

# DEep Learning for Image Restoration and Synthesis (MVA)

Saïd Ladjal ([said.ladjal@telecom-paris.fr](mailto:said.ladjal@telecom-paris.fr))  
Andrés Almansa ([andres.almansa@parisdescartes.fr](mailto:andres.almansa@parisdescartes.fr))  
Alasdair Newson ([alasdair.newson@telecom-paris.fr](mailto:alasdair.newson@telecom-paris.fr))  
<http://delires.wp.imt.fr>

# Partie III: Textures

- Définition des textures
  - Quelques caractéristiques utiles.
- Caractéristiques issues de réseaux: Gatys.
- Un exemple de réseau génératif.

# Texture?

- Une texture est une image qui possède des caractéristiques intuitives
- Une période ou pseudo-période petite devant la taille de l'image.
- Des composants caractéristiques qui se répètent... mais avec des différences possibles.



Crédit: Yann Gousseau





# Description mathématique

- Il a été proposé que les informations d'une texture peuvent être résumées par quelques statistiques
  - Premier ordre: Histogramme, Moyenne, Écart-type.
  - Second ordre: Covariance.

# Description mathématique

- Il a été proposé que les informations d'une texture peuvent être résumées par quelques statistiques
  - Premier ordre: Histogramme, Moyenne, Écart-type.
  - Second ordre: Covariance.
- Il reste à définir à quoi appliquer ces statistiques:
  - Niveaux de gris.
  - Réponses à des filtres (ondelettes, filtres de Gabor) et plus tard des caractéristiques issues de réseaux.



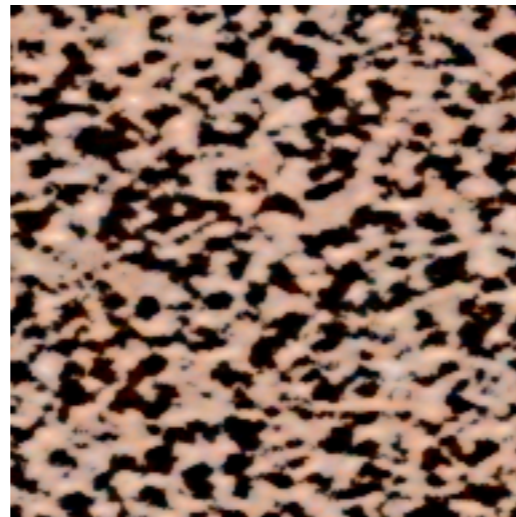
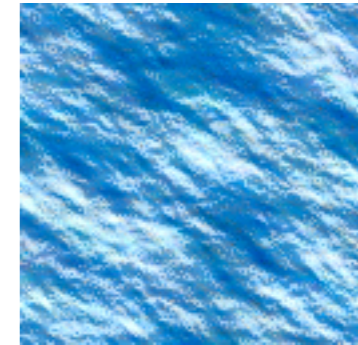
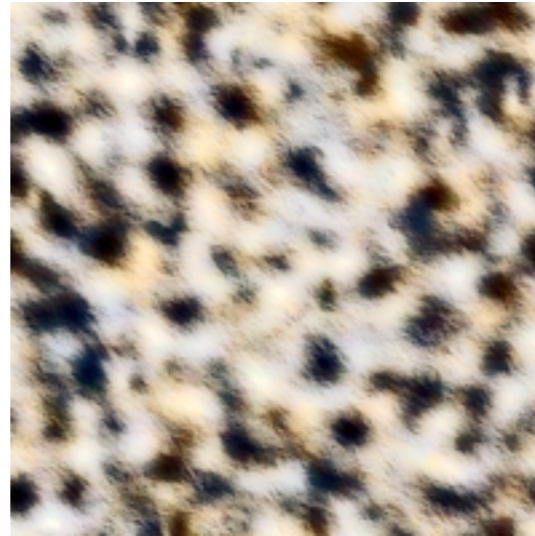
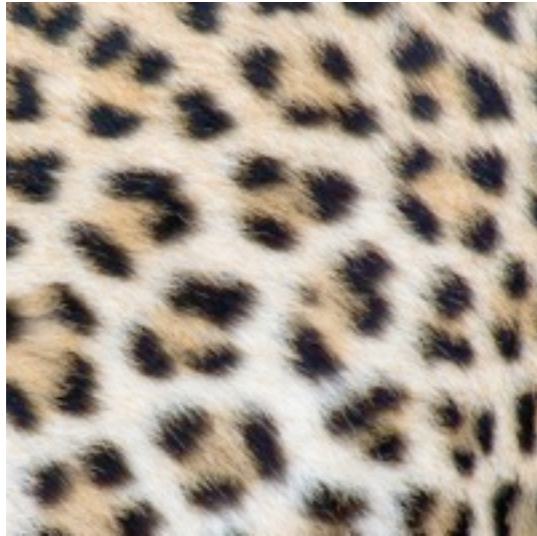
# Reconnaissance/synthèse

