

# Détecteurs et descripteurs



GIF-4105/7105 Photographie Algorithmique, Hiver 2017  
Jean-François Lalonde

Merci à D. Hoiem et A. Efros pour les slides

# Comment aligner deux images?

- Déterminer une transformation globale automatiquement

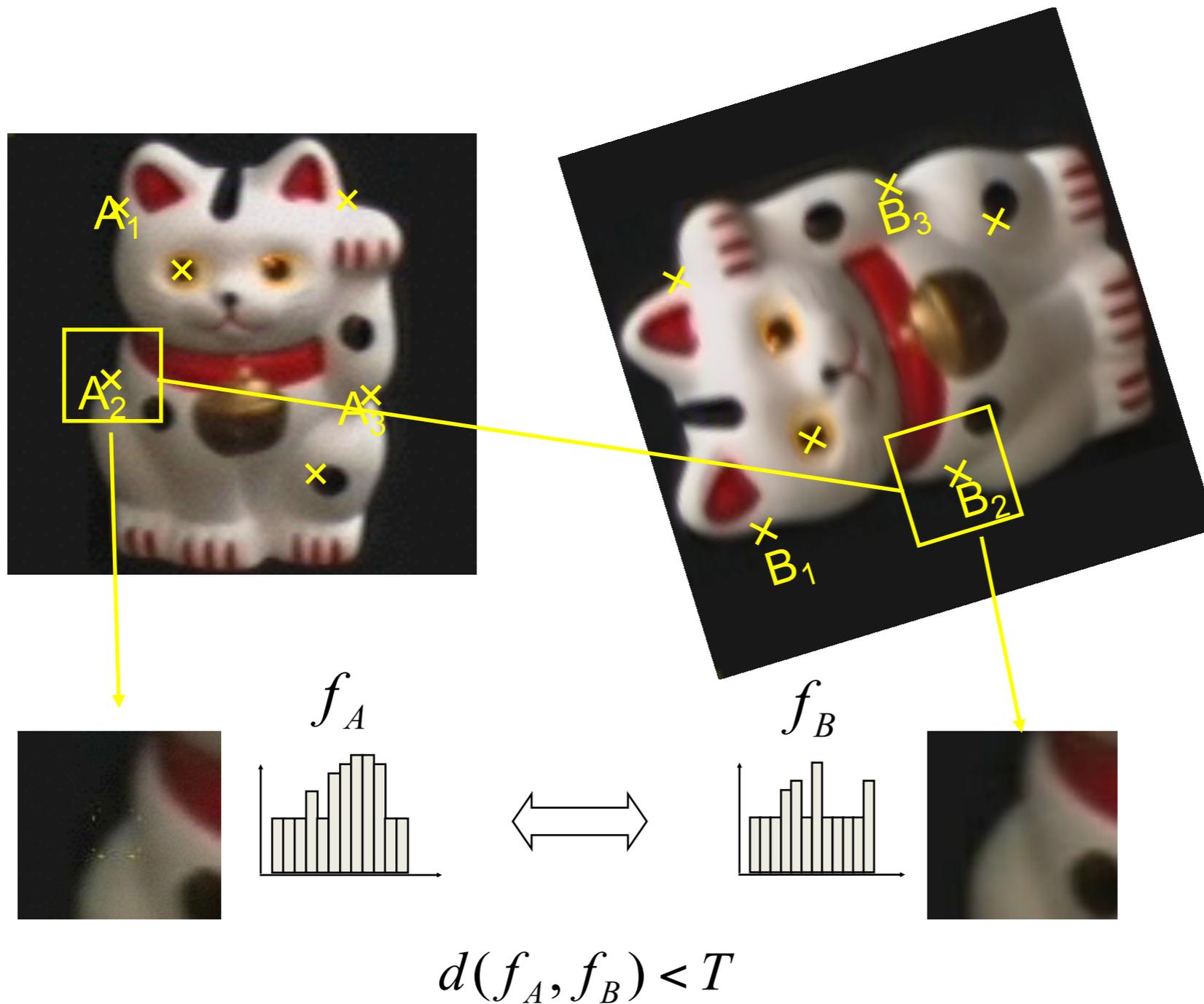


# Comment aligner deux images?

- Quoi faire si:
  - Pas seulement translation, mais rotation et facteur d'échelle?
  - Les images se chevauchent partiellement?



# Aujourd'hui: détecteurs et descripteurs



1. Trouver une série de points distinctifs
2. Définir une région autour de chaque point
3. Extraire et normaliser la région
4. Calculer un descripteur de la région
5. Apparier les descripteurs (de façon robuste)

# Question

- Pourquoi ne pas subdiviser l'image en blocs et apparier les blocs directement?



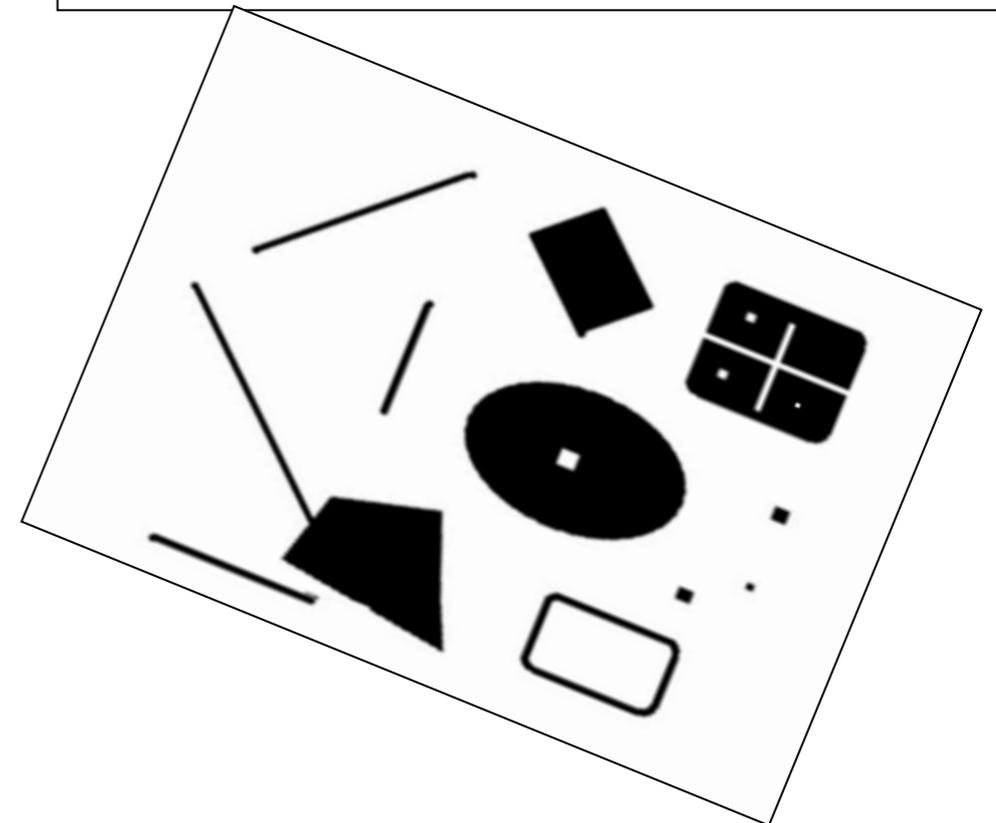
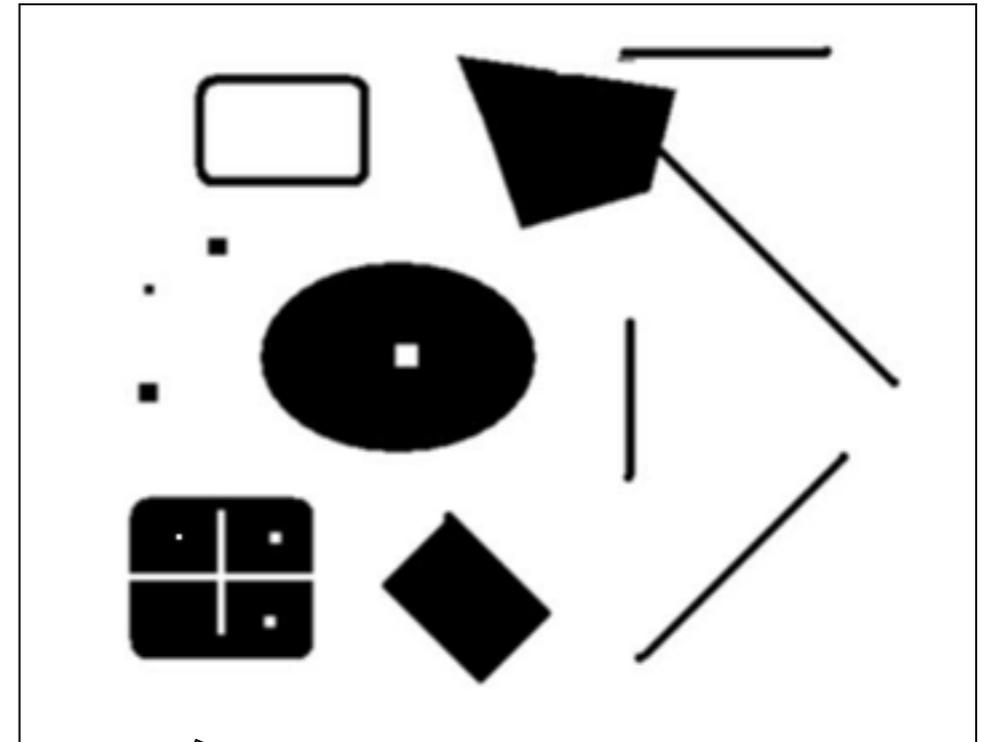
# Points d'intérêt: but



