

Analyse et synthèse de textures

IMA201

Télécom Paris - IP Paris

IMA201
2021

1 Généralités

2 Analyse

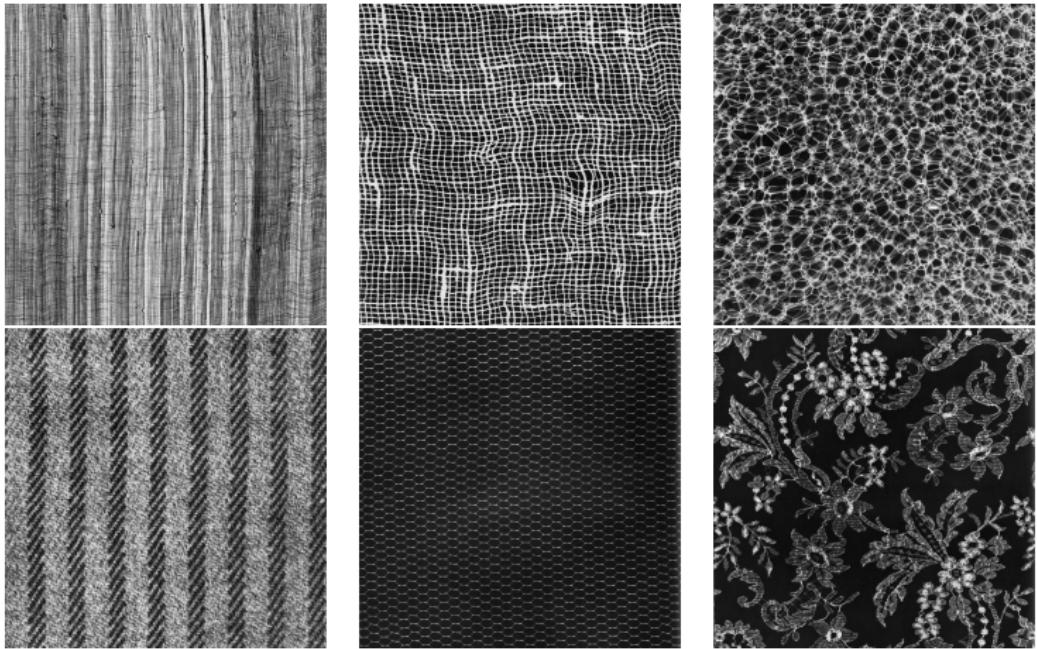
- Statistiques pixéliques
ordre deux et cooccurrences ; champs de Gibbs
- Approche structurelle et morphologie mathématique
- Filtres de Gabor, ondelettes et variantes

3 Synthèse

- Quelques modèles ad'hoc
- Méthodes par contraintes statistiques
- Méthodes par ré-échantillonage de patchs
- Méthodes hybrides



Quelques textures



Textures de l'album de Brodatz

De nombreuses significations

- **Matériaux** : Caractéristique relative à la dimension, la forme, la disposition des grains dans un métal,
- **Culinaire** : Qualité physique des aliments liée à leur densité, leur viscosité, leur caractère homogène, leur dureté,
- **Textile** : disposition et mode d'entrecroisement des fils dans un tissage,
- **Graphisme par ordinateur** : recouvrement d'une surface (généralement par une image) lui conférant son apparence,
- Autres définitions en cristallographie, cosmologie, musique, peinture, etc.

Définitions possibles dans le cadre de la vision

- Zone homogène, constituée de la répétition d'entités élémentaires formant un tout.
« Un subtile équilibre entre répétition et innovation »
Yves Meyer
- Description des variations d'intensité lumineuse sur une surface, rendant compte de propriétés telles que la rugosité, la douceur, la granularité, etc.

De nombreuses significations

- **Matériaux** : Caractéristique relative à la dimension, la forme, la disposition des grains dans un métal,
- **Culinaire** : Qualité physique des aliments liée à leur densité, leur viscosité, leur caractère homogène, leur dureté,
- **Textile** : disposition et mode d'entrecroisement des fils dans un tissage,
- **Graphisme par ordinateur** : recouvrement d'une surface (généralement par une image) lui conférant son apparence,
- Autres définitions en cristallographie, cosmologie, musique, peinture, etc.

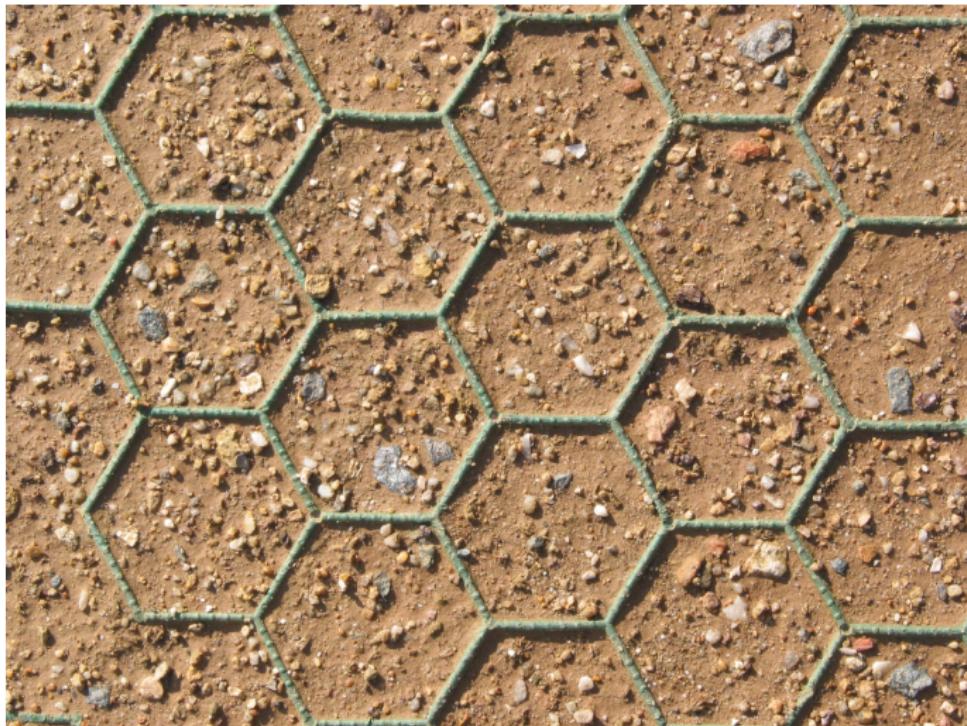
Définitions possibles dans le cadre de la vision

- Zone homogène, constituée de la répétition d'entités élémentaires formant un tout.
« Un subtile équilibre entre répétition et innovation »
Yves Meyer
- Description des variations d'intensité lumineuse sur une surface, rendant compte de propriétés telles que la rugosité, la douceur, la granularité, etc.

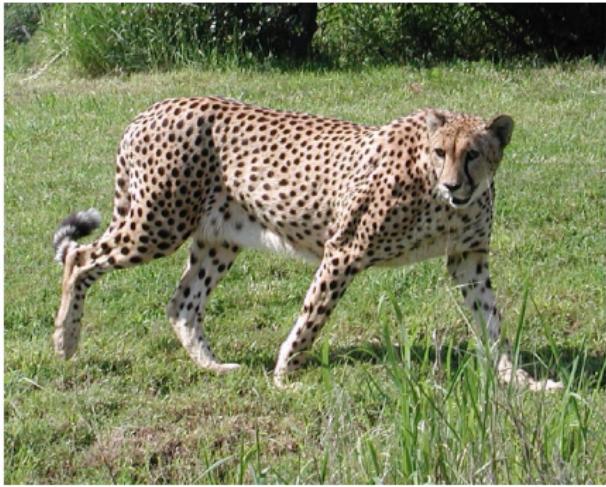
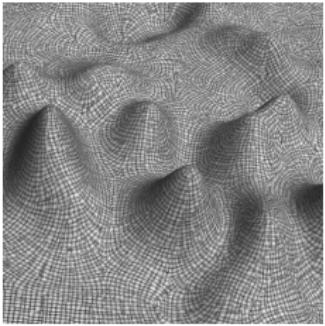




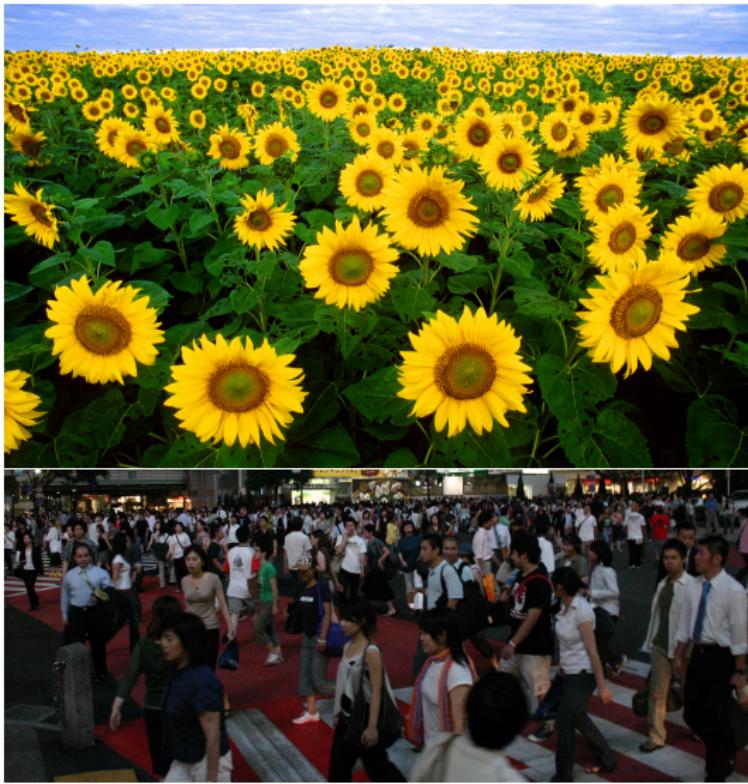






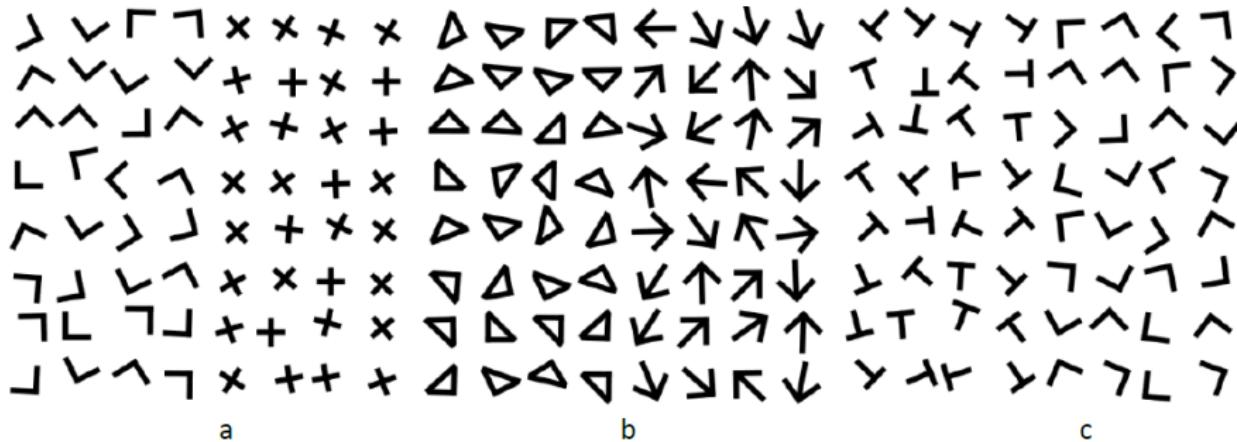


Texture et forme



Perception des textures

Notre système visuel a la capacité de grouper des éléments en régions homogènes en quelques centaines de millisecondes (vision pré-attentive)

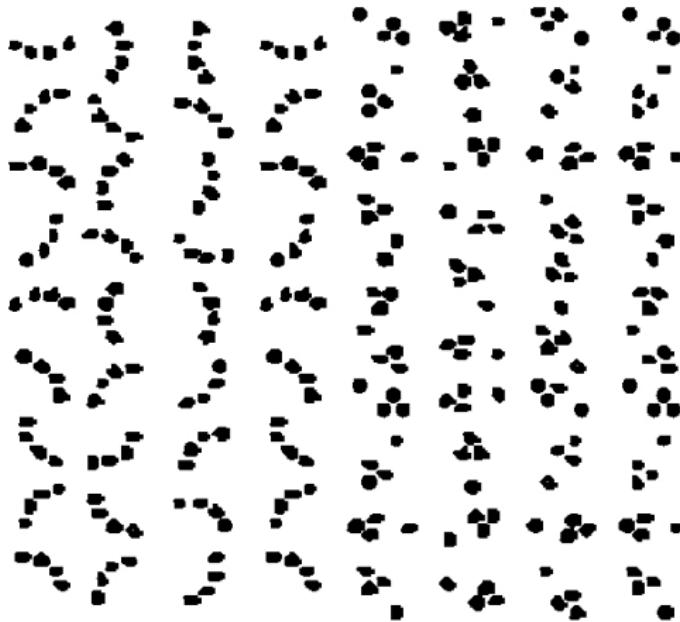


From Rosenholtz 2014

Problème : quelles sont les statistiques pertinentes pour expliquer la discrimination ?

- Beck 1973 : statistiques d'ordre 1 sur des “propriétés” (contraste, orientation, couleurs)
- Julesz 1973-75 : statistiques d'ordre 1 et 2 entre valeurs ponctuelles de niveau de gris
- Julesz 1981 : statistiques d'ordre 1 et 2 entre “textons” (éléments structurés, terminaisons, jonctions en “T”,...)

Problème : à quels objets mathématiques correspondent les textons ?



Contre-exemple à la première conjecture de Julesz

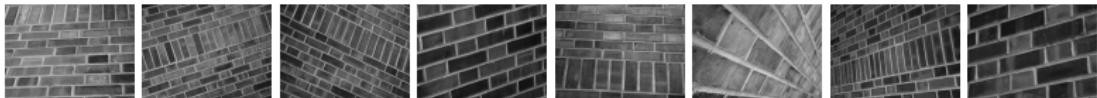
Première partie : analyse de texture

But :

$$I \rightarrow \{a_1(I), \dots, a_n(I)\}$$

où les a_i sont

- aussi compatibles que possible avec notre perception,
- éventuellement invariants à certaines transformations (géométriques et radiométriques)



Echantillons de textures de la base UIUCTex (Lazebnik *et al.* 2005)

Première partie : analyse de texture

- Difficultés

- Pas de modèle mathématique suffisamment général et manipulable
- Pas de notion satisfaisante de distance entre textures
- Notion fortement dépendante de l'échelle

- Applications

- Reconnaissance et classification de matériaux, contrôle de qualité
- Reconnaissance d'objets, segmentation
- Synthèse d'images
- Edition d'images (inpainting)
- Restauration d'images
- Reconstruction de volume ("shape from texture")

Stationnarité

- Un champs aléatoire Z est une collection de variables aléatoires $Z(z)$ indexées par $z \in \mathbb{Z}^2$
- On représente une texture par un champs aléatoire *stationnaire*
- Un champ aléatoire Z est dit *stationnaire* si pour tout $n \in \mathbb{N}$, $z \in \mathbb{Z}^2$ et $z_1, \dots, z_n \in \mathbb{Z}^2$
 $(Z(z_1), \dots, Z(z_n))$ et $(Z(z_1 + z), \dots, Z(z_n + z))$ ont même distribution jointe.
- En pratique, on dispose d'une image I définie sur un domaine limité Ω .
 - on peut considérer un champ périodique associé à l'image
 - ou une notion de stationnarité réduite aux déplacements inclus dans Ω

Les statistiques du premier ordre

On considère une image I définie sur $\Omega = [1, \dots, N]^2$ à valeurs dans $[1, \dots, G]$. Son histogramme

$$h(g) = \frac{1}{N^2} \sum_{z \in \Omega} \mathbb{1}_{I(z)=g}.$$

pour $g \in [1, \dots, G]$

est un estimateur de la distribution des valeurs de niveau de gris en tout pixel
Parmi les statistiques fréquemment utilisées pour décrire une texture :

- Moyenne $\mu(I) = \frac{1}{N^2} \sum_z I(z)$,
- Variance $\sigma^2(I) = \frac{1}{N^2} \sum_z (I(z) - \mu)^2$
- Energie $E(I) = \frac{1}{N^2} \sum_z I(z)^2$,
- Entropy $Ent(I) = - \sum_{g=1}^G f(g) \log(f(g))$, etc.