- Les descripteurs de texture peuvent servir à de la reconnaissance:
  - Par exemple en calculant des distances entre les descripteurs d'une image cible et une banque de données de descripteurs.
- Ils peuvent aussi servir à synthétiser (**Deep**):
  - Essayer de produire une image qui possède les mêmes caractéristiques qu'une image d'exemple (**Loss**).
  - Ou concevoir une "machine" qui transforme un bruit en texture possédant les caractéristiques souhaitées (**Architecture**).

# Synthèse Heeger 1995

- Les descripteurs sont des histogrammes de transformées en ondelettes.
- On part d'une image de bruit blanc.
- Modifier les histogrammes pour atteindre la cible.
- Reconstruire (du mieux que l'on peut, par exemple une solution aux moindres carrés)
- Répéter.

# Synthèse Heeger

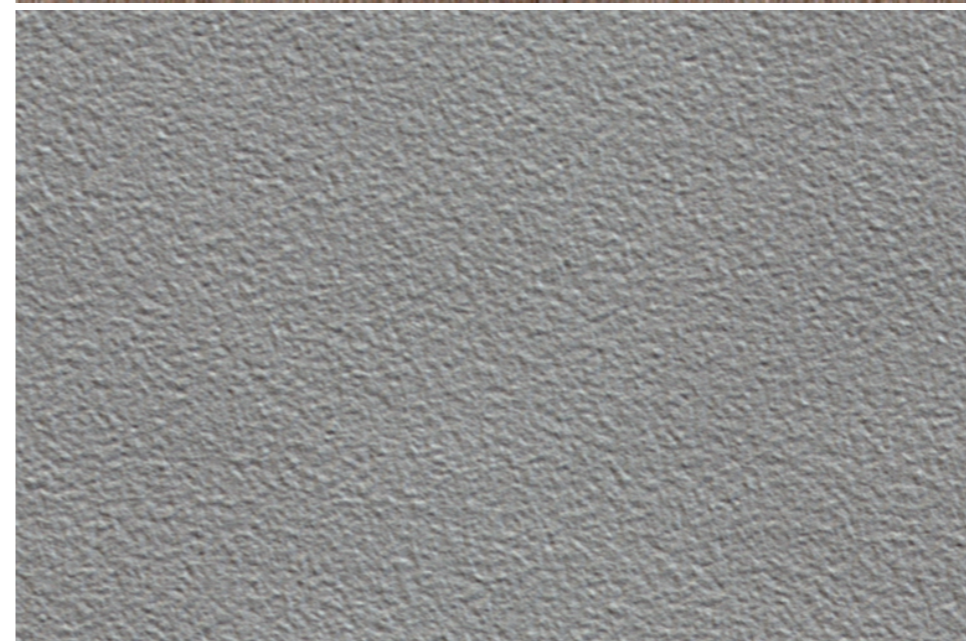


Source IPOL <http://demo.ipol.im/demo/79/>

# Synthèse par phase aléatoire

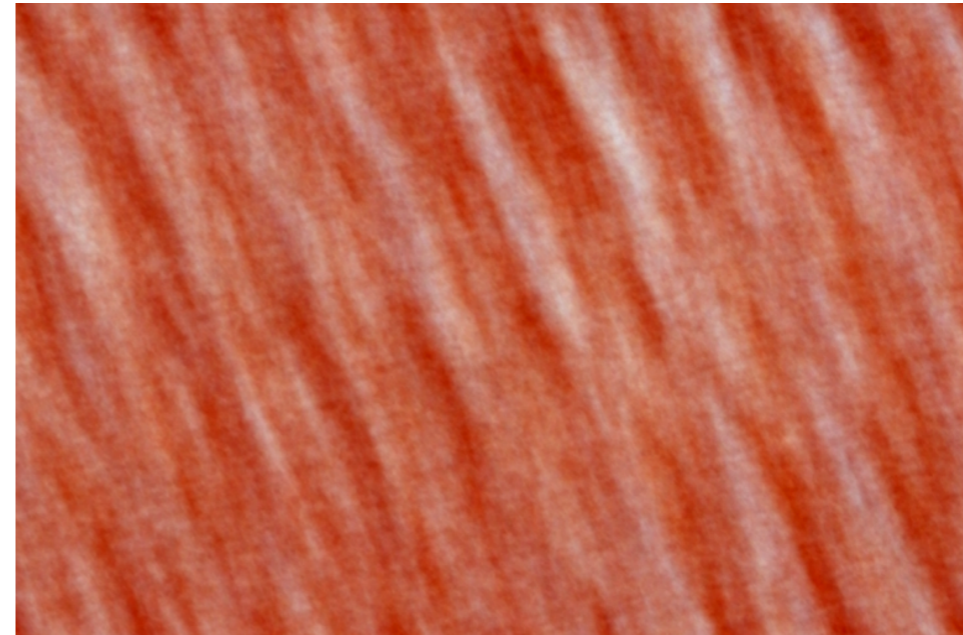
- Les descripteurs sont le spectre (module de la TF)
- Si c'est la seule caractéristique à respecter cela revient à tirer uniformément une phase de TF.
  - ➔ La phase est plaquée sur le spectre caractéristique.
- Cela peut être relié à la limite d'accumulation d'un même objet qui « tombe » aléatoirement une infinité de fois sur l'image à des endroits aléatoires.

# Synthèse par phase aléatoire



[http://demo.ipol.im/demo/ggm\\_random\\_phase\\_texture\\_synthesis/](http://demo.ipol.im/demo/ggm_random_phase_texture_synthesis/)

# Synthèse par phase aléatoire

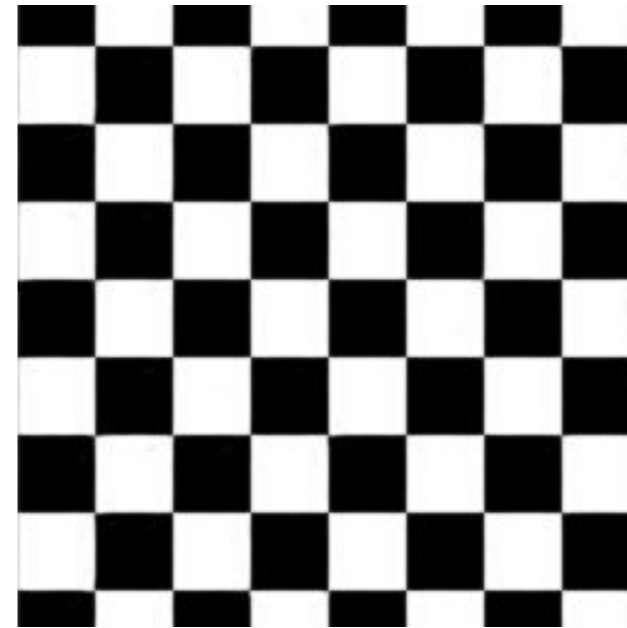
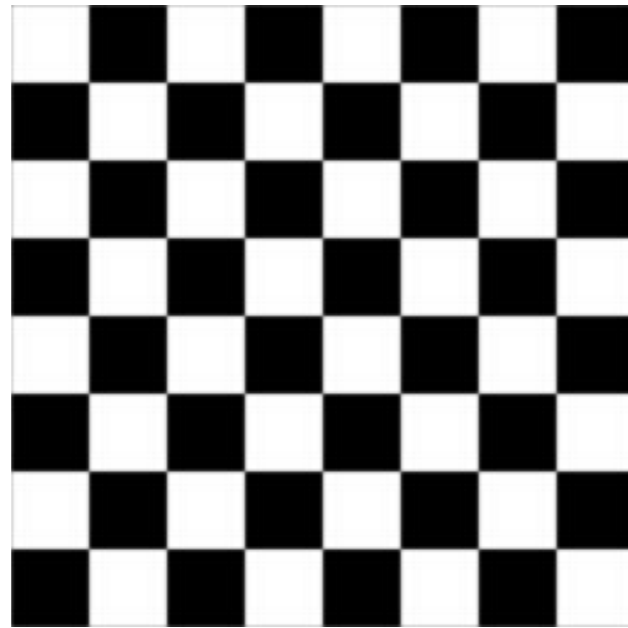