Détecter des points qui sont  
représentatifs  
distincts

# Localisation des points

- Voici une image.
- Toute à l'heure, je vous montrerai une version déformée de l'image.
- Identifiez des points sur l'image qui seront faciles à identifier lorsque l'image sera déformée.



# Choisir des points d'intérêt

Vous devez rencontrer un ami.  
Où lui donnez-vous rendez-vous?



# Choisir des points d'intérêt

Vous devez rencontrer un ami.  
Où lui donnez-vous rendez-vous?



# Choisir des points d'intérêt

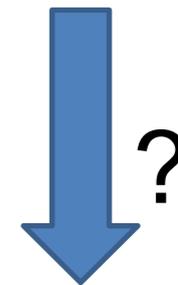
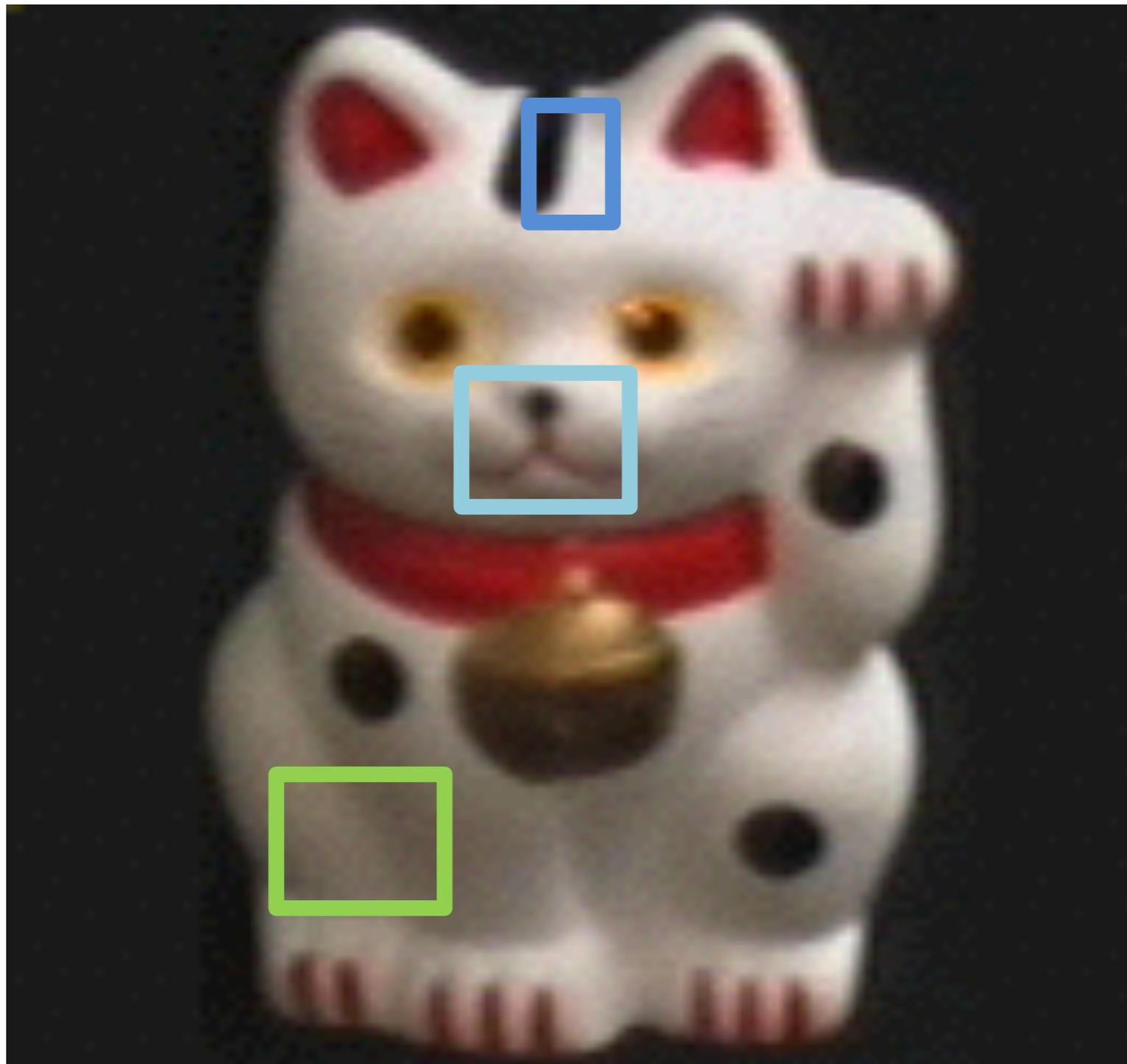
Coins



