général préférable de choisir $p > 0$ (patches au lieu de pixels)

p grand \rightarrow meilleur respect de la géométrie, moins d'"innovation"

q grand \rightarrow meilleures transitions entre patches

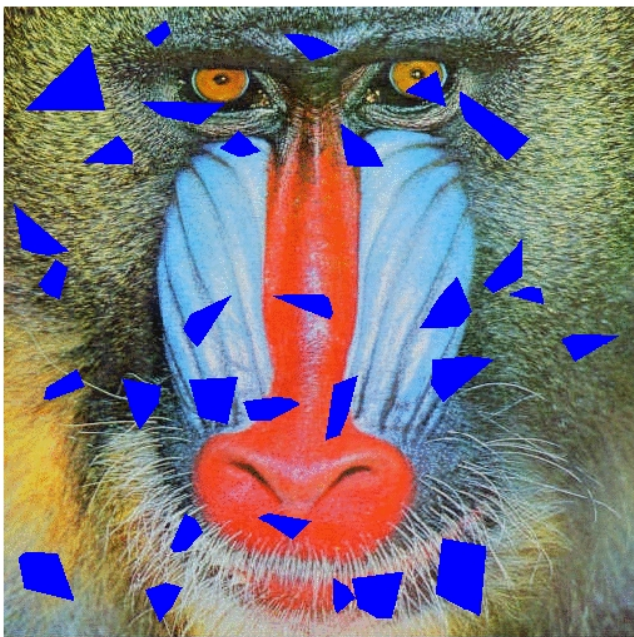
② **ordre de remplissage:**

quel $x_0 \in \partial\Omega$ choisir à chaque itération ?

[Criminisi-Pérez-Toyama '04] : x_0 minimise une fonctionnelle dépendant de

i) la géométrie de Ω ,

ii) la géométrie des lignes de niveau autour de x_0 .