Statistiques du premier ordre

Limitation : toutes ces quantités ne dépendent que du seul histogramme de l'image.

Elles prennent des valeurs arbitraires sous l'effet d'un *changement de contraste*, c'est à dire une transformation

$$I \mapsto h \circ I,$$

avec h une fonction réelle croissante (voir le cours sur la radiométrie).

Néanmoins : la distribution des couleurs joue un rôle important dans notre perception des textures

Remarques sur l'estimation

- En général : on ne dispose que d'une seule réalisation du champ
→ l'estimation se fait par moyennage sur des zones suffisement grandes (hypothèse d'ergodicité)
- Dans le cas de scènes complexes : plusieurs zones ayant des textures différentes : l'hypothèse de stationnarité est fausse.
On estime les paramètres sur des voisinages bornés ("fenêtres glissantes").
Compromis précision de l'estimation - localisation



Statistiques du second ordre : la covariance

- La covariance d'un champ stationnaire Z (de moyenne nulle) est définie par $\mathcal{C}(z) = E(Z(0)Z(z))$.
- La covariance de l'image I est estimée, pour tout $z \in \Omega$, par

$$C(z) = \frac{1}{N^2} \sum_{z' \in \Omega} I(z')I(z + z').$$

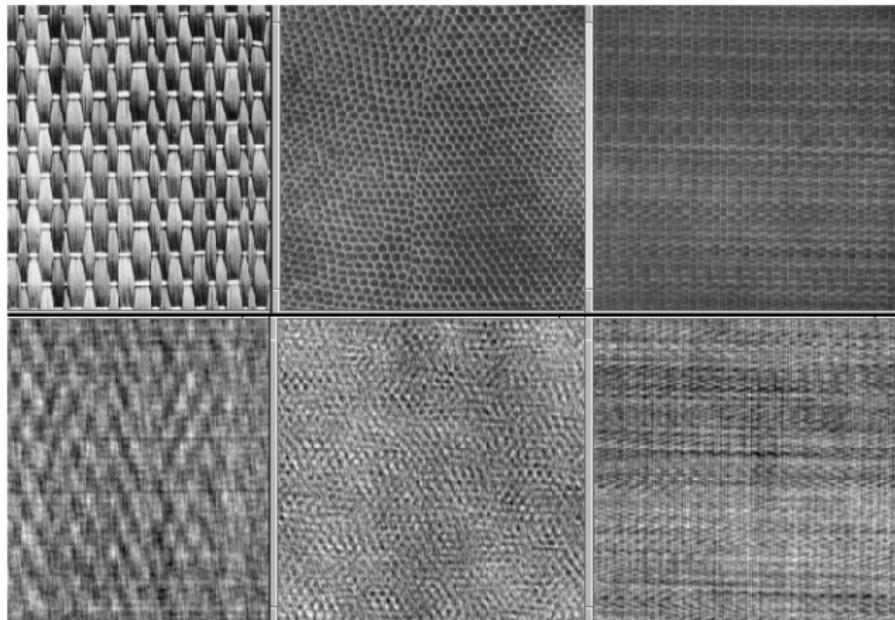
formule où l'on suppose I périodique.

- Théorème de Wiener-Khintchin (version discrète) : si \hat{I} désigne la transformée de Fourier discrète de I , alors

$$|\hat{I}(\omega)|^2 = \hat{C}(\omega).$$

La covariance

- Une large classe de textures ("micro-textures") sont bien représentées par leur covariance (donc par le module de la transformée de Fourier, le *spectre* de l'image).
- Plusieurs méthodes ont été proposées pour extraire l'information pertinente du spectre (e.g. Tang-Stewart 2000, Wang-Knoesen 2007).
- Pour tester cette assertion : méthode de synthèse.
 - On calcule la TFD de l'image I , $\{\hat{I}_{i,j}\}$,
 - Pour chaque i,j soit $\hat{J}_{i,j} = |\hat{I}_{i,j}| \exp(i\phi_{i,j})$, où $\phi_{i,j}$ est uniformément distribué entre 0 et 2π ,
 - J est l'image dont la TFD est $\{\hat{J}_{i,j}\}$.



Haut : images originales
Bas : synthèses

Matrices de cooccurrences (Haralick et al, 1973)

- Pour $(k, l) \in [1, \dots, N]^2$ et $(a, b) \in [1, \dots, G]^2$, la matrice de cooccurrence $M_{k,l}$ de I est définie par

$$M_{k,l}(a, b) = \frac{1}{(N - k)(N - l)} \sum_{i=1}^{N-k} \sum_{j=1}^{N-l} \mathbb{1}(I(i, j) = a \text{ et } I(i + k, j + l) = b).$$

- Estimateur de $\mathbb{P}(z = a, z' = b)$ (caractérise complètement le second ordre)
- Matrices de taille $G \times G$
- On considère généralement (k, l) petits devant la taille de l'image, et on quantifie l'image sur un nombre de niveaux $G' < G$ (typiquement 16).
- Pour → information locale, les matrices sont calculées sur des fenêtres.

Méthode ancienne et élémentaire ... mais efficace.

Matrices de cooccurrences (Haralick et al, 1973)

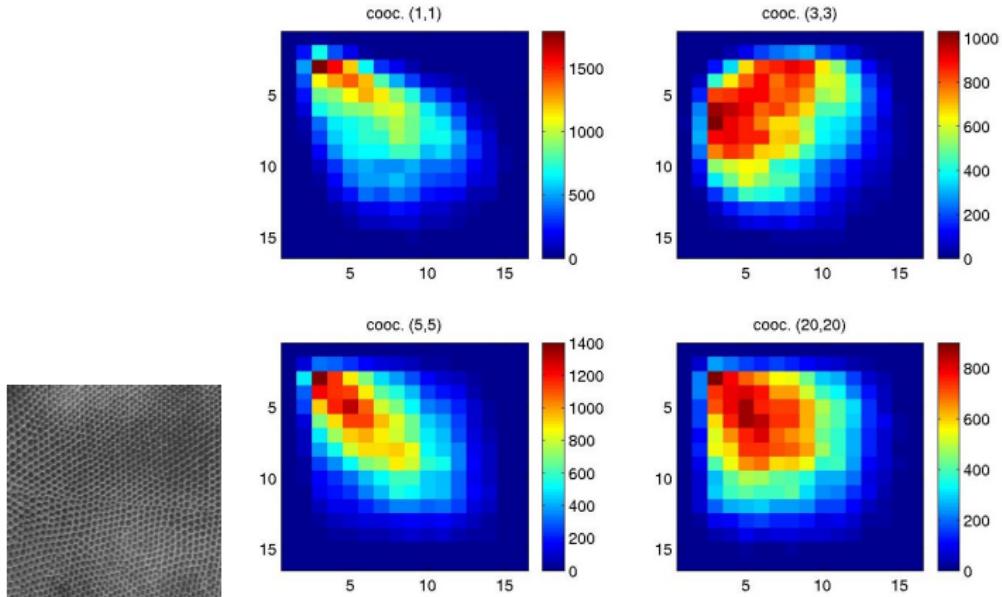
- Pour $(k, l) \in [1, \dots, N]^2$ et $(a, b) \in [1, \dots, G]^2$, la matrice de cooccurrence $M_{k,l}$ de I est définie par

$$M_{k,l}(a, b) = \frac{1}{(N - k)(N - l)} \sum_{i=1}^{N-k} \sum_{j=1}^{N-l} \mathbb{1}(I(i, j) = a \text{ et } I(i + k, j + l) = b).$$

- Estimateur de $\mathbb{P}(z = a, z' = b)$ (caractérise complètement le second ordre)
- Matrices de taille $G \times G$
- On considère généralement (k, l) petits devant la taille de l'image, et on quantifie l'image sur un nombre de niveaux $G' < G$ (typiquement 16).
- Pour → information locale, les matrices sont calculées sur des fenêtres.

Méthode ancienne et élémentaire ... mais efficace.

Exemples de matrices de cooccurrences



Matrices de cooccurrences

Pour réduire la dimension des attributs retenus, on considère des fonctionnelles sur les matrices de cooccurrence.

- Energie $E(k, l) = \sum_{a,b} M_{k,l}(a, b)^2$ (homogénéité),
- Entropie $Ent(k, l) = -\sum_{a,b} M_{k,l}(a, b) \log(M_{k,l}(a, b))$,
- Inertie $I(k, l) = \sum_{a,b} (a - b)^2 M_{k,l}(a, b)$ (dispersion), etc.

Eventuellement → invariance par rotation en moyennant sur plusieurs orientations.

Champs de Markov

- A venir : cours de F. Tupin (IMA203)
- Le champ Z est dit markovien si la distribution de $Z(i,j)$ est complètement caractérisée par ses voisins (sur un voisinage prescrit).
- Les textures sont supposées markoviennes et peuvent être représentées par des champs de Gibbs, c'est à dire que

$$Pr(Z = z) = \frac{1}{K} \exp \left(- \sum_{c \in C} U_c(z) \right),$$

où C est un système de "cliques", c'est à dire de pixels voisins selon un type de voisinage.

- Exemple : voisinage en 8-connectivité et cliques d'ordre 2 ("auto-modèle") :

$$Pr(Z = z) = \frac{1}{K} \exp \left(C + a_0 \sum_{(i,j)} z_{i,j} + \sum_{k=1}^8 a_k \sum_{i,j} z_{(i,j)} z_{(i,j)+\gamma_k} \right),$$

avec $\gamma_k = (0, 1), (1, 0), \dots$

- Problèmes principaux : trouver l'ordre du modèle (des cliques) et les potentiels U_c .

Morphologie mathématique

Idée :

on analyse une texture en la simplifiant graduellement par des opérations correspondant à des éléments structurants de plus en plus grands

Outil d'analyse de textures : les granulométries

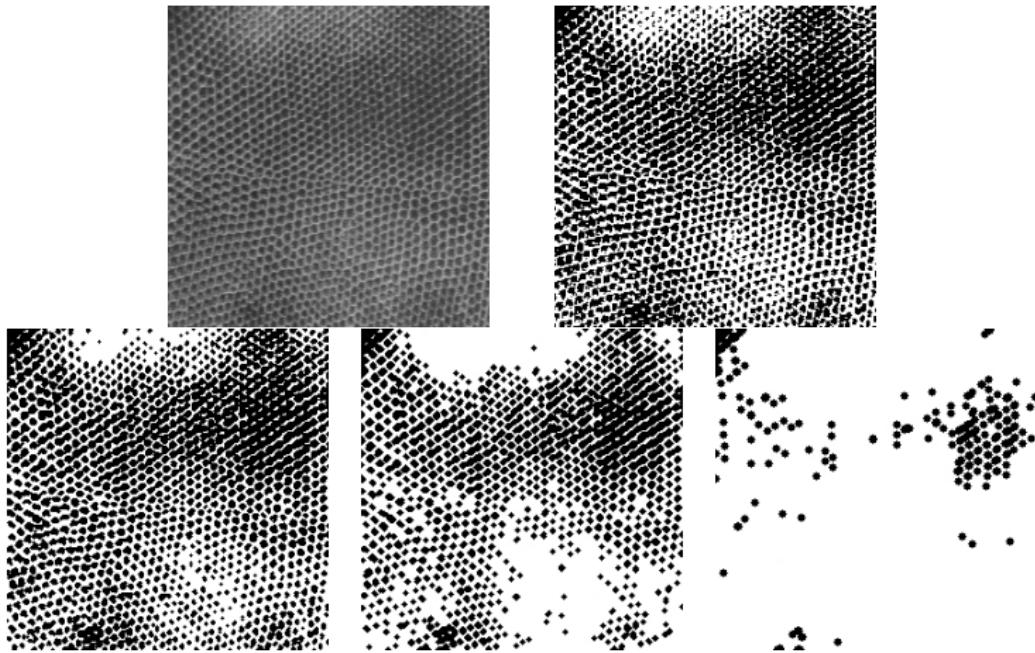
- Pour une texture binaire I on considère la suite des ouvertures $O_{rA}(I)$, où les rA sont des éléments structurants dilatés du facteur r (cf cours de morphologie mathématique).
- La courbe de granulométrie de I est le graphe de la fonction

$$r \rightarrow \nu(O_{rA}(I) = 1))$$

où $\nu(A)$ est la surface de A .

- On caractérise généralement la texture par la dérivée des courbes de granulométrie pour différents éléments structurants.
- Pour une texture à niveau de gris, on considère différents ensembles de niveau

$$\chi_\lambda(I) = \{(i,j) : I(i,j) \geq \lambda\}.$$



Filtres de Gabor

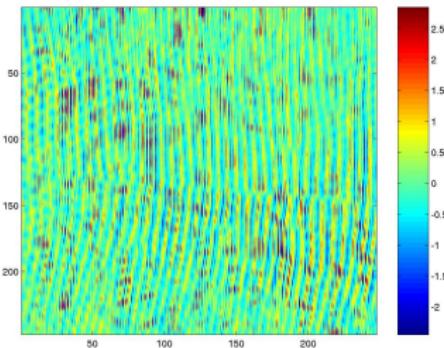
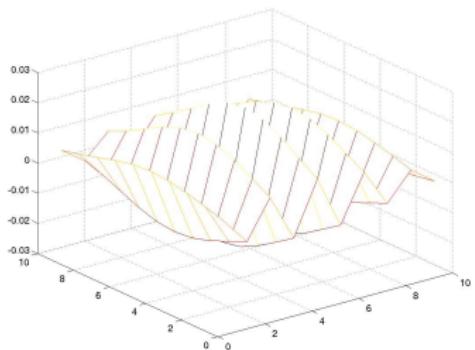
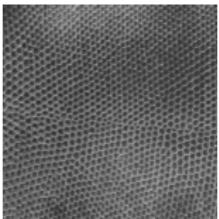
- Les textures sont représentées par leurs réponses à des filtres linéaires extrayant une information fréquentielle localisée et orientée.
- Filtres de Gabor horizontaux :