Sommets

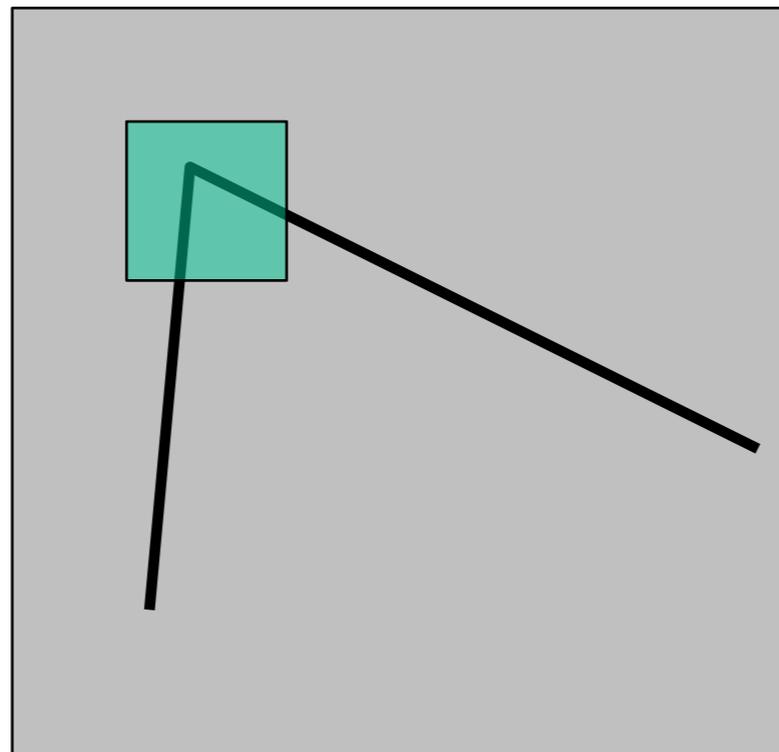


# Quelles fenêtres sont faciles à apparier?

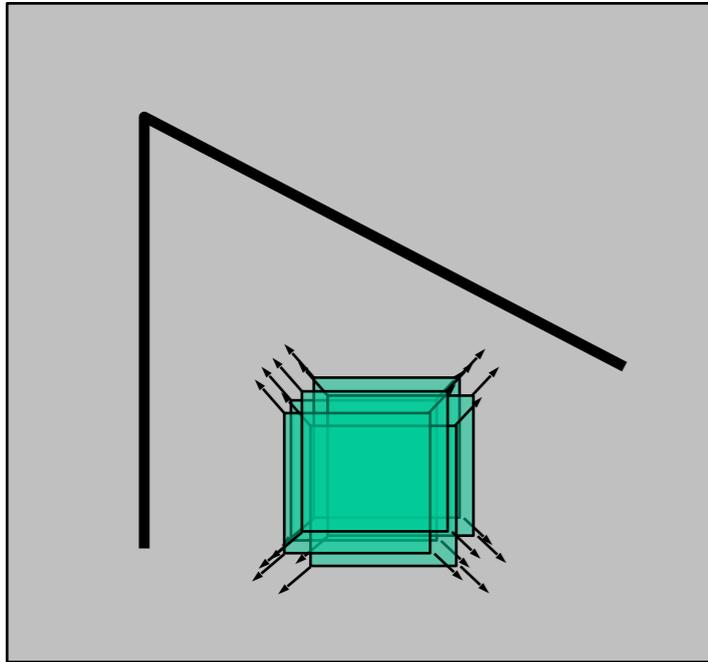


# Détecteur de coins de Harris

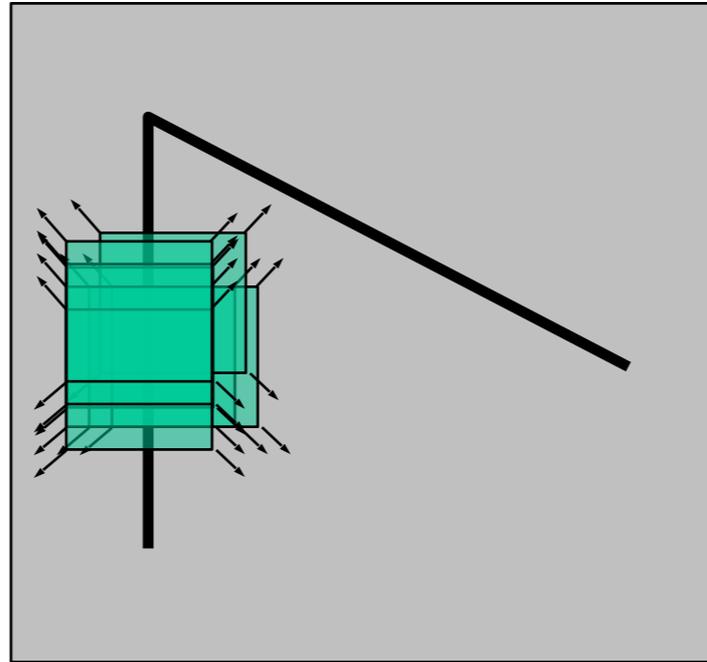
- Nous devrions reconnaître le point en considérant seulement une petite fenêtre autour du point;
- Si on déplace la fenêtre dans n'importe quelle direction, le changement d'intensité devrait être important.



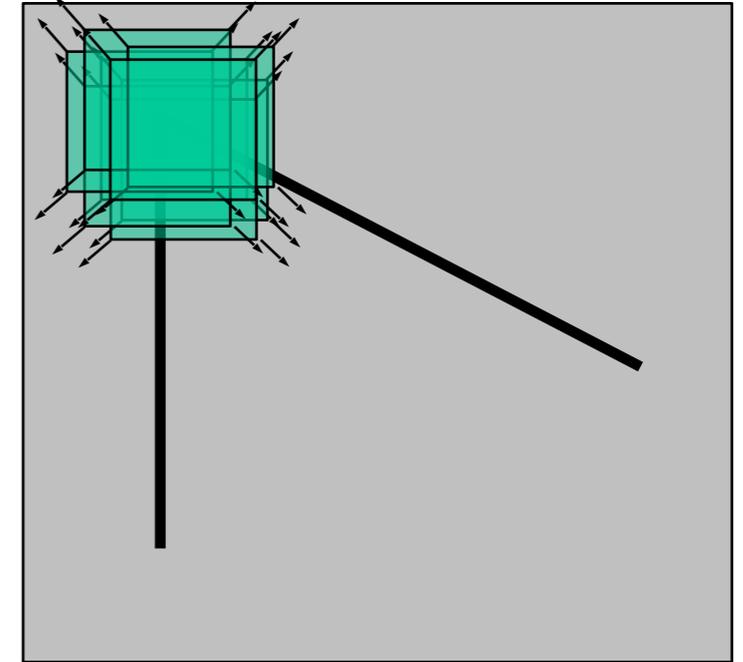
# Détecteur de Harris: intuition



région uniforme:  
aucun changement



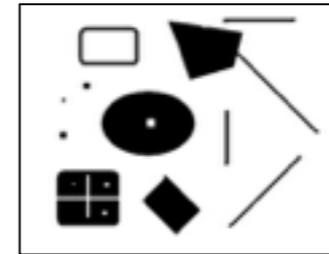
arête:  
pas de changement le long  
de l'arête



coin:  
changement dans toutes les  
directions

# Détecteur Harris

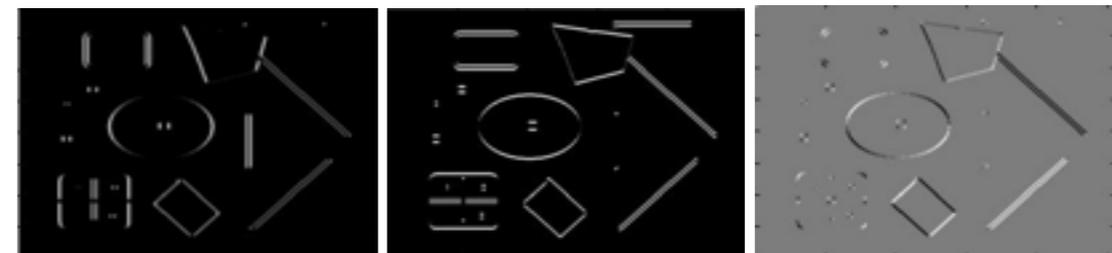
$$\mathbf{M} = g(\sigma) * \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}$$