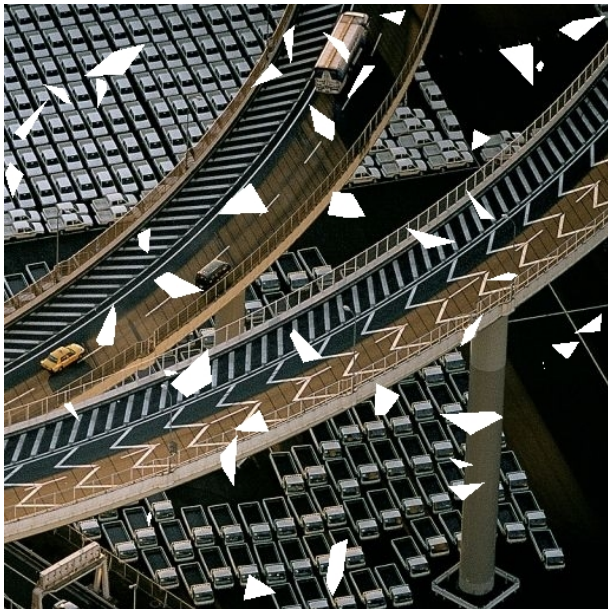




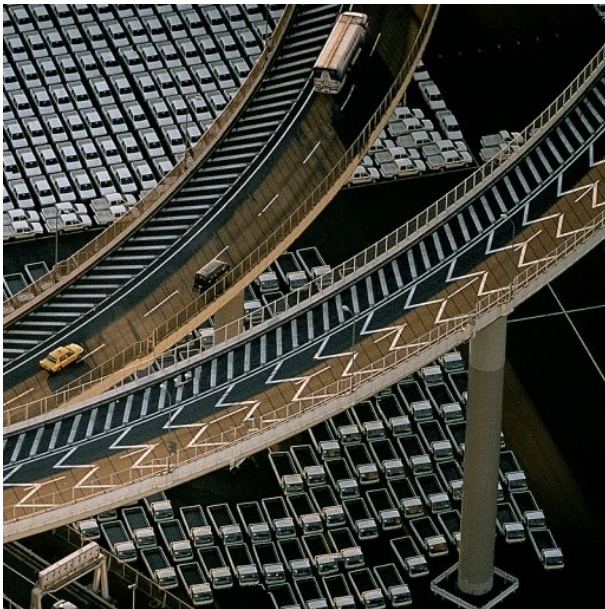




Application to inpainting



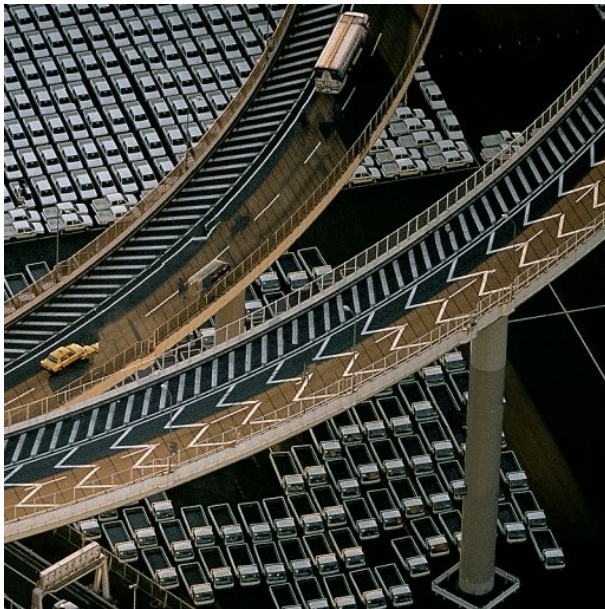
Application to inpainting



Application to inpainting



Application to inpainting



However these methods may fail at reconstructing long-range geometric features, e.g. long edges



Different approaches have been proposed for the **simultaneous restoration of geometry and texture** (Bertalmio et al '03, Fadili-Starck '05, Tschumperlé et al. 2006, Cao et al. 2011, etc.)

Usually rely on hybrid approaches

⇒ Possible solution : use a **geometric guide** computed on a simplified image

Geometrically guided exemplar-based inpainting (Cao et al. 2011)

Step 1: compute a geometric sketch



Step 2: restore the geometric sketch by interpolation of the level lines with Euler spirals

Euler elasticae are solutions of

$$\text{Min} \int_0^L (1 + |\Psi'(s)|^2) ds \quad (\Psi = \text{angle}(\text{tangent}, \text{horizontal axis}))$$

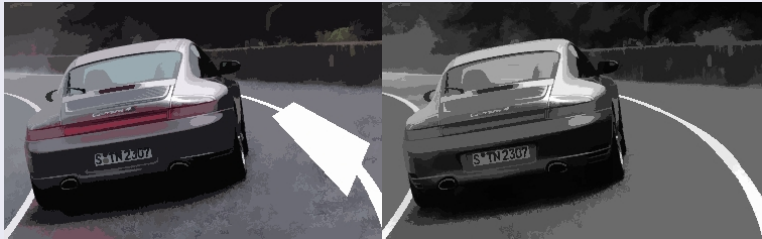
under endpoints and end-tangents constraints.

Euler-Lagrange equation: $(\Psi')^2 = 1 + \lambda \cos \Psi + \mu \sin \Psi$.

After **linearization** $(\Psi')^2 = 1 + \lambda + \mu \Psi$ whose solutions are **Euler spirals**:

- Curvature= **affine function of arc-length**
- Very useful in civil engineering, industrial design, typography. Also used as a model for shape completion in vision.

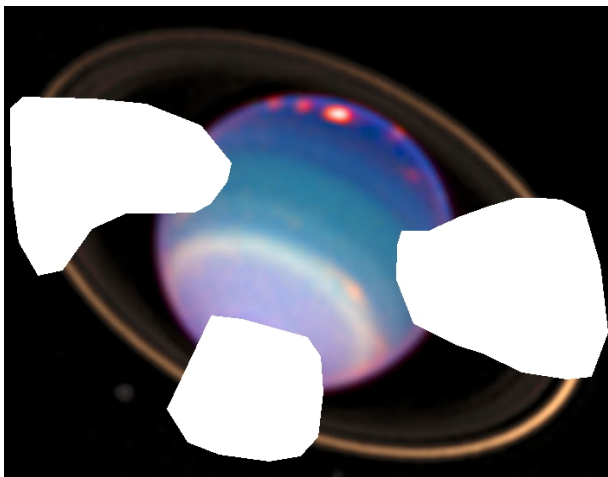
Step 2: restore the geometric sketch by interpolation of the level lines (e.g. with Euler spirals)



Step 3: use the reconstructed sketch as a geometric guide

New metric between patches = Linear combination of a L^2 metric on the original image (conditioned by the inpainting domain) and a L^2 metric on the (complete) geometric sketch (Many possible variants)