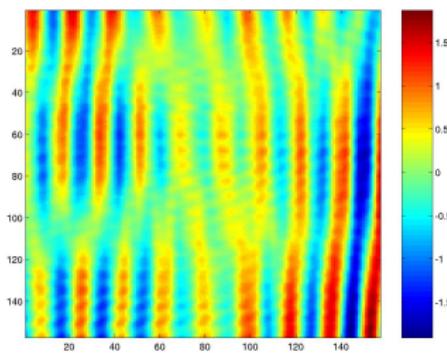
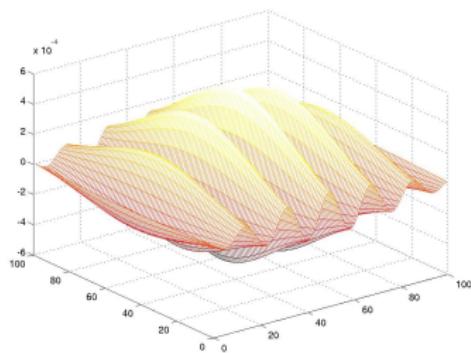
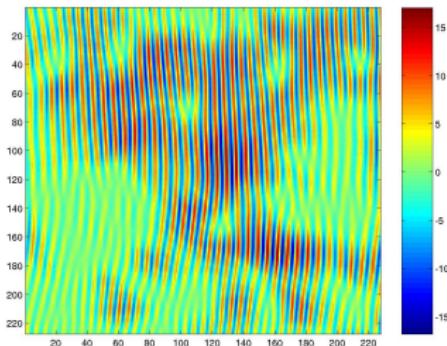
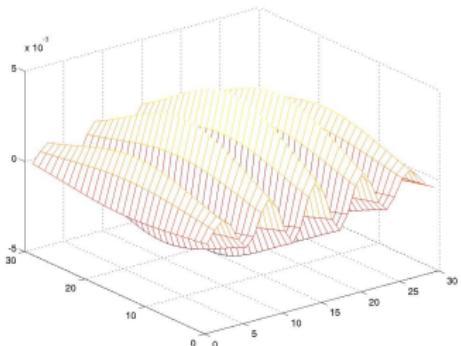
$$\phi(x, y) = e^{-\frac{1}{2}\left(\frac{x^2+y^2}{\sigma}\right)} \cos(2\pi f x),$$

$$\phi(x, y) = e^{-\frac{1}{2}\left(\frac{x^2+y^2}{\sigma}\right)} \sin(2\pi f x),$$

- Les autres filtres s'obtiennent par rotations de θ .
- La texture est caractérisée par différentes statistiques sur les réponses aux filtres (classiquement : l'énergie).
- Ordres de grandeur : 5-10 orientations et 5-10 fréquences.
- Variante : gaussienne non-isotrope

Très nombreuses variantes avec d'autres représentations espace-fréquence





Filtres de Gabor et réponses

Décomposition sur des bases d'ondelettes

- On utilise en général des trames d'ondelettes redondantes, comportant plus de 3 orientations et invariantes par translation.
- Comme précédemment on peut retenir l'énergie comprise dans chaque bande fréquentielle.
- Une solution alternative : modélisation paramétrique des coefficients. Par exemple, si $c_{k,l,\eta}$ désigne le coefficient à la position k , échelle l et orientation η , on peut supposer qu'à l et η fixés, les $\{c_{k,l,\eta}\}_k$ sont distribués selon

$$c_{k,l,\eta} \sim Ce^{-\left(\frac{|x|}{\beta}\right)^\alpha}.$$

- Les textures sont alors représentées par les coefficients α et β (pour chaque l et η).

Analyse de textures et invariances

- Représentation des textures

$$I \rightarrow \{a_1(I), \dots, a_n(I)\}$$

- En fonction de l'application, on peut imposer des invariances

- géométriques

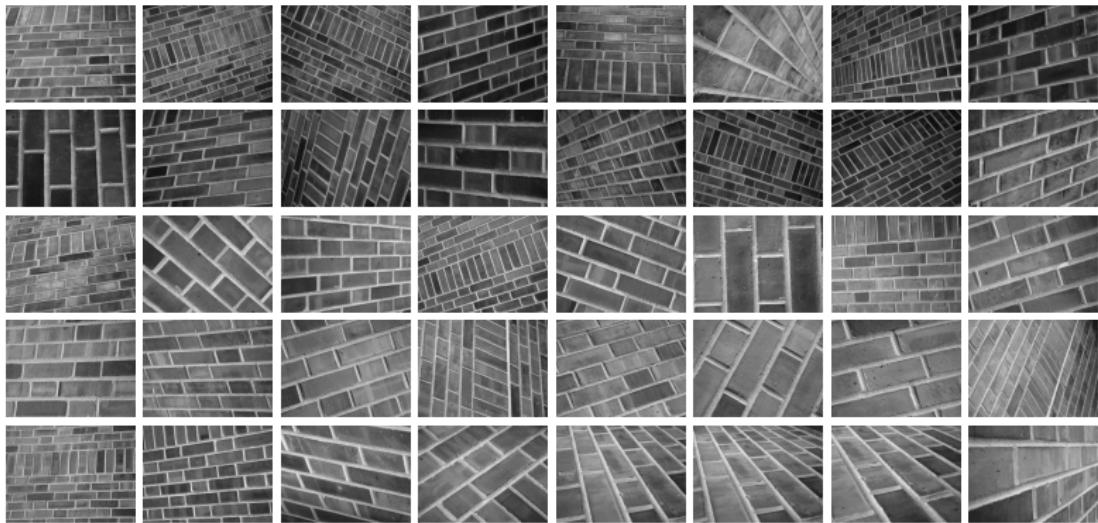
$$a_i(I \circ T) = a_i(I),$$

où T est une transformation géométrique (rotation, changement d'échelle, transformation affine, transformation projective, etc.)

- radiométriques

$$a_i(f \circ I) = a_i(I),$$

où f est une fonction croissante (changement de contraste).

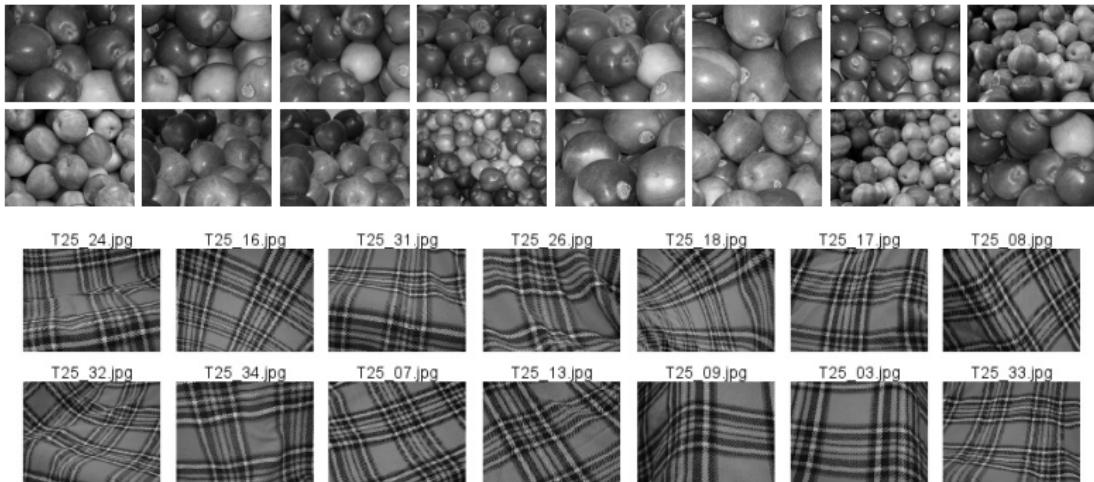


Echantillons de textures de la base UIUCTex (Lazebnik *et al.* 2005)

Analyse de texture et invariances

- Statistiques ponctuelles (covariances, co-occurrences, champs de Markov)
 - Rotations : moyennes des réponses à différentes orientations
 - Autres transformations (en particulier changements d'échelle) : difficile
- Filtrage (Gabor, ondelettes)
 - Transformations géométriques (rotations, échelle) : utilisation de familles redondantes
 - Changements de contraste linéaires : normalisation des réponses

Textures non-planes, non-rigides



T25_24.jpg

T25_16.jpg

T25_31.jpg

T25_26.jpg

T25_18.jpg

T25_17.jpg

T25_08.jpg

T25_32.jpg

T25_34.jpg

T25_07.jpg

T25_13.jpg

T25_09.jpg

T25_03.jpg

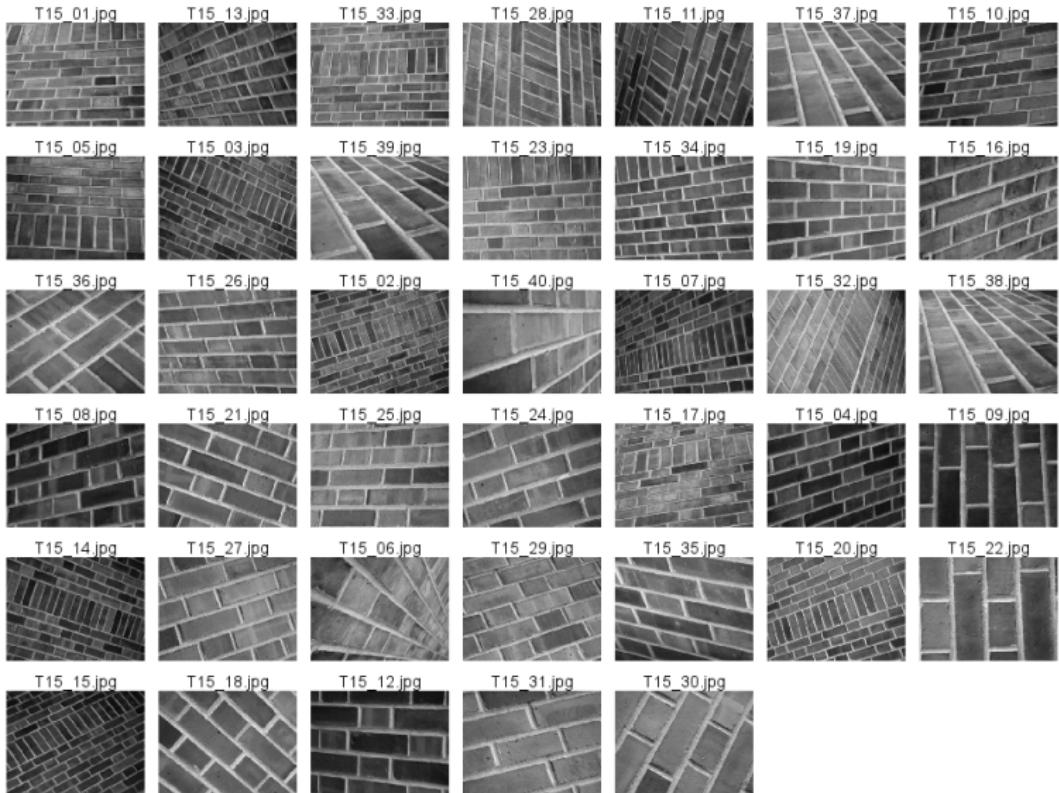
T25_33.jpg

- Invariance affine ou projective insuffisante
- Approches possibles :
 - Apprentissage
 - Invariance aux déformations géométriques (similitudes ou affinités) **locales** (Lazebnik et al. 05, Mellor et al. 08, Xia et al. 10, Nguyen et al. 2011)

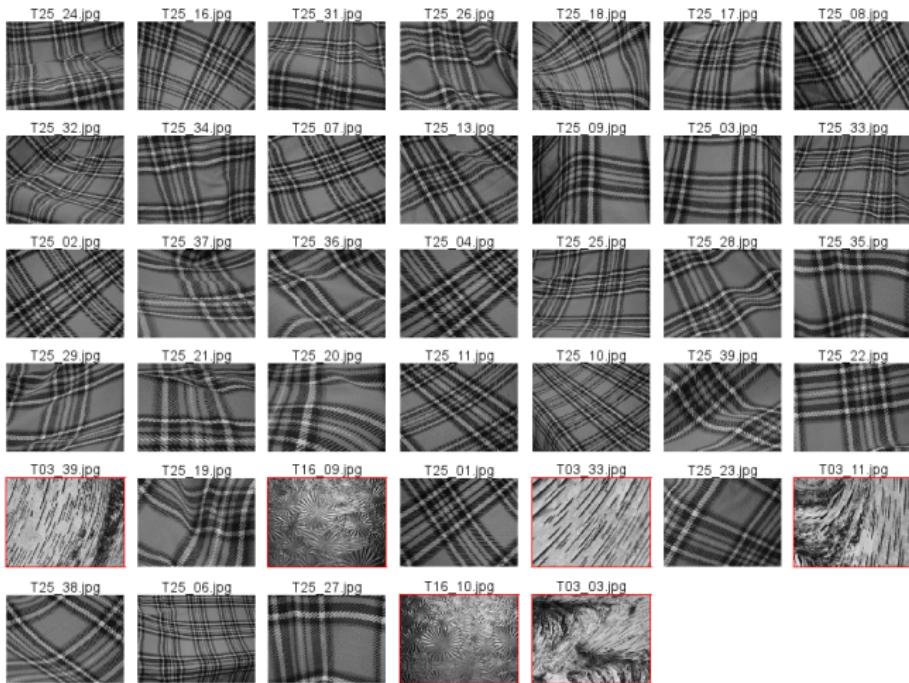
Utilisation des descripteurs SIFTs (Lazebnik et al. 2005)

- Extraction de points clés et de descripteurs SIFTs
- Construction d'un dictionnaire par classification non-supervisée (k-means)
- Calcul des fréquences d'utilisation des mots (sacs de mots)

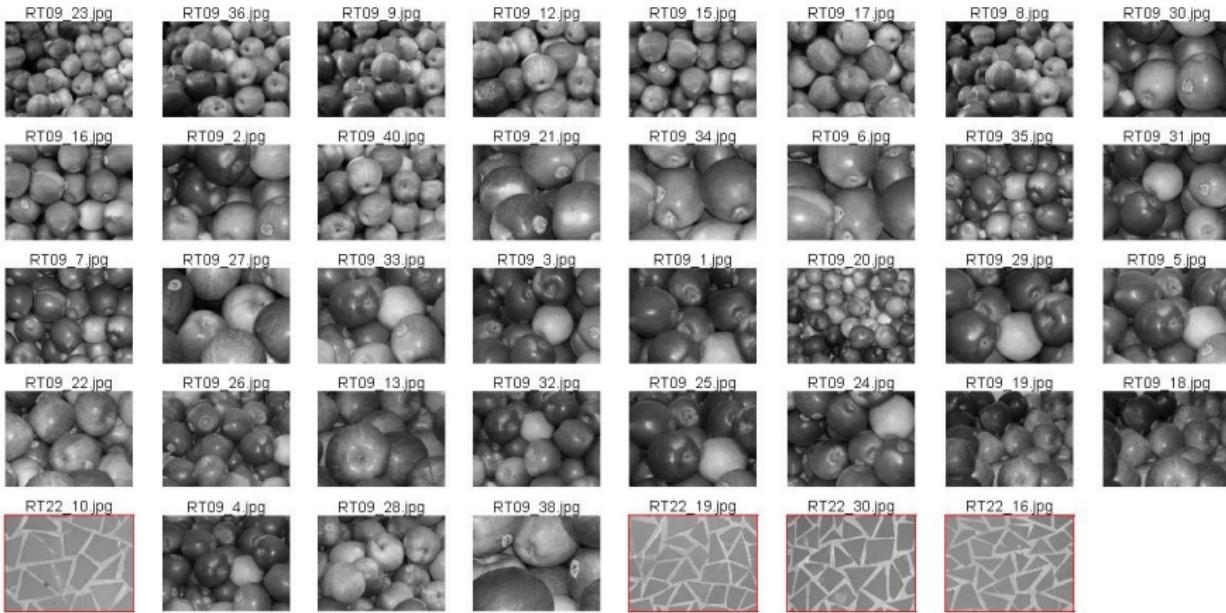
L'invariance à des transformations complexes est obtenue par invariances à des transformations plus simples (similitudes, affines) *localement*.