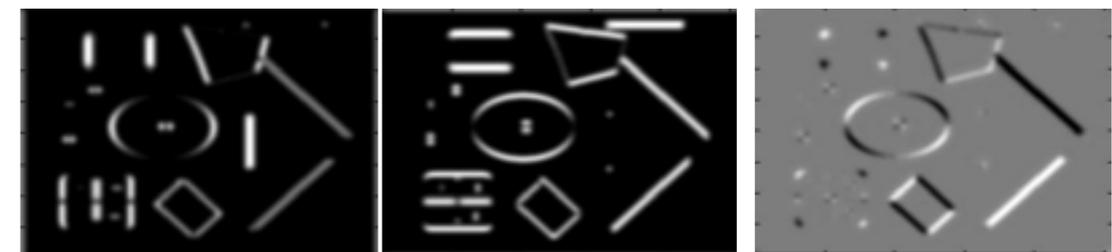
1. Dérivées



2. Dérivées au carré



3. Dérivées au carré, filtrées avec gaussienne



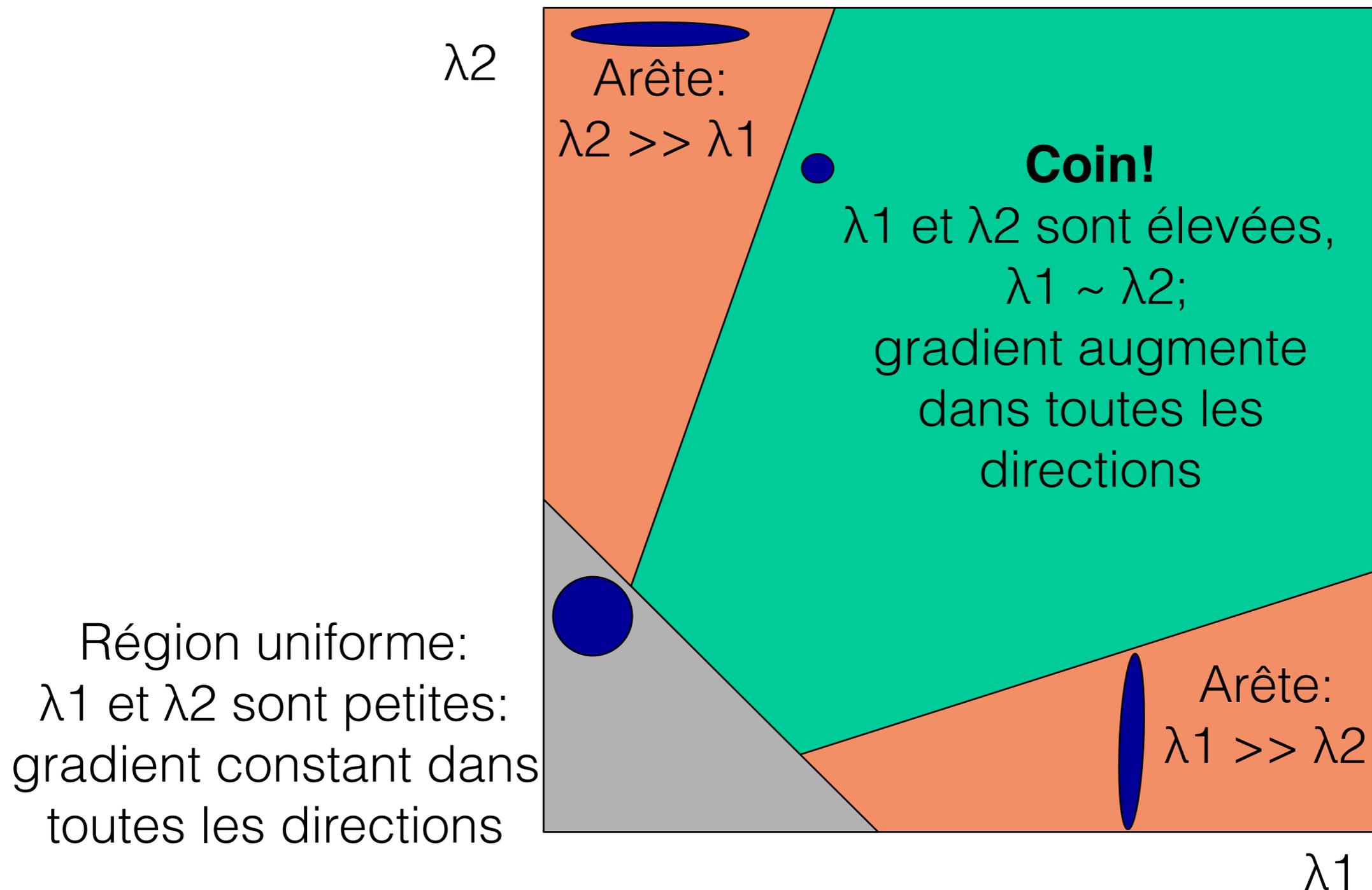
4. Calculer fonction des valeurs propres de  $\mathbf{M}$



# Démonstration (demo.m)

# Détecteur de Harris: interprétation

Classification des points en fonction des valeurs propres



# Détecteur de Harris: math

En pratique, nous n'avons pas besoin de calculer les valeurs propres

$$R = \frac{\det \mathbf{M}}{\text{tr } \mathbf{M}} \quad \begin{array}{l} \det \mathbf{M} = \lambda_1 \lambda_2 \\ \text{tr} = \lambda_1 + \lambda_2 \end{array}$$