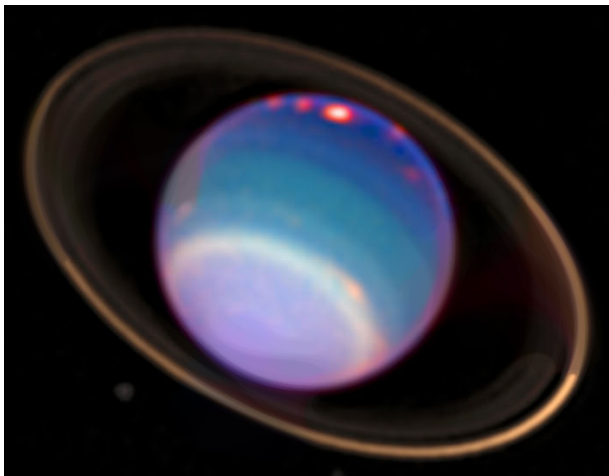




Image with missing region



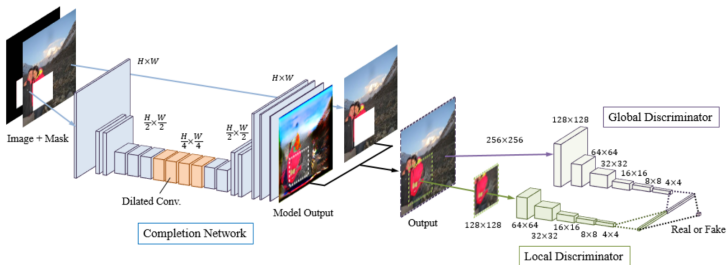
Method from Cao et al. '11

Global optimization-based approaches

- Initiated by the works of Demanet et al. 2003, Wexler et al. 2005
- The missing region is reconstructed by stitching patches from the images, whose coherence is ensured by an iterative approach
- → texture + relatively good global geometric coherence (best approaches to date without learning)
- → detailed for *video inpainting* in the second part of the course

Inpainting has recently been tackled with Convolutional Neural Networks (CNN) (Pathak et al. 2016, Iizuka et al. 2017, Yang et al. 2017, Liu et al. 2018, 2019, Yi et al. 2020, Suvorov et al. 2022, etc.)

- Use ideas from autoencoders and Generative Adversarial Networks (Goodfellow et al. 2014)
- Implicitly use information not from the inpainted image (this was sometimes done explicitly before, see Hays-Efros 2007)
- Training involves several millions images and weeks of computation



Global architecture of the method from Iizuka et al. 2017



From Iizuka et al. 2017



From lizuka et al. 2017
Typically outside the reach of patch-based methods



ccn-based method ; the image is aligned, as in the training dataset



after a 10 pixels translation



patch-based method

Experiment courtesy of E. Bonnal



May yield artefacts



May yield artefacts

Contextual attention (DeepFill, Yu et al. 2018)

Hybrid method (CNN / patch-based)

Take into account patches near the missing region at training time

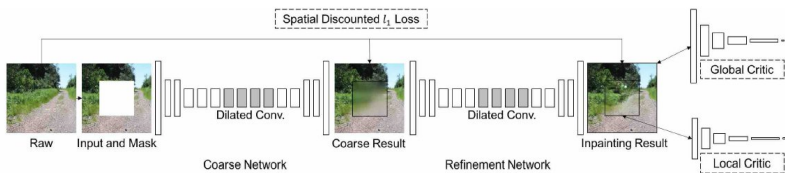


Figure 2: Overview of our improved generative inpainting framework. The coarse network is trained with reconstruction loss explicitly, while the refinement network is trained with reconstruction loss, global and local WGAN-GP adversarial loss.

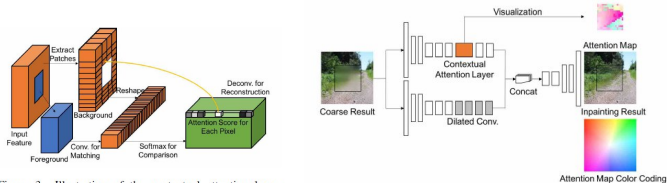
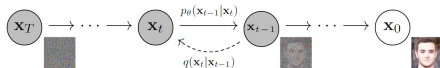


Figure 3: Illustration of the contextual attention layer.