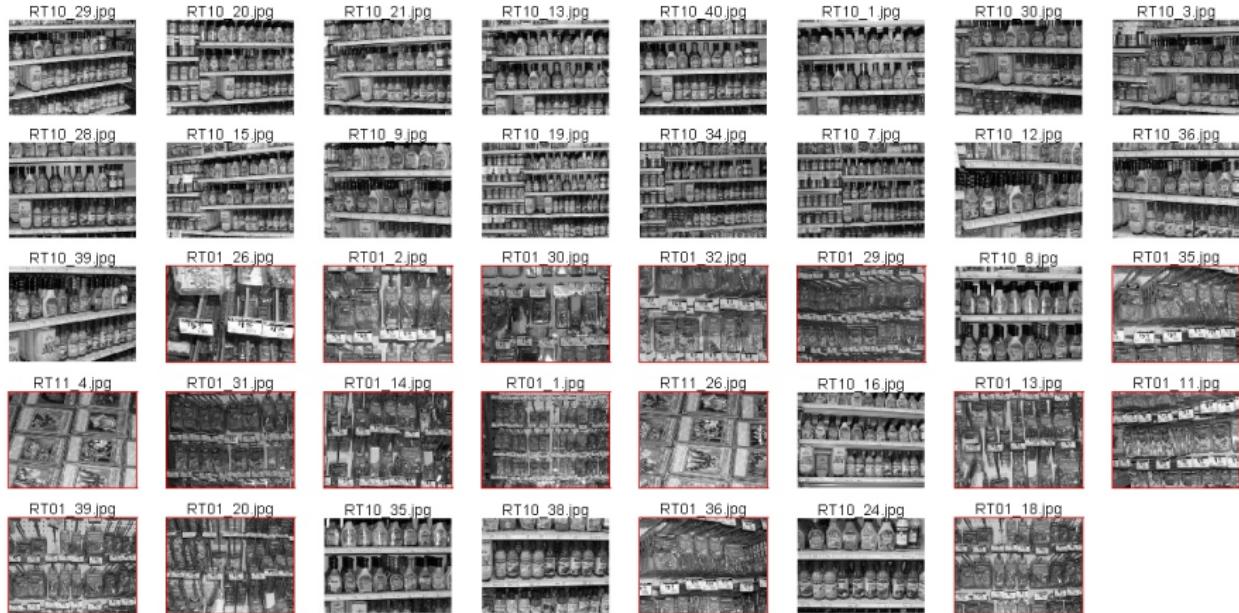
Expérience de recherche par le contenu (25 classes de 40 échantillons)



Expérience de recherche par le contenu



Experience de recherche par le contenu



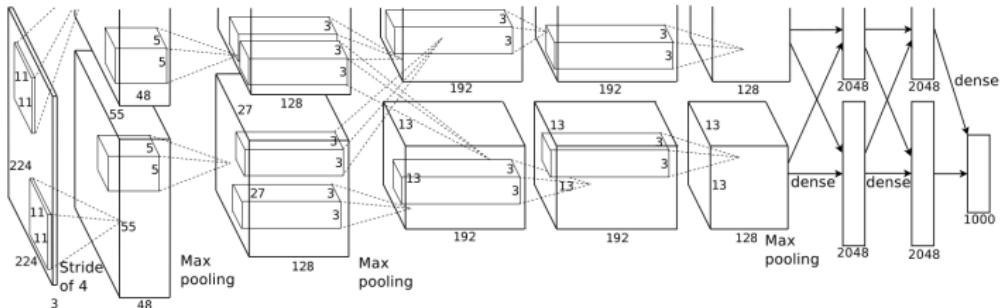
Expérience de recherche par le contenu

Reconnaissance par réseaux de neurones

- Tendance actuelle pour l'analyse d'image dans un sens large
- Repose sur des réseaux **profonds et convolutionnels** (CNNs convolutional neural networks)
- Permet la reconnaissance de textures et de matériaux en environnement non-contrôlés (“in the wild”).
- Nécessite de très grandes quantités d’images **annotées**
- cours IMA205 à venir

Reconnaissance par réseaux de neurones

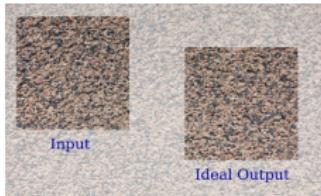
- Idée : utiliser les couches convolutionnelles d'un CNN comme un banc de filtres (Cimpoi et al. 2015)
- Repose sur un réseau entraîné sur environ 1M d'images organisées en environ 1000 classes (ImageNet, Krizhevsky et al., 2012)



- Puis apprentissage des textures à partir de 50000 régions, issues de 10000 images, pour 23 classes : statistiques des réponses aux filtres (par exemple en utilisant des sacs de mots comme précédemment, ou d'autres représentations, e.g. des vecteurs de Fisher)

Synthèse de textures

- **But** : partant d'une image de texture A , synthétiser des images ponctuellement différentes de A , mais qui produisent la même impression visuelle.
- **Applications** :
 - Graphisme
 - Amélioration, modification d'images
 - Psychophysique
 - Test de la pertinence des modèles
- **Difficultés** :
 - Pas de modèle mathématique complètement satisfaisant des textures
 - Pas de critère quantitatif d'évaluation des résultats



III. B. Galerne



Disparition du commissaire Yezhov (image www.newseum.org)

Quelques méthodes ad'hoc

- Stockage d'une image de texture, puis plaquage de la texture avec des conditions aux bords → problèmes topologiques.
- Perturbation des normales (Blinn 78) :

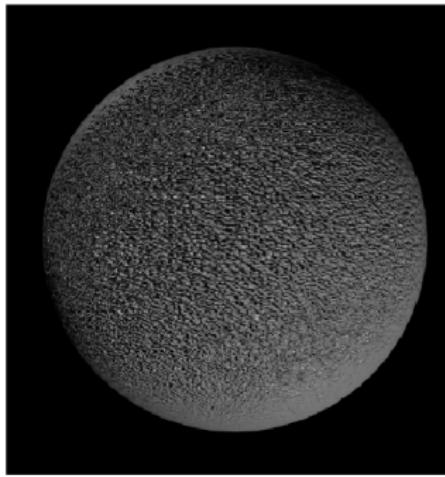
à partir d'une surface, on calcule en chaque point une direction normale fictive, après perturbation de la surface, puis on utilise cette nouvelle normale dans un modèle de reflection :

$$\mathbf{M}(x, y) \rightarrow \mathbf{M}(x, y) + F(x, y)\mathbf{n}(x, y),$$

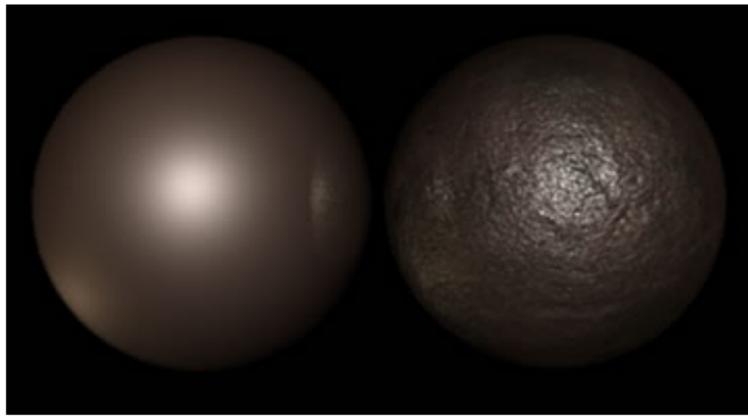
ce qui implique

$$\mathbf{n} \rightarrow \mathbf{n} + \frac{\partial \mathbf{M}}{\partial x} \wedge \mathbf{n} \frac{\partial F}{\partial y} - \frac{\partial \mathbf{M}}{\partial y} \wedge \mathbf{n} \frac{\partial F}{\partial x}$$

c'est donc la fonction F qui modélise physiquement la texture : table de valeurs, valeurs aléatoires filtrées ...



Perturbation des normales d'une sphère



Perturbation des normales d'une sphère
Illustration de <http://www.interstation3d.com>

- Reaction-diffusion :

modèle initialement proposé par Turing (1952) pour les motifs apparaissant sur la peau de certains animaux. Un motif est le résultat de la concentration en morphogènes, qui apparaissent, diffusent, et disparaissent au cours du temps.

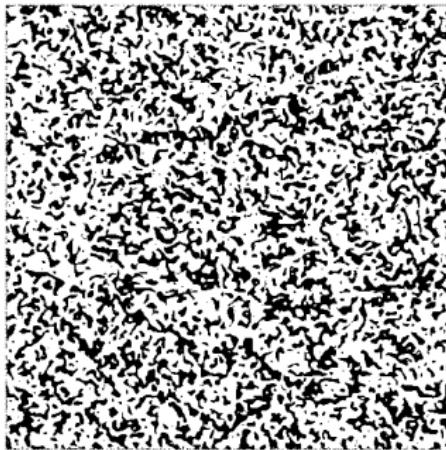
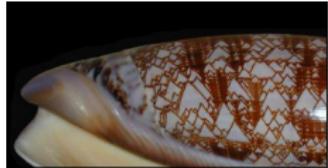
$$\frac{\partial C}{\partial t} = a^2 \Delta C - bC + R.$$

Pour modéliser l'anisotropie, ΔC devient

$$a_1 \frac{\partial^2 C}{\partial x^2} + a_2 \frac{\partial^2 C}{\partial x \partial y} + a_3 \frac{\partial^2 C}{\partial y^2}$$



All the same, but different



Reaction-diffusion

Modèles “fractals”

Essentiellement des simulations de généralisations du mouvement brownien : les mouvements browniens fractionnaires (Mandelbrott-Van Ness 68).

- dimension 1 : V_H processus tel que

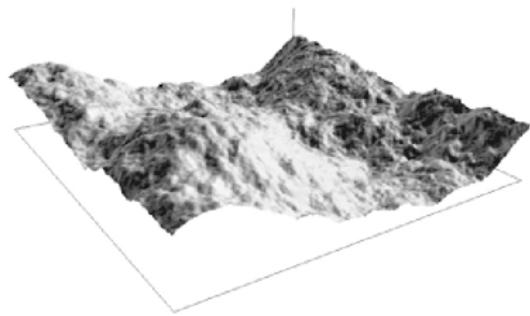
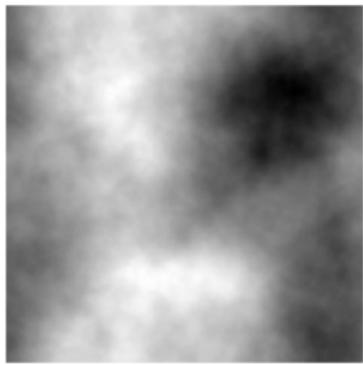
$$V_H(t_1) - V_H(t_2) \sim N(0, |t_2 - t_1|^{2H}).$$

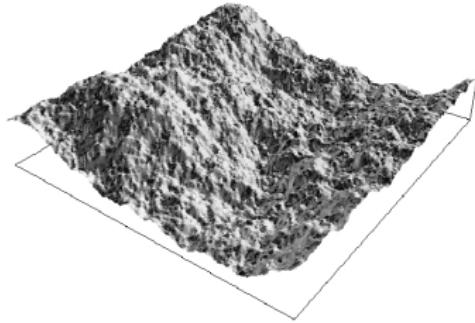
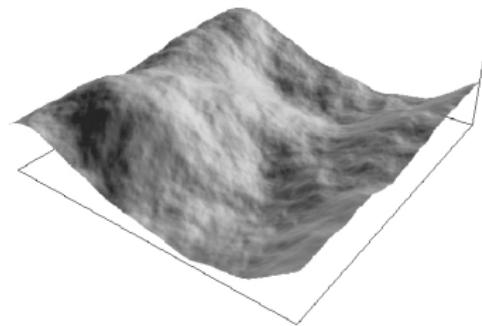
Remarque :

$V_H(t_0) - V_H(t_0 + t)$ et $r^{2-H} (V_H(t_0) - V_H(t_0 + rt))$ ont alors même distribution.

- dimension 2 : $V_H : \mathbb{R}^2 \rightarrow \mathbb{R}$ tel que

$$V_H(x_1) - V_H(x_2) \sim N(0, ||x_1 - x_2||^{2H}).$$







<http://www.decarpentier.nl>

Modèles génératifs

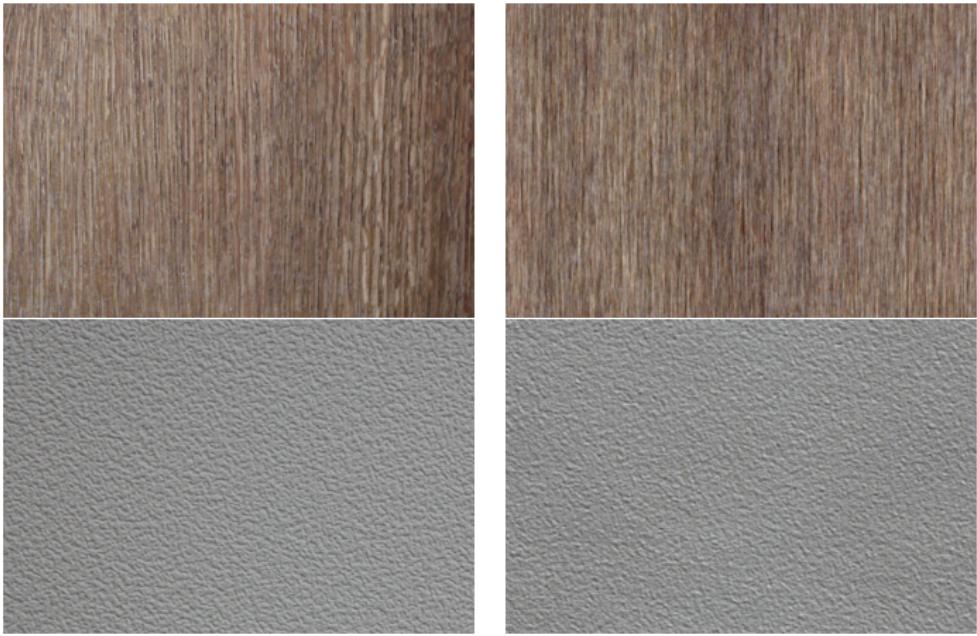
Deux grandes familles de méthodes :

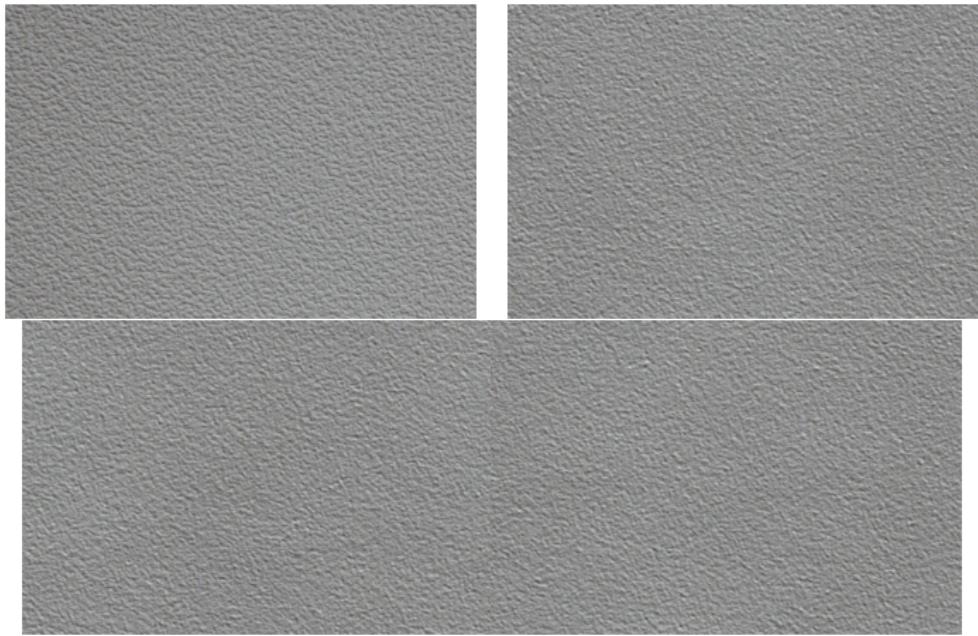
- Réalisations de modèles aléatoires avec des caractéristiques statistiques inspirées des textures naturelles
→ projections sur des ensembles d'images définis par des contraintes
- Images obtenus par ré-échantillonnage de voisinages locaux (patches)