$$\det \mathbf{M} = m_{11}m_{22} - m_{21}m_{12}$$

$$\text{tr } \mathbf{M} = m_{11} + m_{22}$$

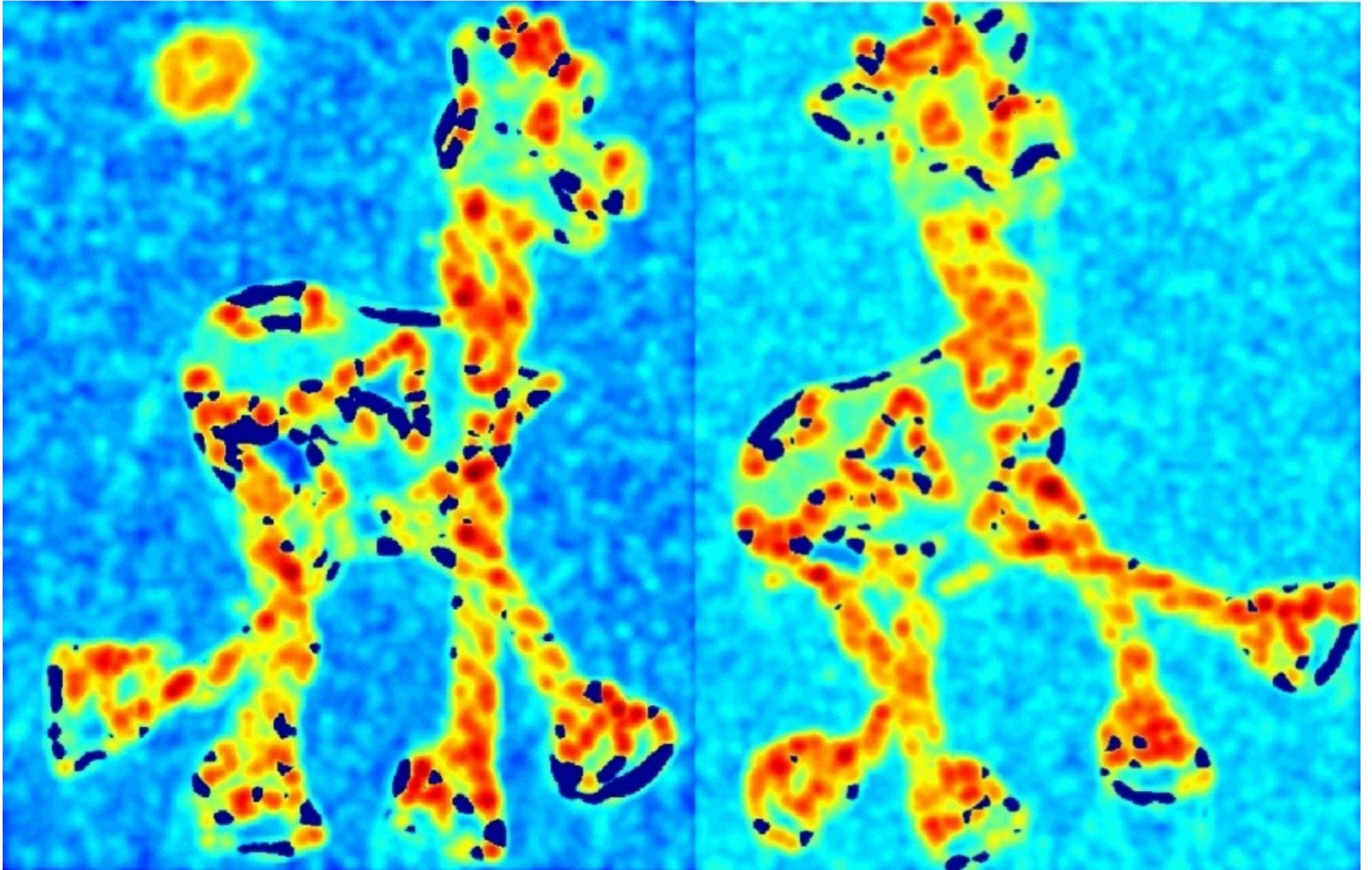
# Algorithme

- Calculer  $R$  pour tous les points dans l'image
- Appliquer:  $R > \text{seuil}$
- Retenir les maximums locaux seulement