[http://demo.ipol.im/demo/ggm\\_random\\_phase\\_texture\\_synthesis/](http://demo.ipol.im/demo/ggm_random_phase_texture_synthesis/)

# Synthèse Simoncelli-Portilla (2000)

- Les descripteurs sont des statistiques d'ondelettes (moyenne, variance, kurtosis)
- On ajoute également des corrélations spatiales entre coefficients d'ondelettes. De l'ordre de 1000 paramètres.
  - On part d'une image de bruit blanc.
  - Pour chaque paramètre on modifie l'image pour atteindre la cible. (les auteurs suivent de gradient de  $F$ , si la cible est  $F(x)=c$ )
- Répéter.

# Synthèse Simoncelli-Portilla (2000)



source: Simoncelli





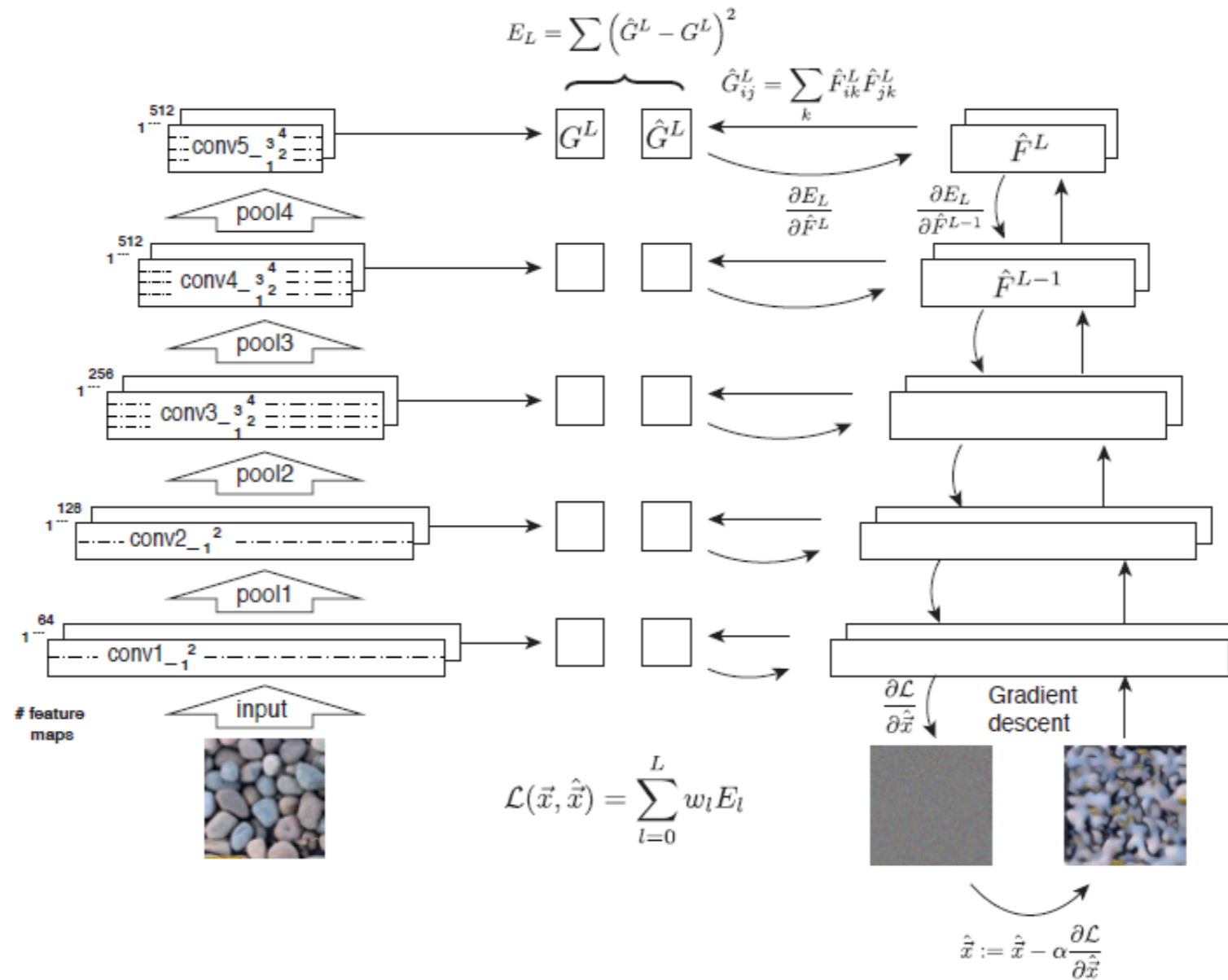