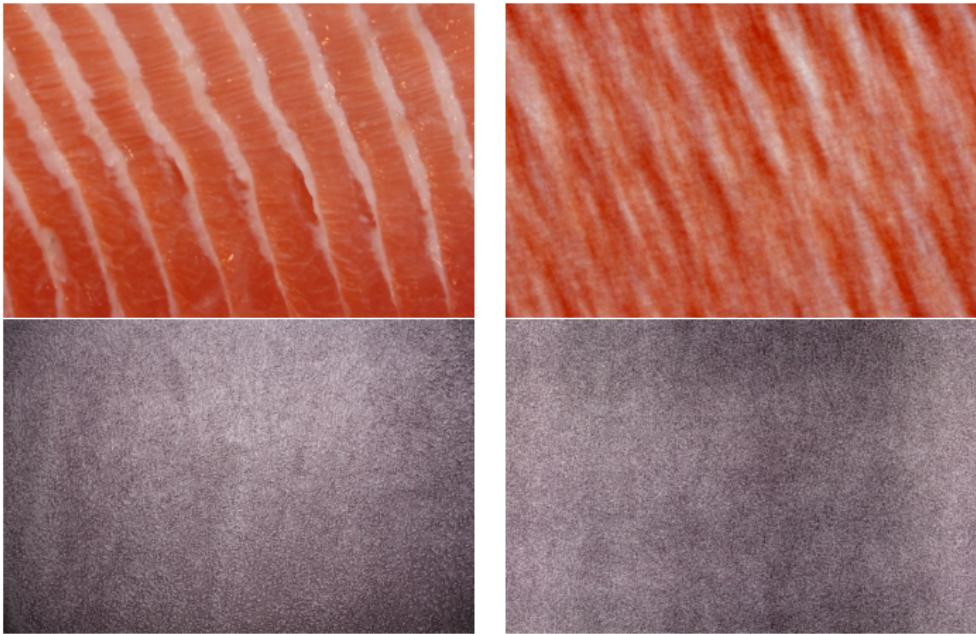
Synthèse par contraintes

Synthèse par phases aléatoires (Galerne et al. 2011)

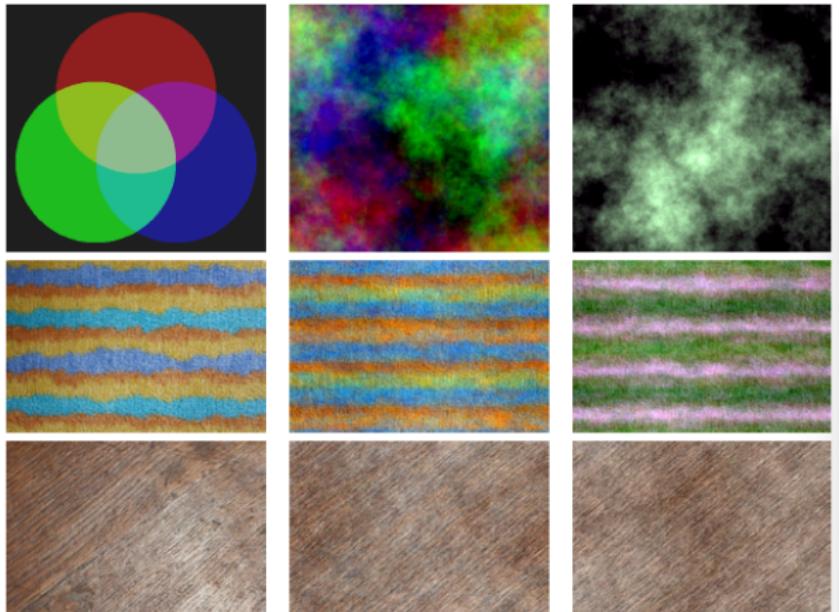
- On synthétise des textures en imposant aux images d'avoir le même spectre (module de la transformée de Fourier) que l'image exemple
- Une solution : imposer des phases aléatoires à la transformée de Fourier de l'exemple
- On impose une cohérence de phase entre canaux couleurs
- Suppression des artefacts de non-périodicité (recours à une décomposition en composantes périodiques et régulières)
- Phases aléatoires : équivalent à la projection orthogonale sur un ensemble d'images à spectre prescrit
- → Démo en ligne sur *ipol* (*random phase demo* sous google)







Corrélations couleur



Spot h

BPA

“Mauvais” BPA

$$\hat{h} = \begin{pmatrix} |\hat{h}_R| e^{i\varphi_R} \\ |\hat{h}_G| e^{i\varphi_G} \\ |\hat{h}_B| e^{i\varphi_B} \end{pmatrix}$$

$$\hat{Z} = \begin{pmatrix} |\hat{h}_R| e^{i(\varphi_R + \theta)} \\ |\hat{h}_G| e^{i(\varphi_G + \theta)} \\ |\hat{h}_B| e^{i(\varphi_B + \theta)} \end{pmatrix}$$

$$\hat{Z}_M = \begin{pmatrix} |\hat{h}_R| e^{i\theta} \\ |\hat{h}_G| e^{i\theta} \\ |\hat{h}_B| e^{i\theta} \end{pmatrix}$$

On ajoute la même phase aléatoire à chaque canal
→ permet de conserver les corrélations entre canaux

Les champs markoviens

Représentation sous forme de champs de Gibbs avec énergie, inférence des paramètres, échantillonnage de Gibbs.

Difficultés : inférence de l'énergie - structures multi-échelles

Méthodes par ondelettes : ajustement d'histogrammes (Heeger-Bergen 1995)

Pour synthétiser une texture B partant d'un échantillon A :

- Calcul d'une décomposition en ondelettes $\{a_{i,j,k}\}$ de A : i orientation, j échelle, k position.
 - On part de B bruit blanc, et le décompose en $\{b_{i,j,k}\}$.
 - Pour chaque i,j , on modifie l'histogramme de $\{b_{i,j,k}\}$ pour le rendre égal à celui de $\{a_{i,j,k}\}$.
 - On reconstruit l'image B .
 - On modifie l'histogramme (de couleurs) de B pour le rendre égal à celui de A
 - On itère les trois étapes précédentes.
- Démo en ligne (*heeger ipol sous google*)

Ajustement d'histogramme

Pour deux images A et B (à valeurs dans $[0,1]$), on considère le changement de contraste

$$\phi_{BA} : [0, 1] \rightarrow [0, 1],$$

tel que

$$h(\phi_{BA} \circ B) = h(A),$$

où $h(I)$ est l'histogramme de I . L'ajustement de contraste de B selon A est défini par

$$A \rightarrow \phi_{BA} \circ B.$$

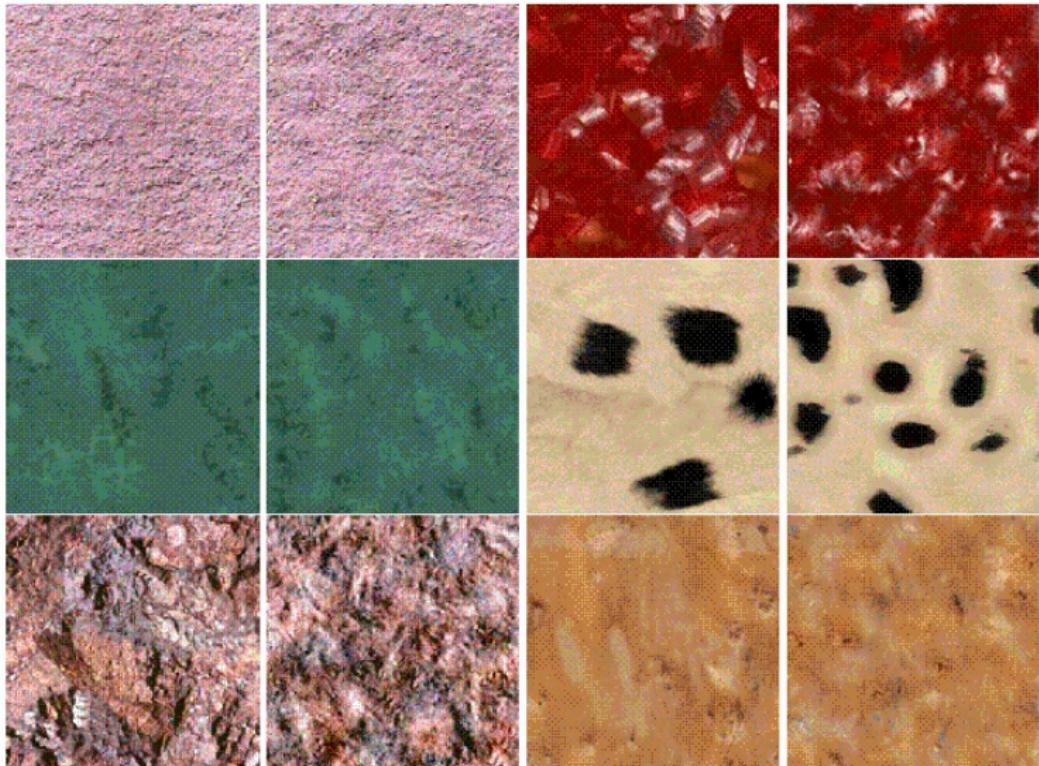
(cf cours radiométrie)

Si on note $H_I(x) = \int_0^x h(I)(t) dt$, on a (si H_A est inversible)

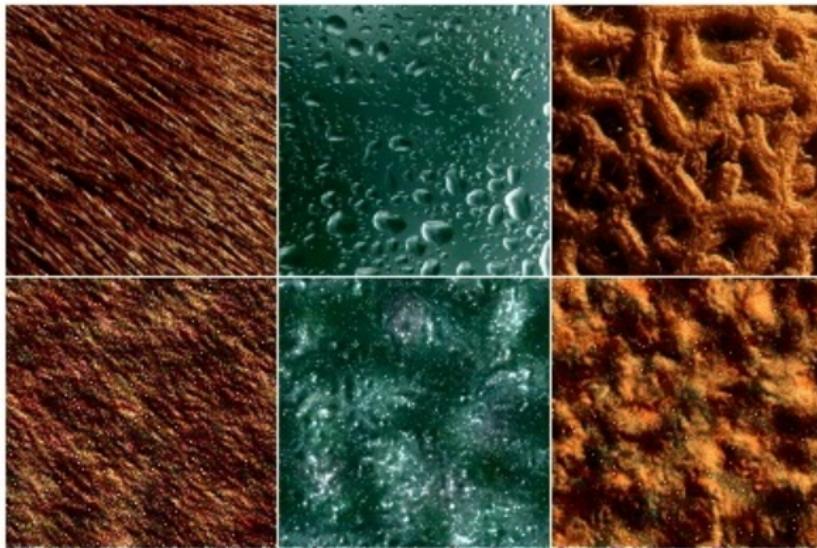
$$\phi_{BA} = H_A^{-1} \circ H_B$$

Remarques :

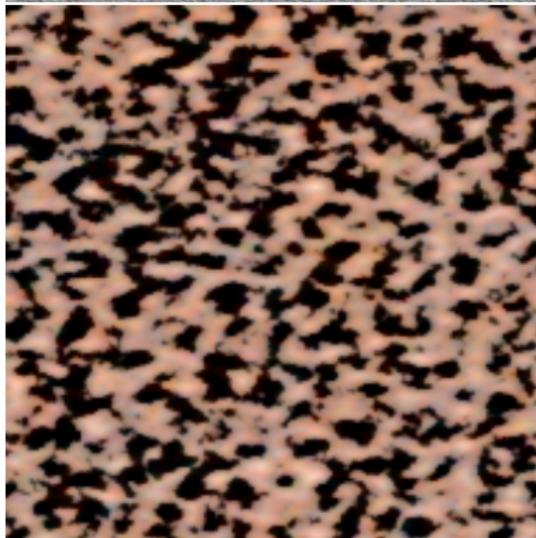
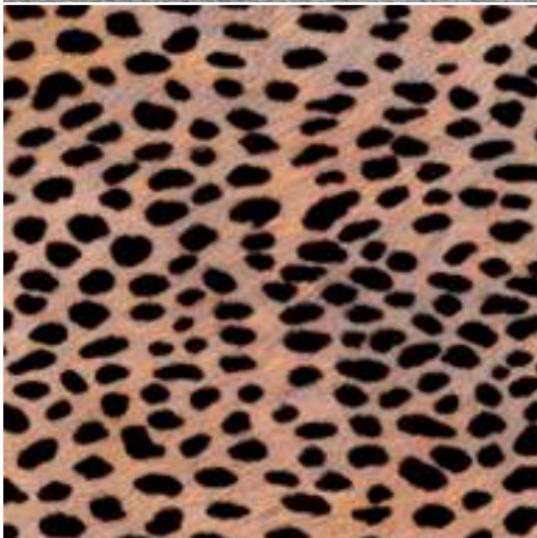
- En pratique, on utilise pas une décomposition sur une base, mais sur une "trame", pour disposer de plus de 3 orientations.
- Cette méthode s'inspire d'observations psychophysiques sur l'importance des reponses à des familles de filtres pour la discrimination des textures,
- Les contours ne sont pas reproduits, car ils correspondent à des relations inter-échelles.



Ajustement des marginales de decompositions en ondelettes. Extrait de Heeger-Bergen 1995



Ajustement des marginales de decompositions en ondelettes. Extrait de Heeger-Bergen 1995



Idem (Démo ipol)

Méthodes par ondelettes : ajustement des statistiques d'ordre supérieur (Simoncelli-Portilla 1998, 2000)

Pour synthétiser une nouvelle image B à partir d'un échantillon A :

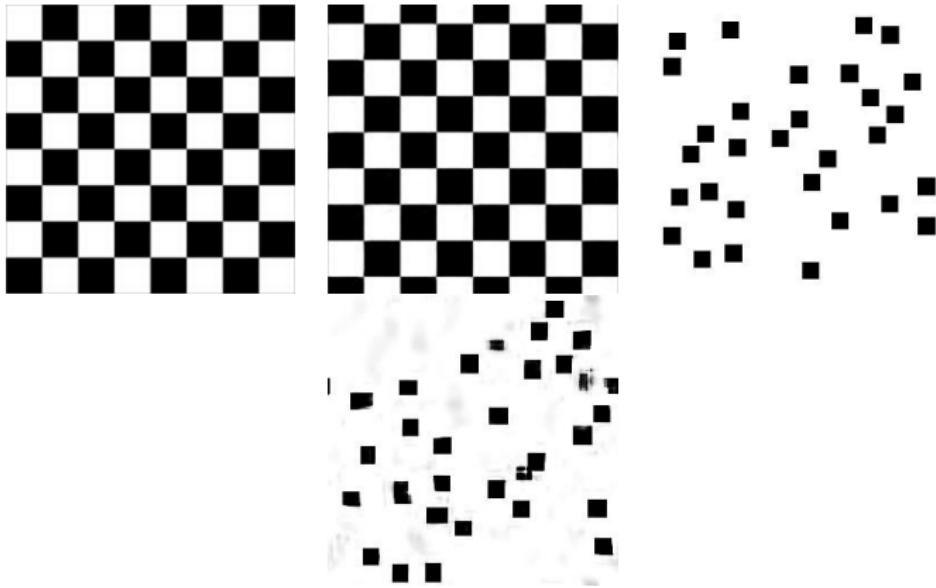
- Calcul d'une décomposition en ondelettes $\{a_{i,j,k}\}$ de A (i : orientation, j : échelle, k : position).
- On part de B égale à un bruit blanc, et le décompose en $\{b_{i,j,k}\}$.
- On projette les $\{b_{i,j,k}\}$ sur des sous-espaces d'éléments ayant les mêmes statistiques (à préciser) que $\{a_{i,j,k}\}$.
- on reconstruit l'image B .