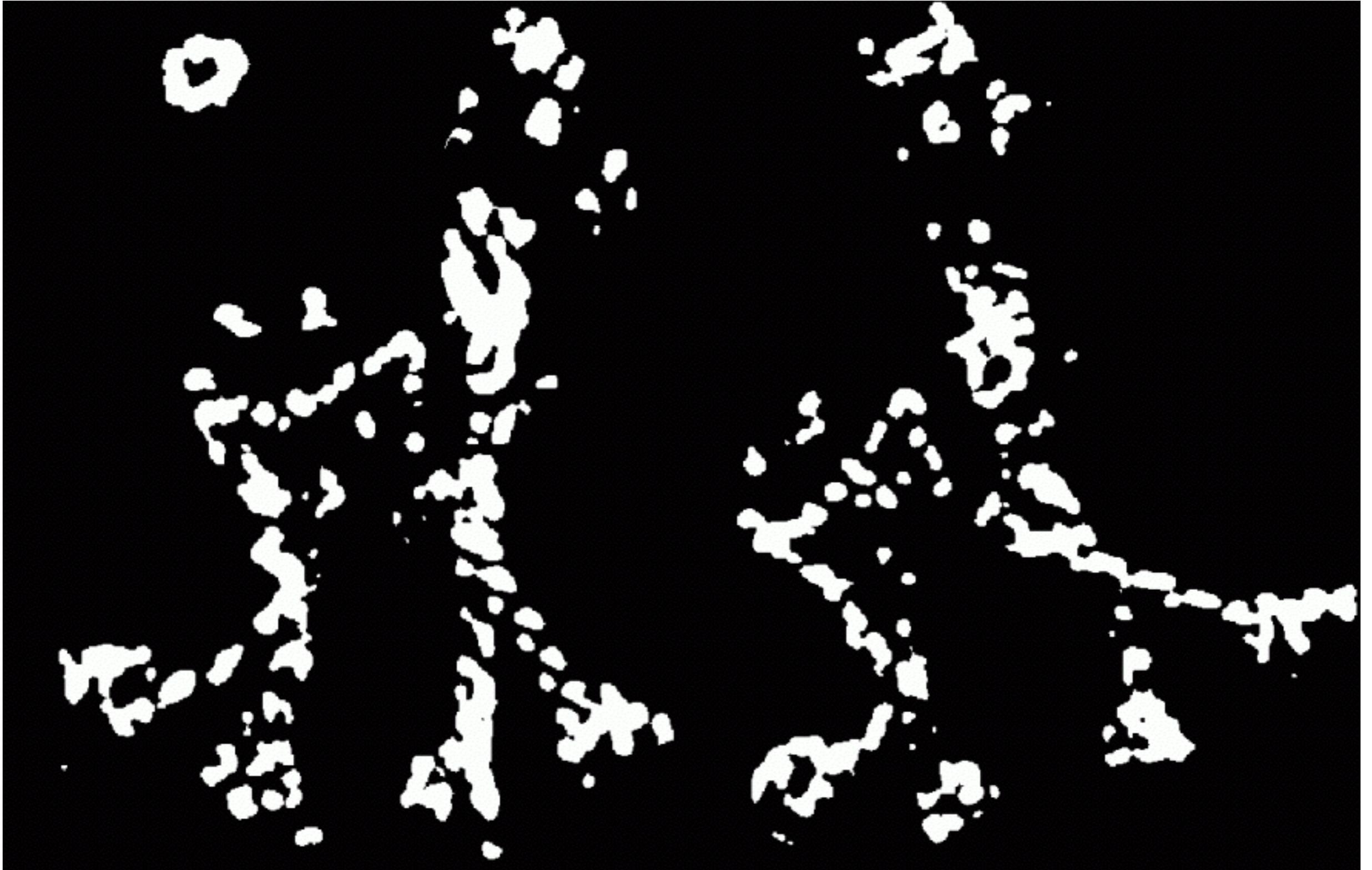
# Exemple: images



# Exemple: calculer R



Exemple: appliquer  $R > \text{seuil}$



# Exemple: maximum locaux

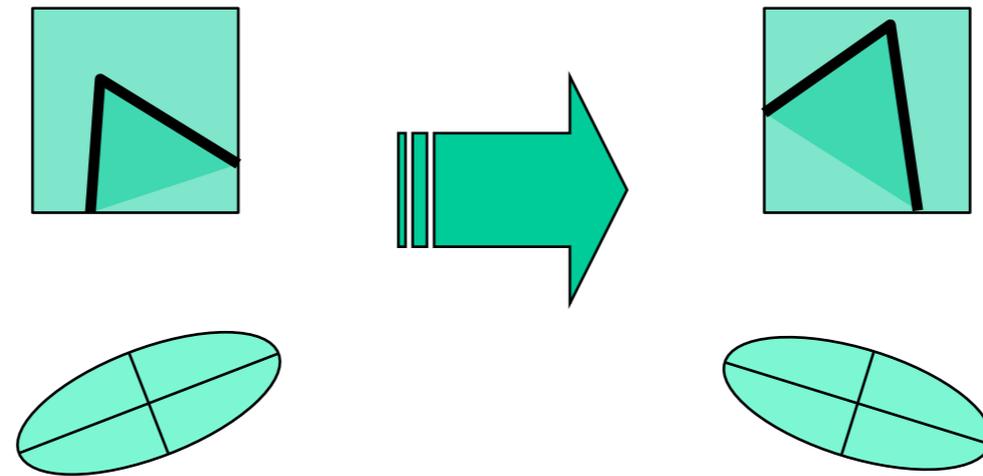


# Exemple: résultats!



# Détecteur Harris: propriétés

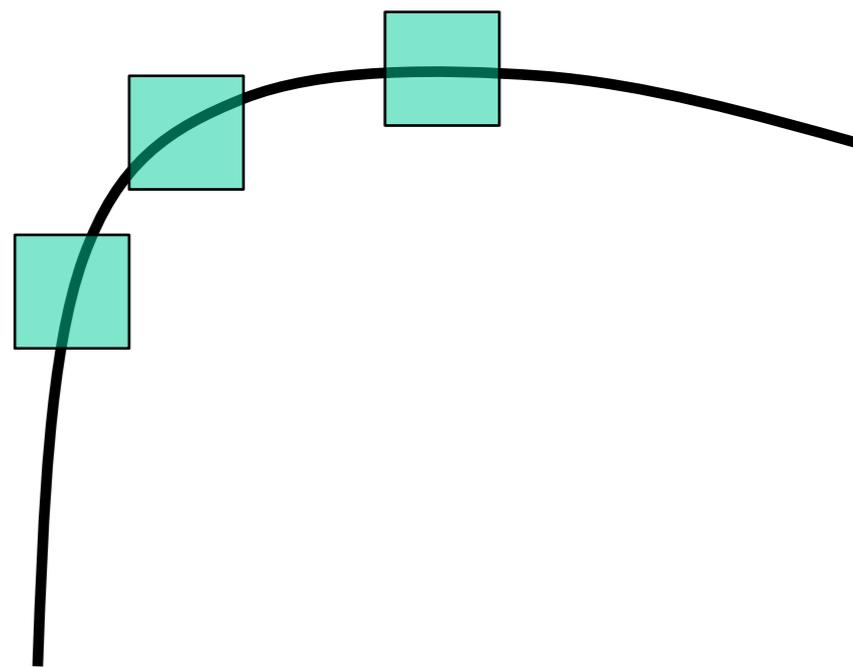
- Invariance à la rotation



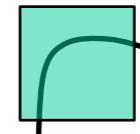
L'ellipse tourne, mais la longueur de ses axes  
(valeurs propres) restent les mêmes

# Détecteur Harris: propriétés

- Dépend de la taille de la fenêtre!