# Synthèse Gatys 2015

- Les descripteurs sont extraits des features de réseau (vgg19)
- Les covariances des features forment des matrices de Gram  $G$ . Ce sont ces matrices qui sont la cible.

$$g_{ij} = \frac{1}{wh} \sum_{l=1}^h \sum_{k=1}^w B(l, k, i) B(l, k, j)$$

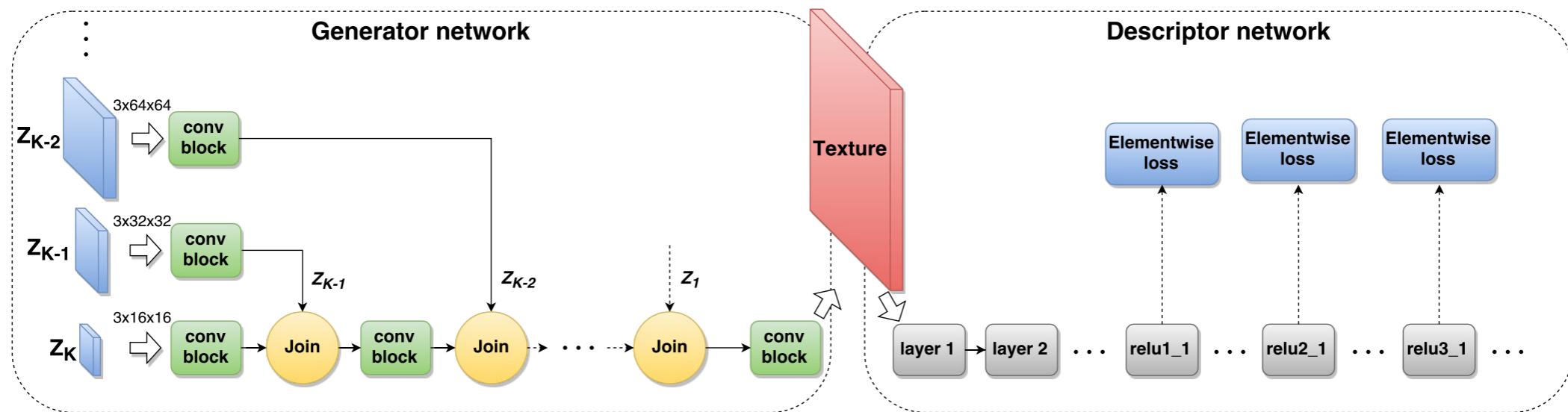
Loss: 
$$\mathcal{L}(x) = \sum_i \|G^{l_i}(x) - G^{l_i}\|^2$$

# Synthèse Gatys 2015



# Synthèse Ulyanov 2016

- Les descripteurs sont ceux de Gatys
- Le modèle est génératif: Un réseau est entraîné pour produire une image qui respecte les statistiques de Gatys à partir d'un bruit blanc.



Génération

Entraînement

$$\mathcal{L}(x) = \sum_i \|G^{l_i}(x) - G^{l_i}\|^2 \quad (x: \text{bruit blanc})$$