Typiquement, ces statistiques incluent :

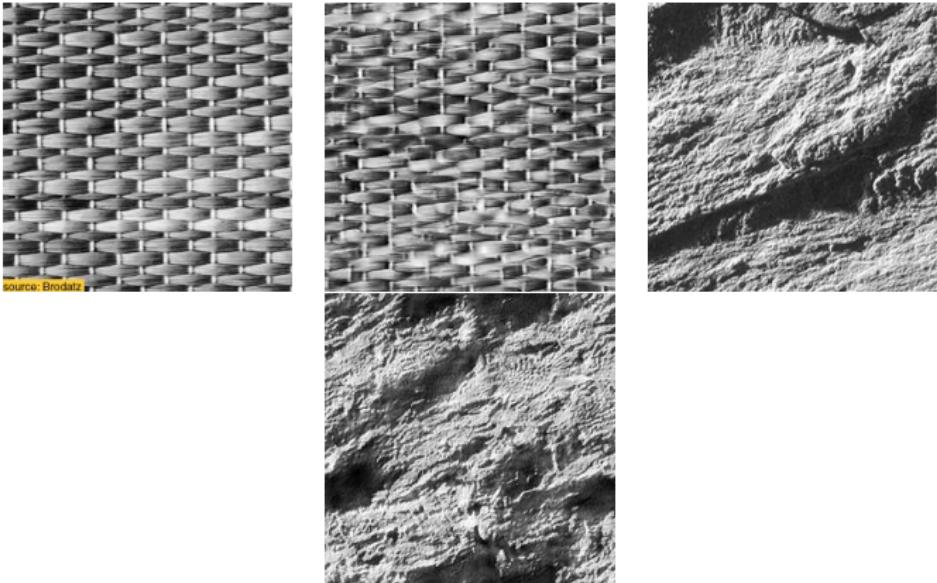
- Moyenne, variance, kurtosis, etc... de B .
- Covariances de $\{|b_{i,j,k}|\}$ et $\{|b_{i,j,k+p}|\}$, pour i,j,p donnés.
- Covariances de $\{|b_{i_1,j_1,k}|\}$ et $\{|b_{i_2,j_2,k}|\}$, pour i_1,i_2,j_1,j_2 donnés.

Ordres de grandeur : quelques orientations (typiquement 4 à 6) ; p de l'ordre de la dizaine, ce qui donne de l'ordre de 1000 paramètres.

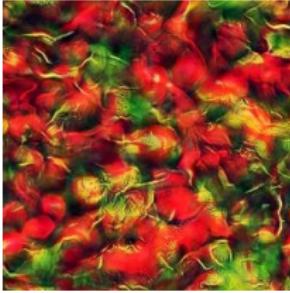
L'image B est reconstruite à partir de coefficients $\{|b_{i,j,k}|\}$ obtenues par projection sur des sous-espaces.



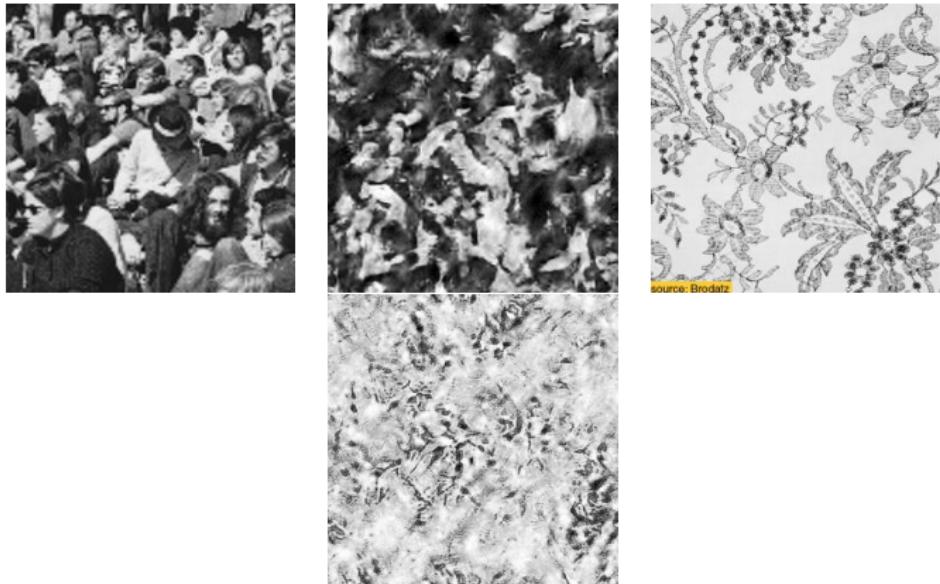
Ajustement de statistiques d'ordres supérieurs de décompositions en ondelettes. Extrait de Portilla-Simoncelli 2000



Ajustement de statistiques d'ordres supérieurs de decompositions en ondelettes. Extrait de Portilla-Simoncelli 2000



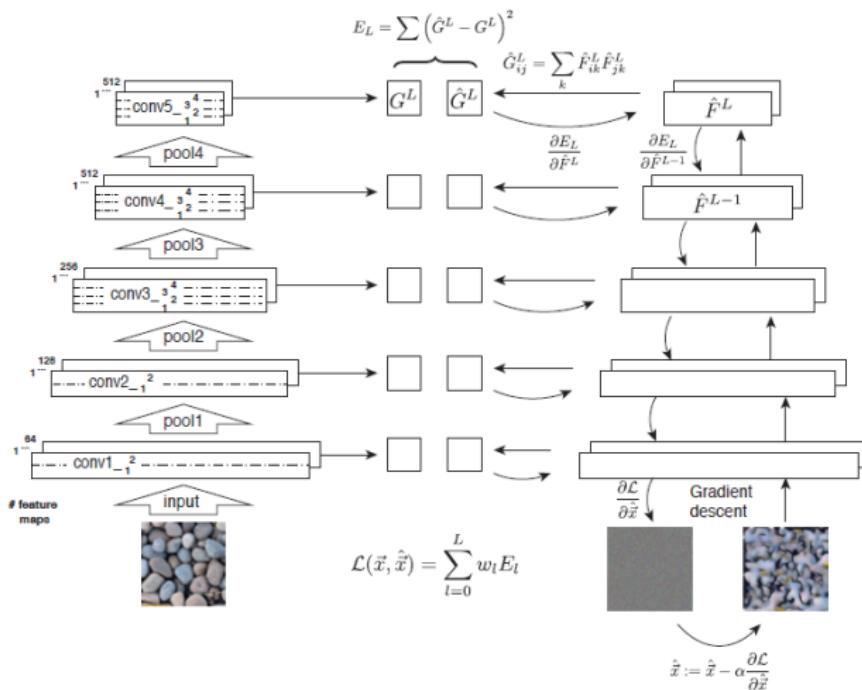
Ajustement de statistiques d'ordres supérieurs de decompositions en ondelettes. Extrait de Portilla-Simoncelli 2000



Ajustement de statistiques d'ordres supérieurs de decompositions en ondelettes. Extrait de Portilla-Simoncelli 2000

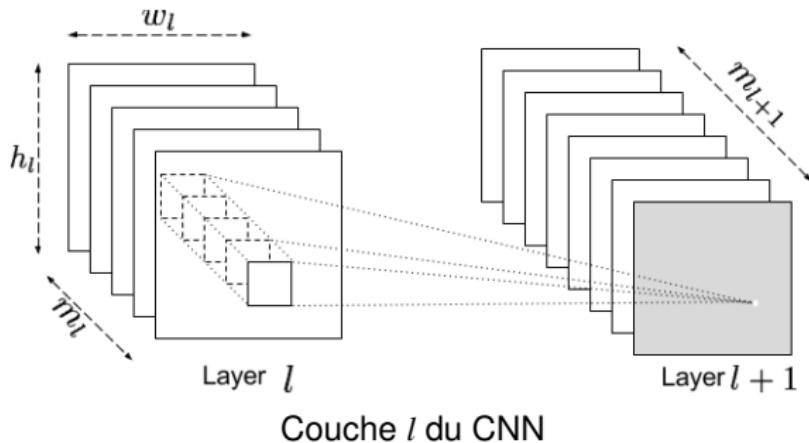
Réseaux profonds convolutionnels

(Gatys-Ecker-Bethge 2015)



Idée : imposer les corrélations entre les différents filtres à la même position

Synthèse par CNN



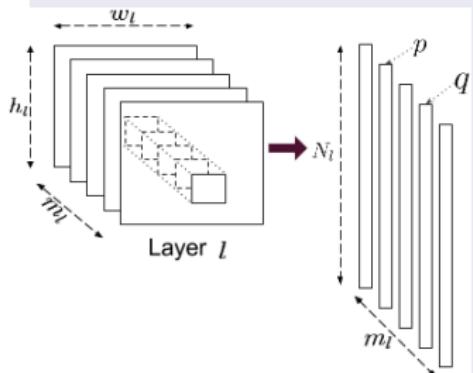
Attributs de texture (features)

Pour une texture exemple I la carte d'attributs (feature map) à la couche l est
 $f^l \in \mathbb{R}^{m_l \times N_l}$.

m_l : nombre de filtres, $N_l = w_l \cdot h_l$, w_l, h_l taille de la carte d'attributs.

Modèle de texture

Inspiré des méthodes par ondelettes que l'on vient de voir (Simoncelli et al.)
On retient les matrices de corrélation $G^l \in \mathbb{R}^{m_l \times m_l}$,



$$\begin{aligned} G_{p,q}^l &= f_p^l \times (f_q^l)^T \\ &= \sum_{i=1}^{N_l} f_p^l(i) \cdot f_q^l(i), \end{aligned} \quad (1)$$

où p, q est l'indice des filtres $p, q \in \{1, \dots, m_l\}$.

Synthèse

On cherche à synthétiser une image ayant les mêmes matrices de corrélation que l'exemple

On part d'un bruit blanc t et on définit une fonction de cout à la couche l par

$$E_l = \frac{1}{4(N_l \cdot m_l)^2} \sum_{p=1}^{m_l} \sum_{q=1}^{m_l} \|G_{p,q}^l - \tilde{G}_{p,q}^l\|^2. \quad (2)$$

On combine certaines couches

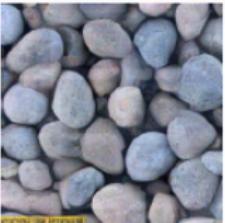
$$\mathcal{L}_{cnn}(I, \tilde{I}) = \sum_{l=0}^L w_l E_l, \quad (3)$$

où w_l sont des poids

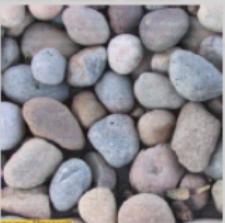
Puis on minimise la fonction de cout (back propagation et méthode itérative quasi-Newton)

Réseaux profonds convolutionnels

pool4



original



Portilla & Simoncelli



Synthèse par CNNs

Synthesised



Source



Synthèse par CNNs



Synthèse par CNNs



Synthèse par CNNs



Artefacts de synthèse

Transfert de style

Et aussi (à partir d'idées très semblables)

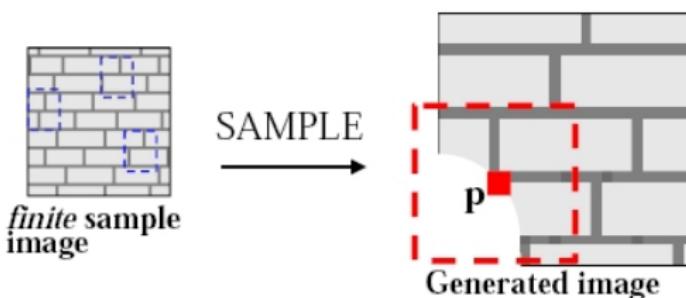


Les méthodes de rééchantillonnage

(Efros-Leung '99)

Idée :

- considérer la texture comme la réalisation d'un **champs de Markov**
- au lieu d'échantillonner à partir d'un modèle du type Gibbs (dont les paramètres sont appris sur la texture), on **échantillonne directement sur une image exemple A**
- la synthèse s'effectue pixel par pixel
- pour chaque nouveau pixel de la zone à synthétiser, on suppose qu'une partie de son voisinage est déjà synthétisé, et **on cherche dans A le pixel dont le voisinage est le plus proche.**



(fig : Efros-Leung 99)

- Soient Ω_1 et Ω_2 les domaines de A et B respectivement (sous-ensembles de \mathbf{Z}^2)
- Soit $F = \{a_1, \dots, a_K\}$ un sous-ensemble de \mathbf{Z}^2 .
- Pour $p \in \Omega_1$, soit $N_A(p) = \{A(p + a_1), \dots, A(p + a_K)\}$, le voisinage de p correspondant à F .
- On définit une distance entre voisinages, e.g. :

$$d_v(N_A(p), N_B(p')) = \frac{1}{K} \sum (A(p + a_i) - B(p' + a_i))^2$$

- pour un pixel $p' \in \Omega_2$, et $\epsilon > 0$ soit

$$O_\epsilon(p') = \{p \in \Omega_1 | d_v(N_A(p), N_B(p')) < \epsilon\}$$

Pour synthétiser une nouvelle texture B : pour chaque $p' \in \Omega_2$

- On calcule $O_\epsilon(p')$,
- on tire un pixel p de $O_\epsilon(p')$ uniformément,
- on affecte la valeur $B(p') = A(p)$

... mais bien sûr, on ne sait pas calculer $O_\epsilon(p')$, puisque les valeurs prises par B dans le voisinage de p' ne sont pas connues.

La distance d est alors modifiée pour ne prendre en compte que les pixels de B où la synthèse a déjà été effectuée.

Puis la synthèse s'effectue séquentiellement, en partant d'un petit morceau de l'image originale A (quelques pixels).

Avantages : Excellent résultats, même avec des textures très structurées.

Inconvénients : Très lent, très sensible au choix de B , instable.