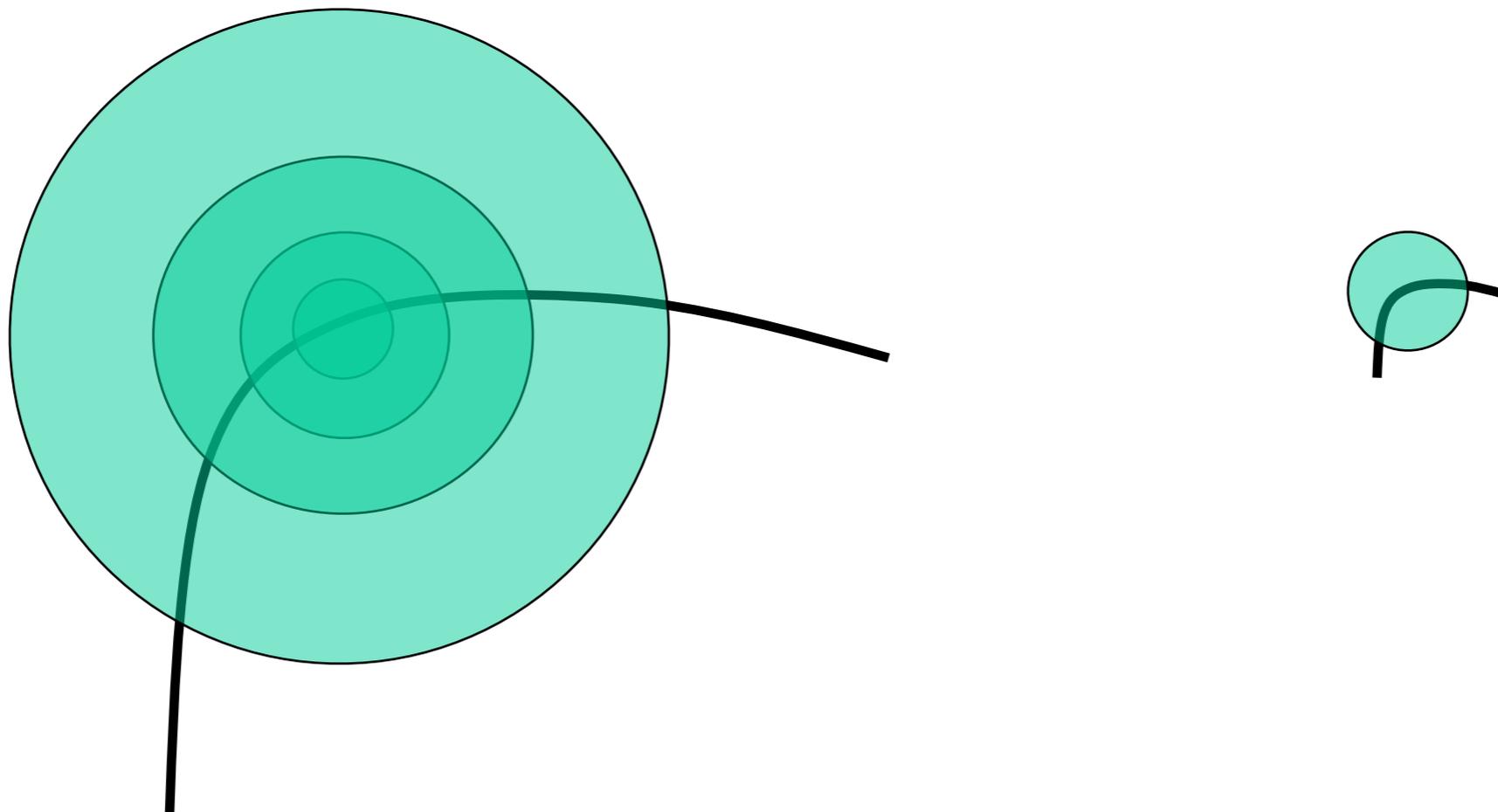
Tous les points sont  
des arêtes



Coin

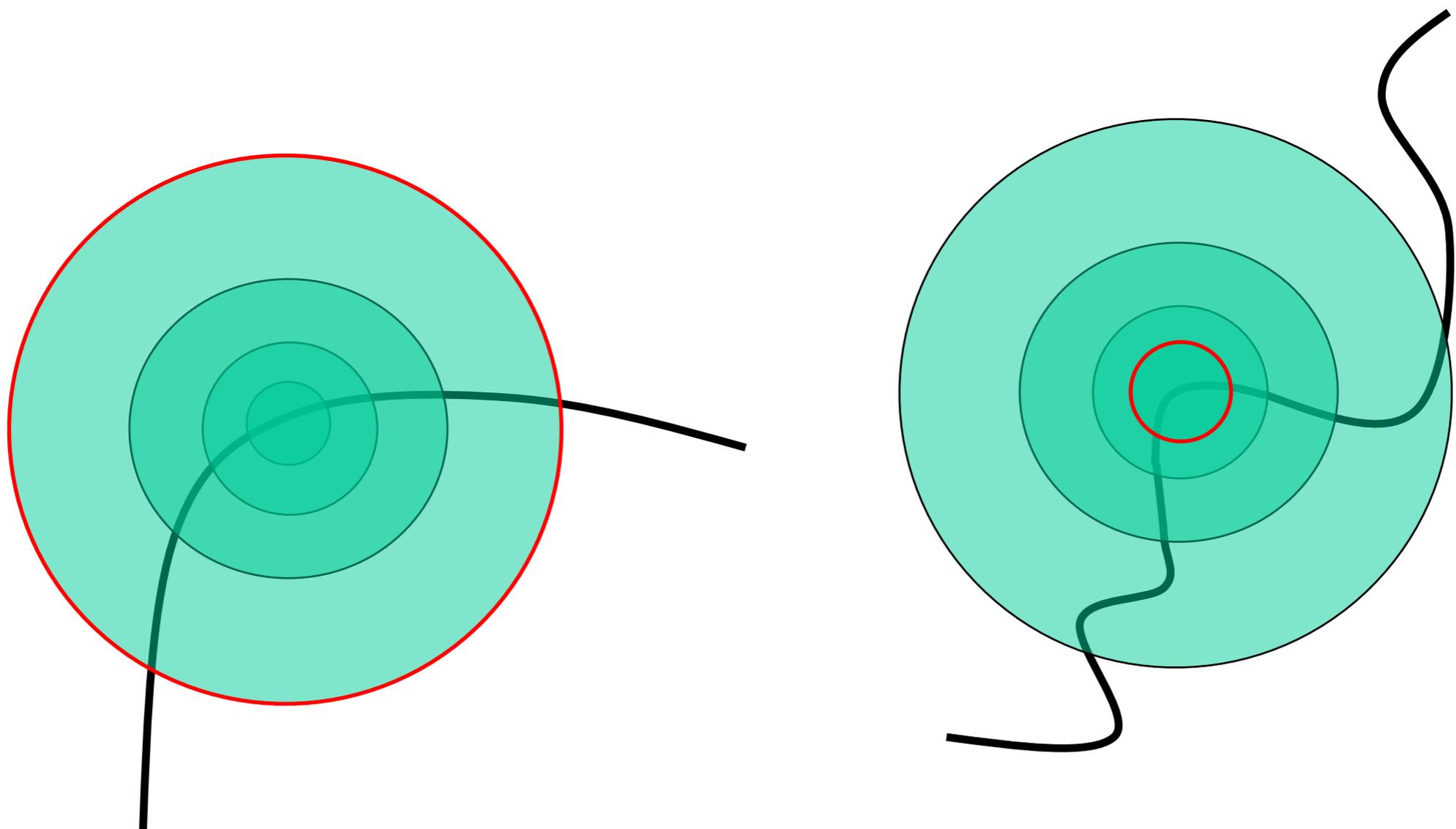
# Invariance à l'échelle

- Calculer réponse sur plusieurs échelles
- Réponse est similaire même si on réduit la taille de l'image



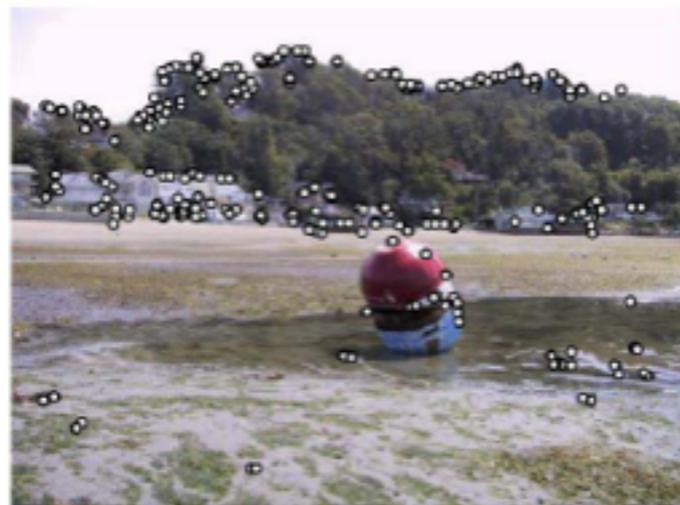
# Invariance à l'échelle

- Problème: comment déterminer la taille de la fenêtre indépendamment pour chaque image?
- Choisir la taille en fonction du “meilleur” coin



# Maximum locaux

- N coins les plus importants, distribués dans l'image
- Voir [Brown, Szeliski, Winder, CVPR'05] pour le TP4



(a) Strongest 250



(b) Strongest 500



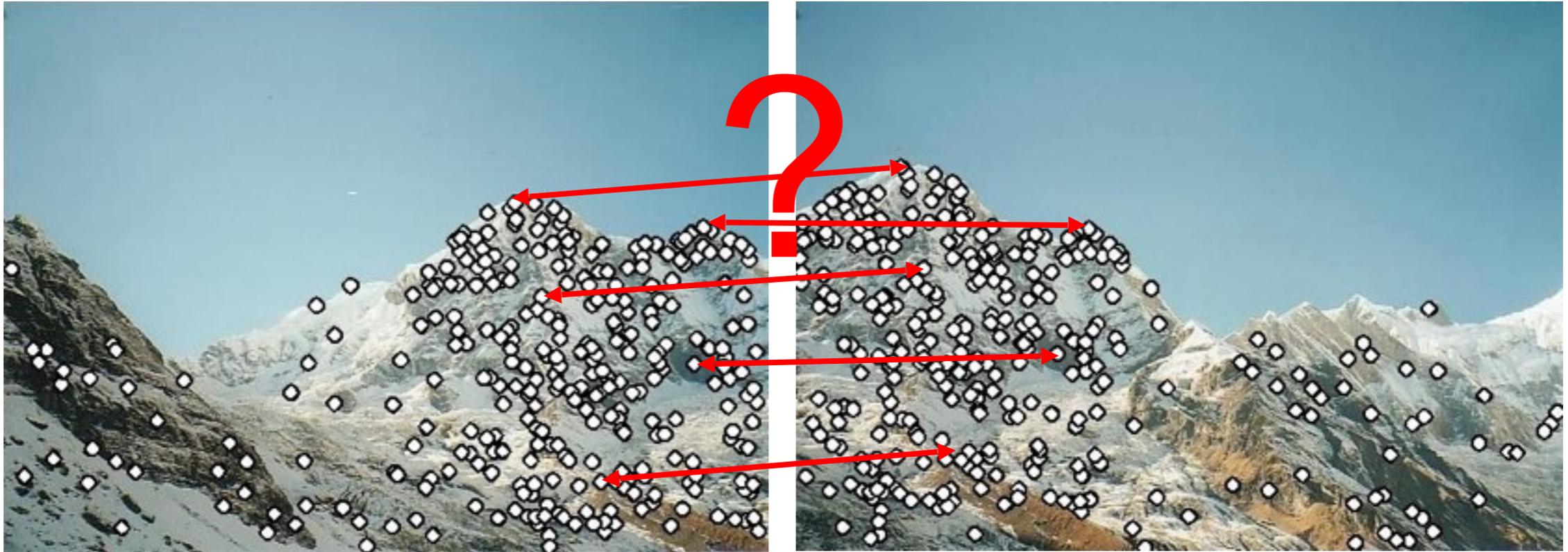
(c) ANMS 250,  $r = 24$



(d) ANMS 500,  $r = 16$

# Descripteurs

- Comment faire pour apparier nos points d'intérêt?

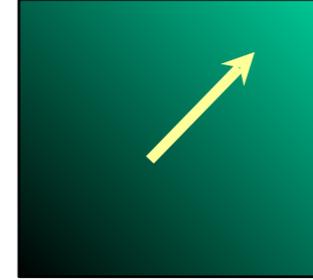
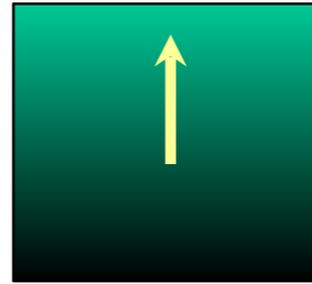