- Edge connect (Nazeri et al. 2019)
Learn a sketch reconstruction component
- Free-form image inpainting (DeepFill v2, Yu et al. 2019)
Use of Gated convolution
- Contextual residual aggregation (Yi et al. 2020)
- Local inpainting in the Fourier domain(LAMA, Suvorov et al. 2022)
- Diffusion models (REPAINT Lugmayr et al. 2022, PALETTE Saharia et al. 2022, latent diffusion models, aka stable diffusion, Rombach et al 2022, text-guided inpainting, Smartbrush 2023, etc.)

Diffusion models for inpainting



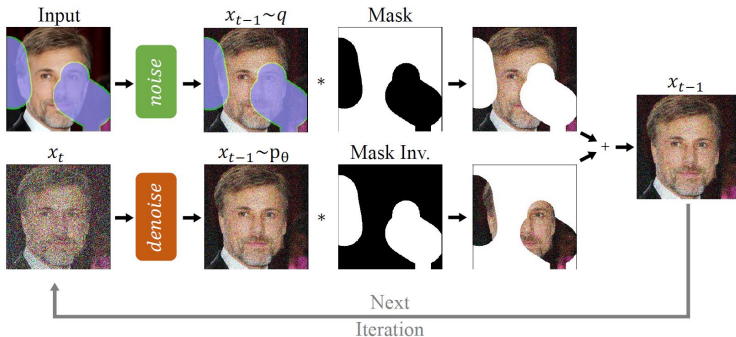
- Rely on Denoising Diffusion Probabilistic Models (DDPM)
- DDPMs generate images by progressive denoising of a noise input (Sohl-Dickstein et al. 2015, Ho et al. 2020)
- “Denoising” rely on a CNN trained to reverse the following process

$$q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{1 - \beta_t}x_{t-1}, \beta_t\mathbf{I})$$

- More precisely, the network is trained to learn μ_θ and Σ_θ of the process

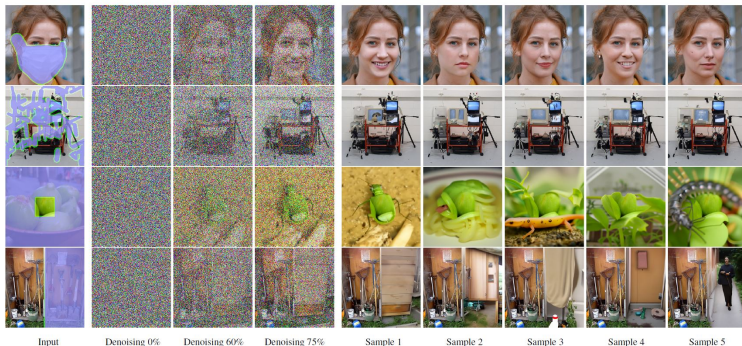
$$p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t))$$

- The framework is adapted to the inpainting task



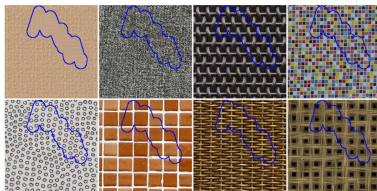
$$\begin{aligned}
 x_{t-1}^{\text{known}} &\sim \mathcal{N}(\sqrt{\bar{\alpha}_t}x_0, (1 - \bar{\alpha}_t)\mathbf{I}) \\
 x_{t-1}^{\text{unknown}} &\sim \mathcal{N}(\mu_\theta(x_t, t), \Sigma_\theta(x_t, t)) \\
 x_{t-1} &= m \odot x_{t-1}^{\text{known}} + (1 - m) \odot x_{t-1}^{\text{unknown}}
 \end{aligned}$$

- enables unprecedented quality and diversity

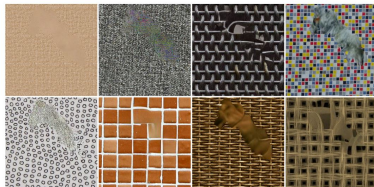


Reasonable generalization capacity

- trained on a relatively generic scene database (Places2)



- trained on a face database (CelebA-HQ)



Experiments courtesy of N. ChereI

But rely on huge networks ...

- about 500M parameters
- memory impact is about 3GB for 256x256 images
- heavy environmental impact of the training stage :
for celebA-HQ (30000 images):
500h + training time, about 10kg CO₂