→ Démo en ligne (*efros ipol* sous google)

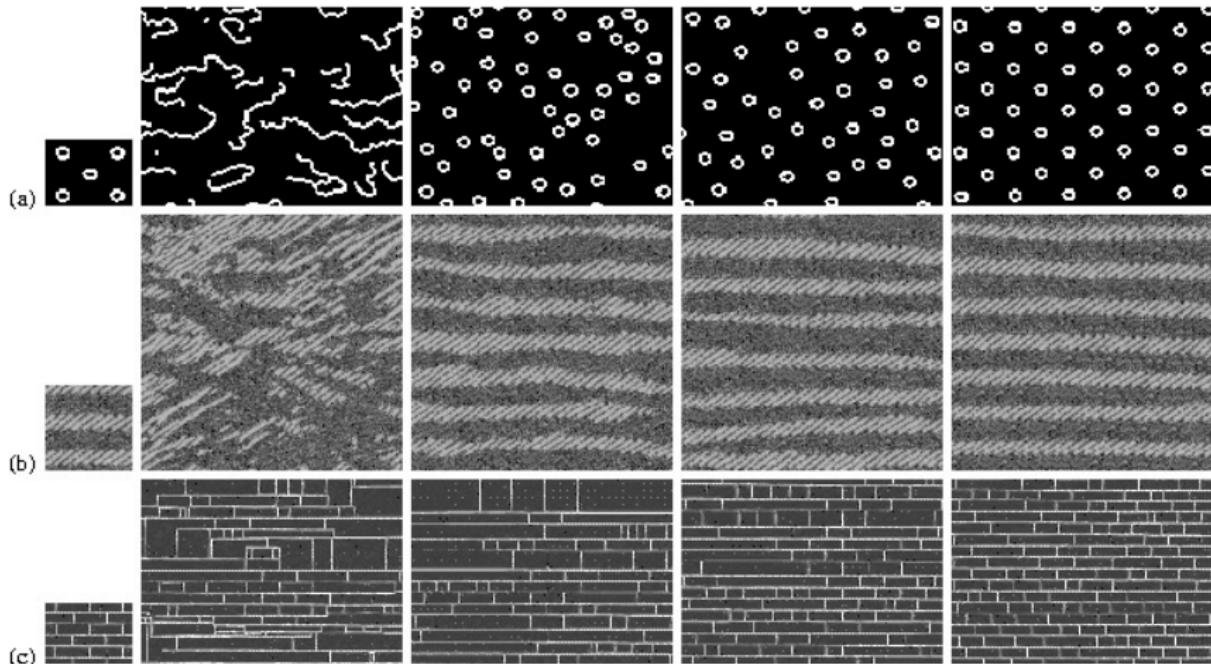
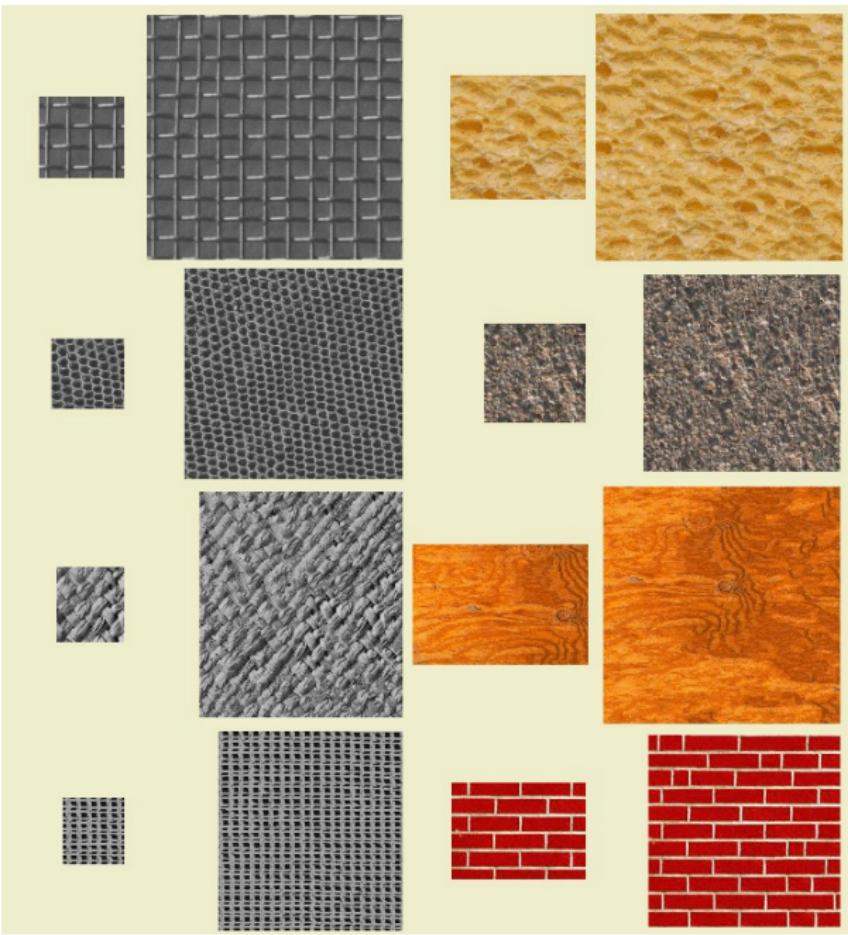


Figure 2. Results: given a sample image (left), the algorithm synthesized four new images with neighborhood windows of width 5, 11, 15, and 23 pixels respectively. Notice how perceptually intuitively the window size corresponds to the degree of randomness in the resulting textures. Input images are: (a) synthetic rings, (b) Brodatz texture D11, (c) brick wall.

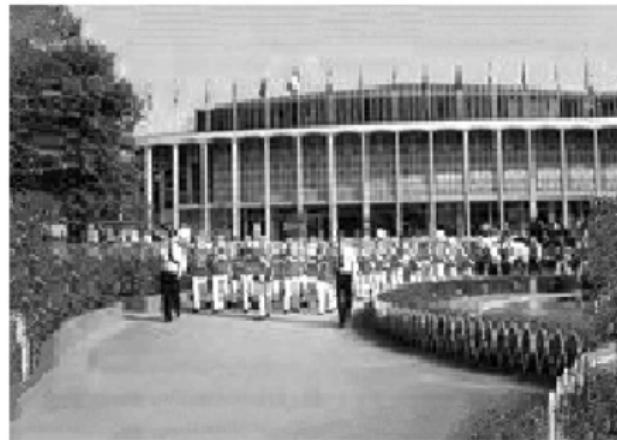
Extrait de Efros-Leung 1999



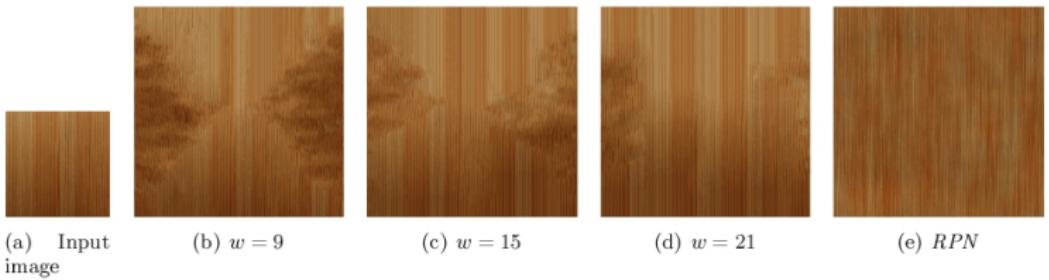
ut it becomes harder to lau-
ound itself, at "this daily-
wing zooms," as House Der-
escribed it last fall. He fail-
ut he left a ringing question
more years of Monica Lewin-
inda Tripp?" That now seem
Political comedian Al Frat-
extphase of the story will

de reme a the ral counta rself, at "this da Lew accey-
it ndatyears counf Tring rooms," as Heft he fast nd it l-
ars dat noears ortseas ribed it last nt hest bedian Al. I
economical Homd it h Al. Heft ars Gf, as da Lewindailf J
lian Al Ths," as Lewing questies last aticarsticall. He
is dian Al last fal counda Lew, at "this dailyyears d illy
edianicall. Hoozewing rooms," as House De fale f De
und itical counoestscribed it last fall. He fall. Hefft
rs orohoneon it nd it he left a ringing questica Lewin.
icars coecoms," astore years of Monica Lewinow seee
a Thas Fring zoome stooniscat nowea re left a rouuse
bouestof Mhe lelfta Lest fast ngine läumesticars Hef
nd it rip?" Trhousef, a ringind it'sonestid it a ring que-
astical cois ore years of Moung fall. He ribof Mouse
ore years ofanda Tripp?" That hedian Al Lest fassee yea-
nda Tripp? Tpolitical comedian Aléthe fw se ring que
olitical cona re years of the storears oafas l Frat nica L
ros Lew se lesta rime l He fas quest nging of, at beou

Extrait de Efros-Leung 1999



Extrait de Efros-Leung 1999



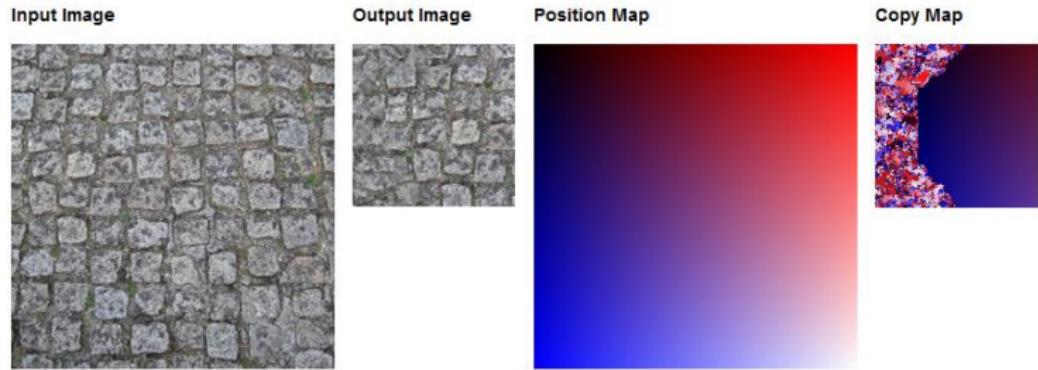
(a) Input
image

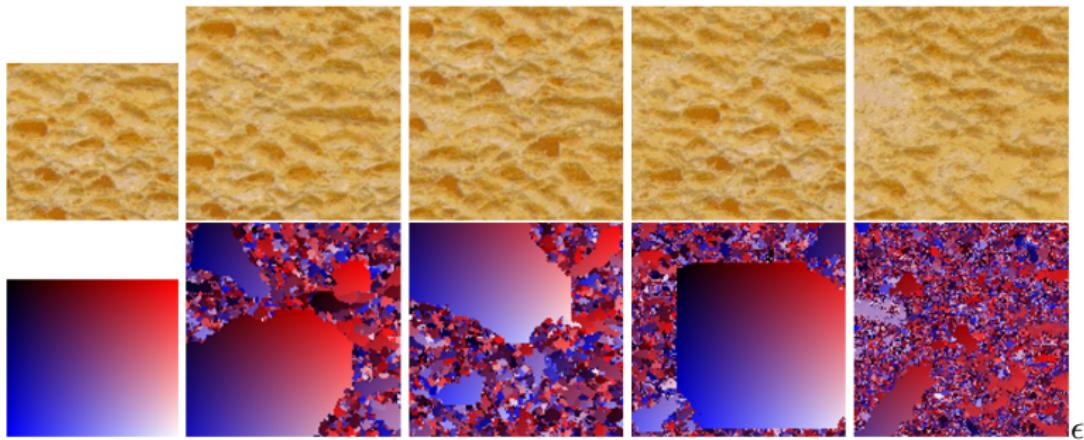
(b) $w = 9$

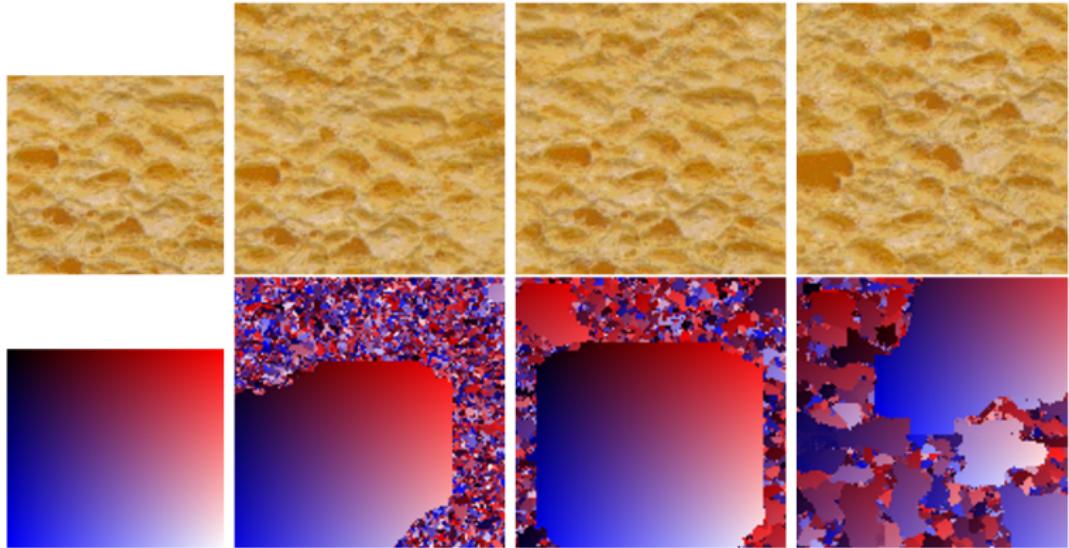
(c) $w = 15$

(d) $w = 21$

(e) RPN







Patch size : 5, 7, 9

De nombreuses variantes ont été proposées pour la synthèse de texture non-paramétrique par *patchs*
Essentiellement pour améliorer la vitesse de synthèse

- Quantification vectorielle
Fast Texture Synthesis using Tree-structured Vector Quantization L. L. Wei et M. Levoy, 2000
- Synthèse directement par patchs (au lieu de pixels)
Image quilting for texture synthesis and transfer, A. Efros and B. Freeman, 2001
→ Démo IPOL
- Implémentations parallèles ; GPU
State of the art in example-based texture synthesis, LY Wei, S Lefebvre, V Kwatra, G Turk, 2009

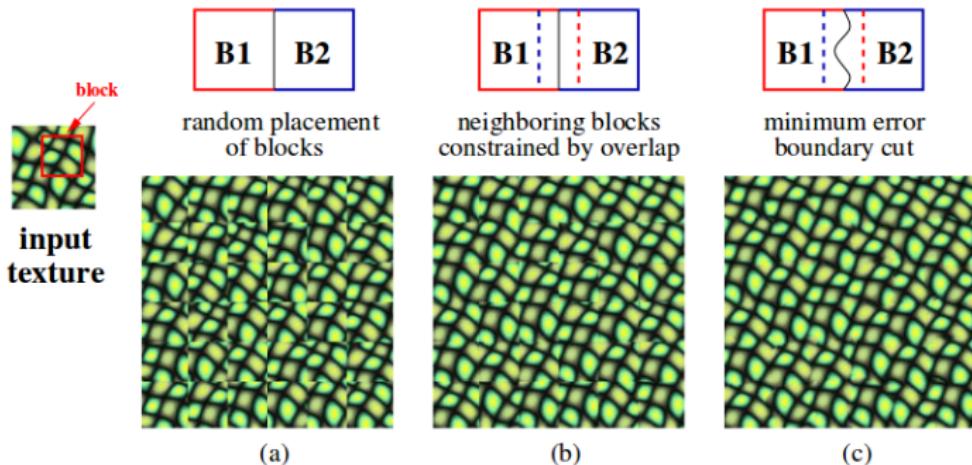


Illustration de Efros-Freeman 2001

Les patches sont accolés séquentiellement avec une procédure de recollement (frontière entre patchs minimisant les discontinuités)

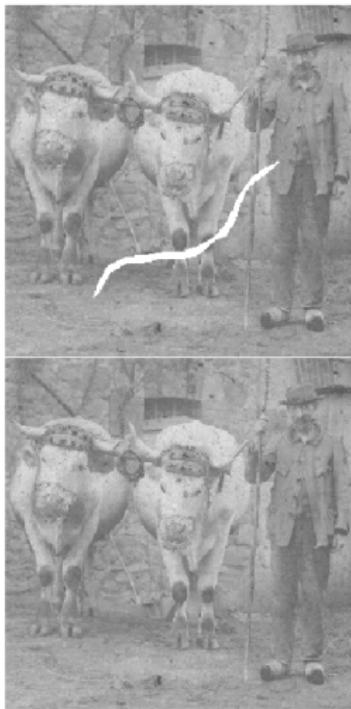
Suite : cours sur l'inpainting

Inpainting (ou désoccultation) : comment retrouver les parties manquantes d'une image ?

De nombreuses applications :

- restauration des photographies et films anciens ;
- édition numérique de photos ;
- post-processing video ;
- truquages.

Problème : comment interpoler une image de la façon la plus **réaliste** ?



Depuis quinze ans : de très nombreuses méthodes

- **variationnelles**

(Masnou-Morel '98, Chan-Shen '01, Caselles et al '01, Bertalmio et al '01, Tschumperlé-Deriche '04, Bornemann et al '05, etc.) ;

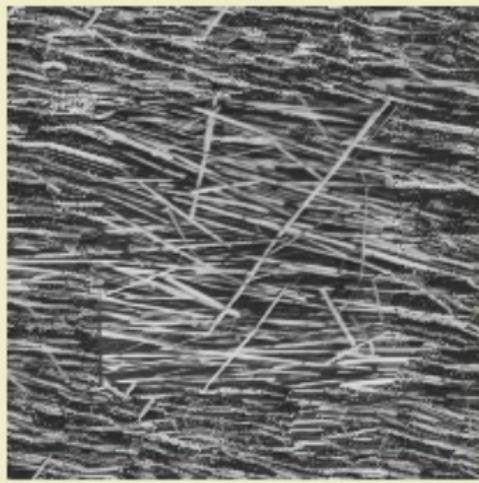
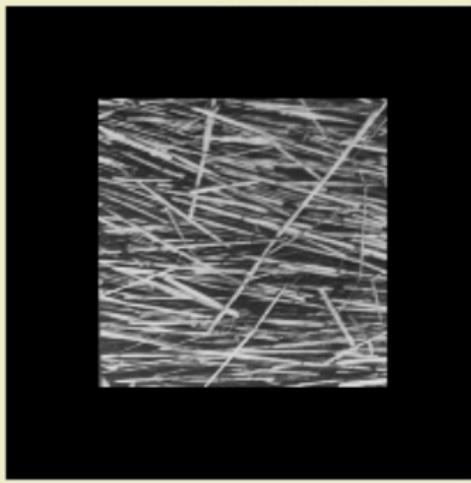
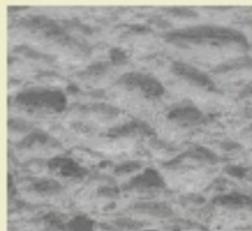
- **rééchantillonage ("copier-coller")**

Prolongement des méthodes vues pour la synthèse de texture
(Efros-Leung '99, Wei-Levoy '00, Efros-Freeman '01, Ashikmin et al '01, Harrison '01, Barnard et al '02, Criminisi-Pérez-Toyama '04, Pérez-Gangnet-Blake 04, de Bonet '97, Igely-Pereira '97, etc.).

- **Décompositions dans un dictionnaire**

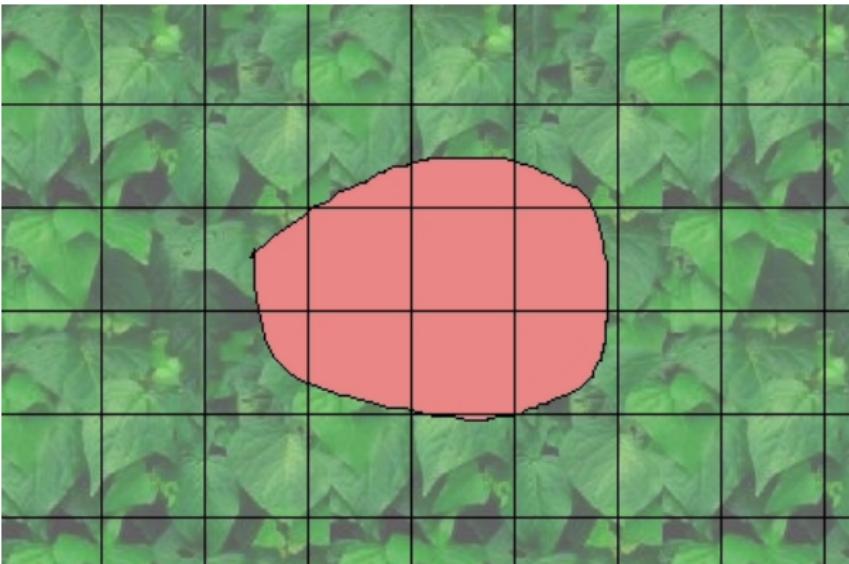
(Fadil-Stark 05, Peyré 09, etc.)

Même méthode que pour la synthèse de texture



Extrait de Efros-Leung 1999

Extensions : → synthèse par "patches" (Efros-Freeman 2000,
Pérez-Gangnet-Blake 04, Wei-Lefebvre-Kwatra 09)

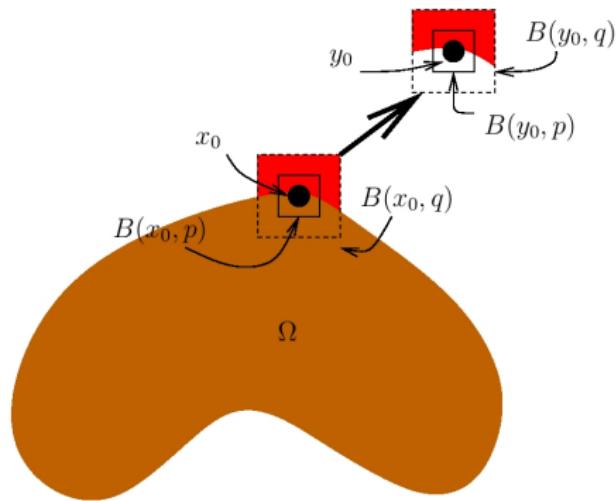


Principe (Pérez-Gangnet-Blake 04)

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 patchs, etc.)

Les résultats dépendent fortement de

① **paramètres p, q**

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

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

q grand \rightarrow meilleures transitions entre patchs

② **ordre de remplissage :**

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

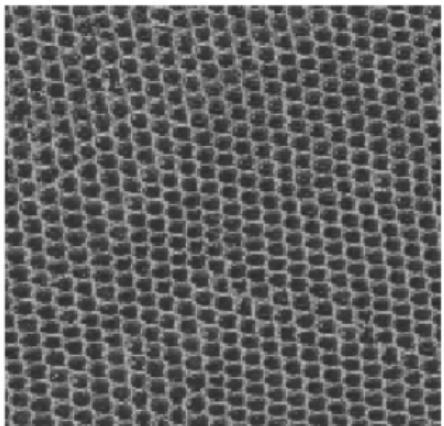
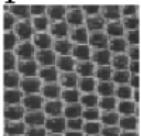
[Criminisi-Pérez-Toyama '04] : x_0 minimise une fonctionnelle dépendant de

- i) la géométrie de Ω ,
- ii) la géométrie des lignes de niveau autour de x_0 .

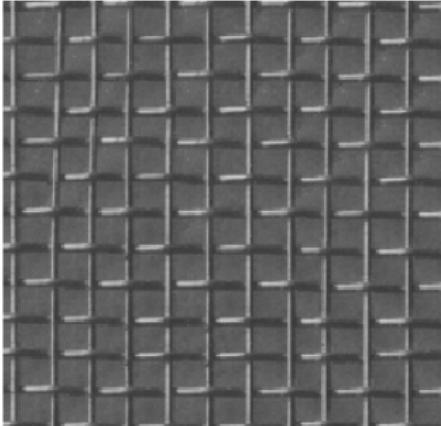
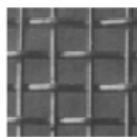
Toujours d'excellents résultats sur les textures :

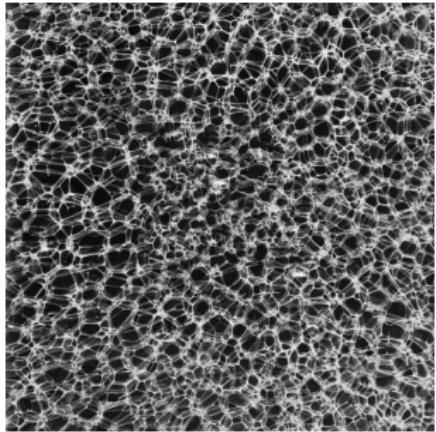
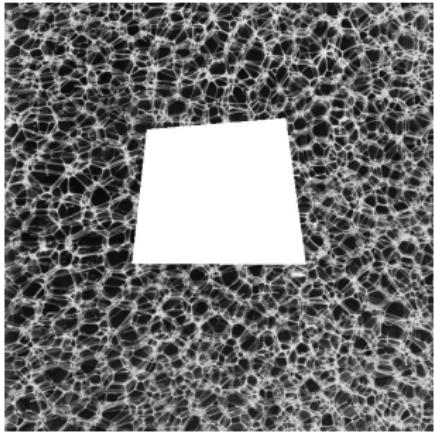
Brodatz Results

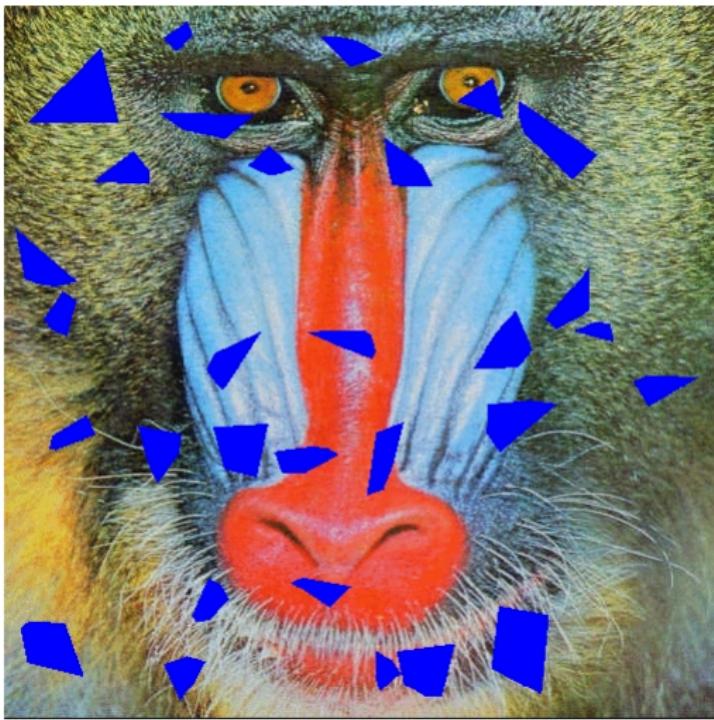
reptile skin

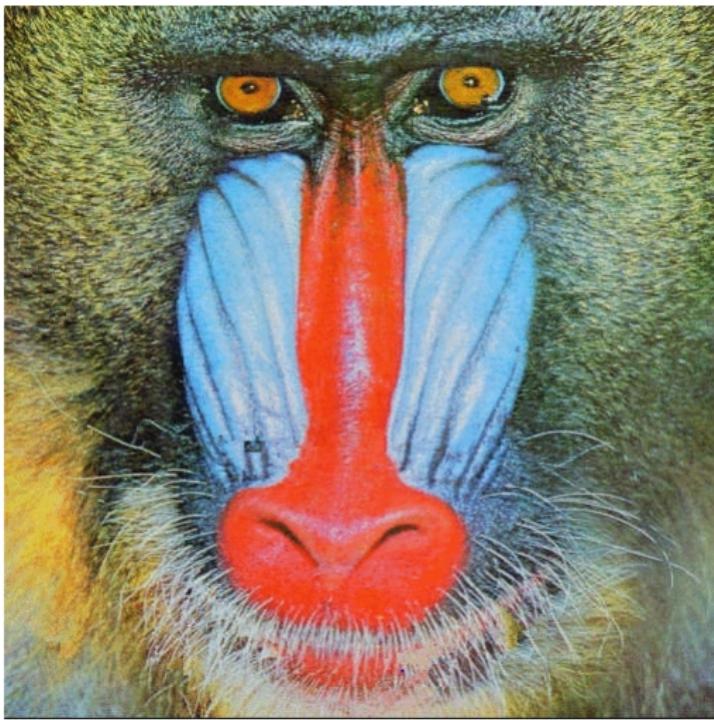


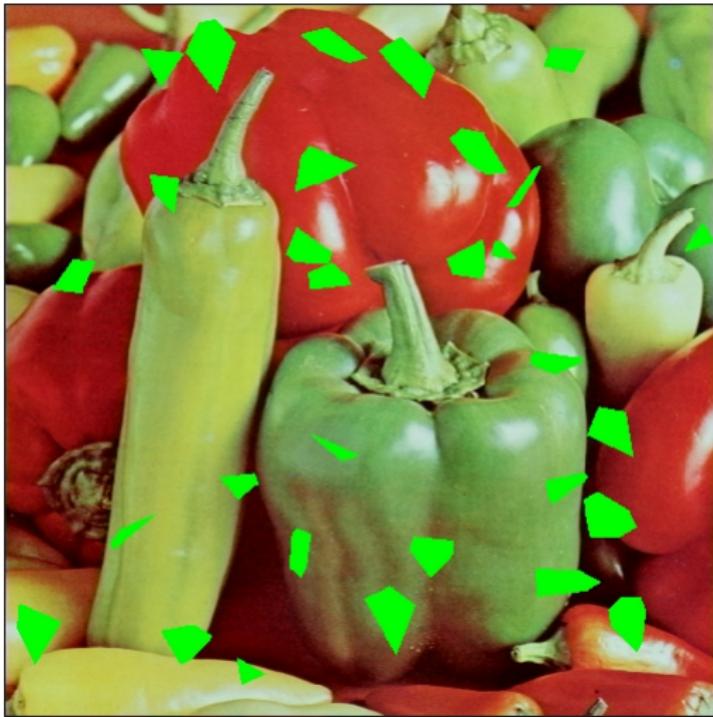
aluminum wire



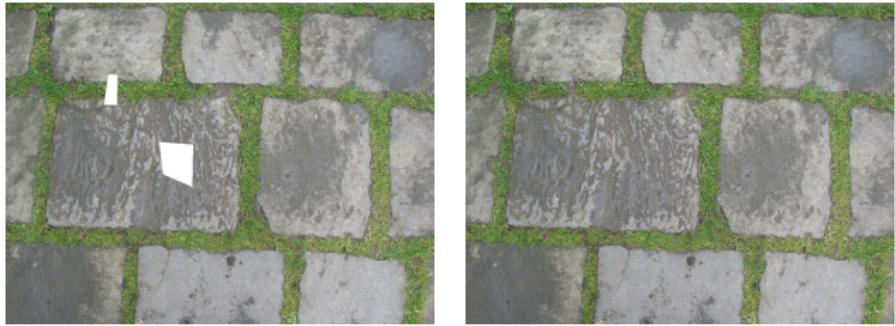












Mais aussi sur la restauration des structures de l'image



Space-time video completion, Wexler, Shechtman, Irani, 2007

- On considère des patchs spatio-temporels
- Partant d'une initialisation, chaque patch est remplacé par son plus proche voisin à l'extérieur de l'occultation
- L'image est reconstruite en remplaçant chaque pixel par la moyenne des valeurs des patchs le recouvrant
- On itère

Voir :

http:

//perso.telecom-paristech.fr/gousseau/video_inpainting/

Donne aussi de très bons résultats dans le cas des images

Grâce à une implémentation multi-échelle : très bon respect de la structure des images et des vidéos

Très lent ...