Descripteur doit être:  
distinct  
invariant

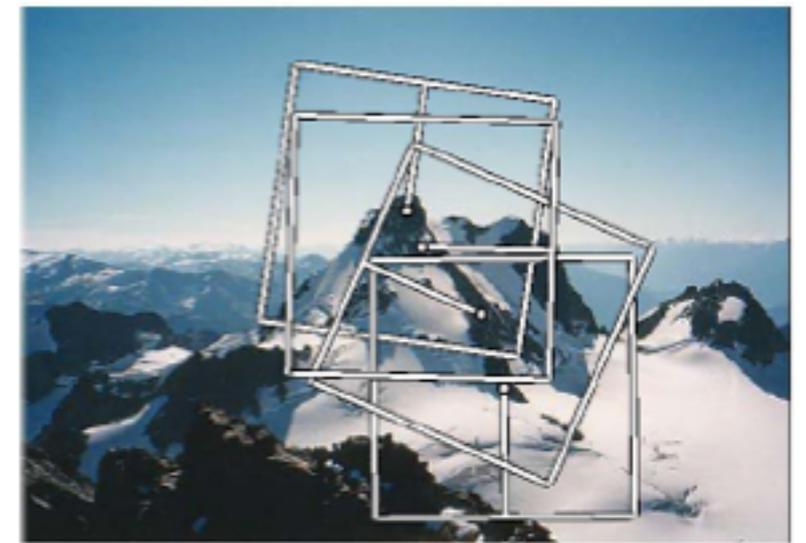
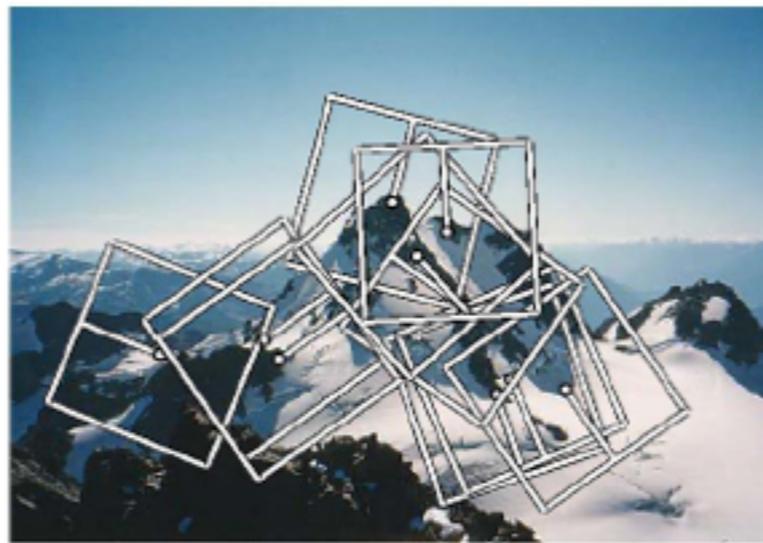
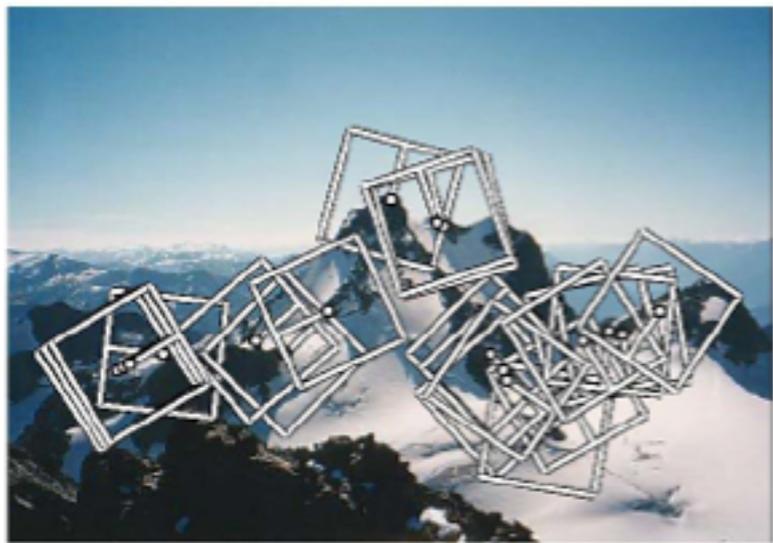
Voir [Brown, Szeliski, Winder, CVPR'05]

# Descripteur

- Orientation = gradient
- Calculer une fenêtre
  - Position et échelle (x, y, s) + orientation ( $\theta$ )

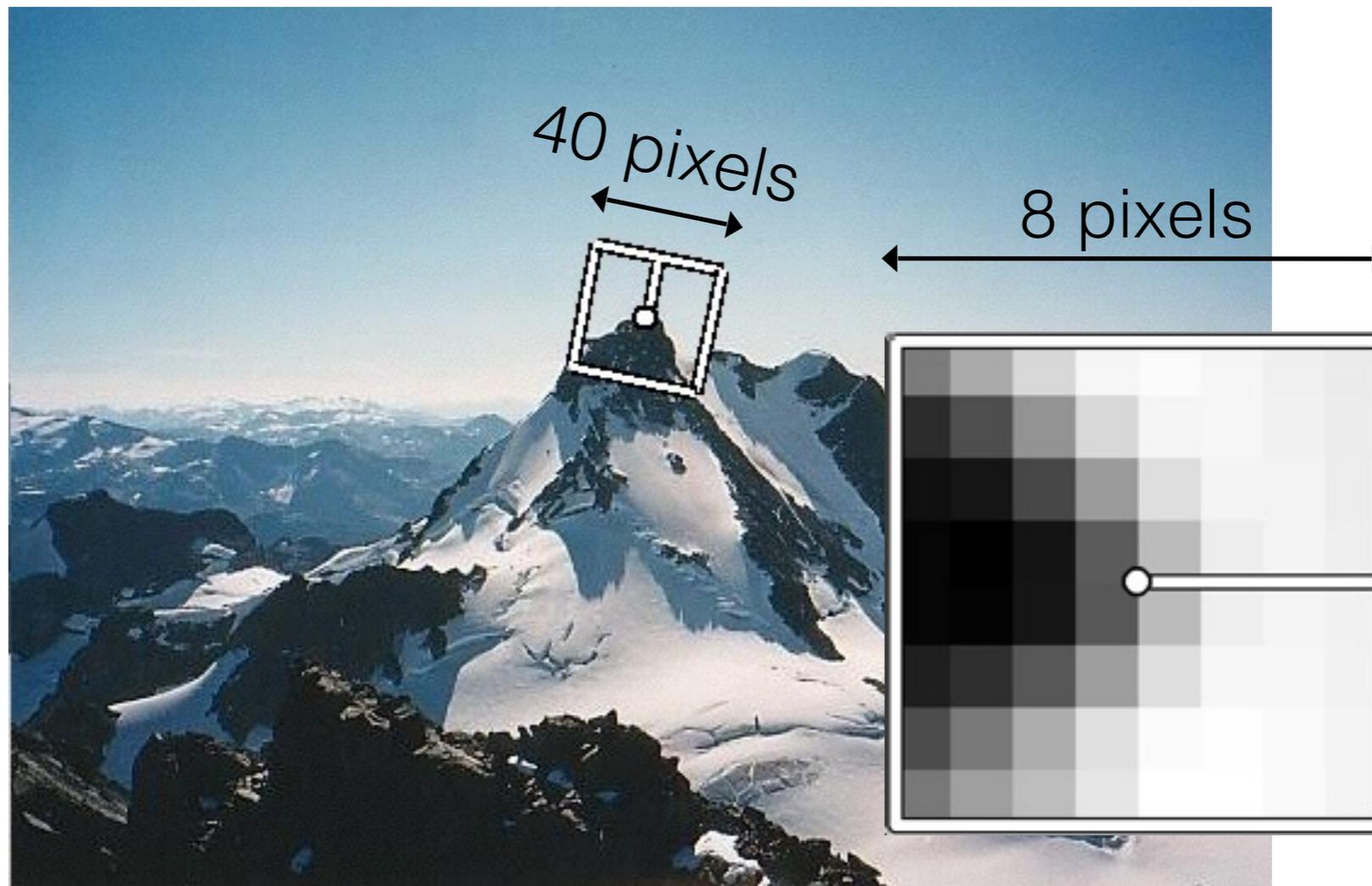


# Détections à plusieurs échelles

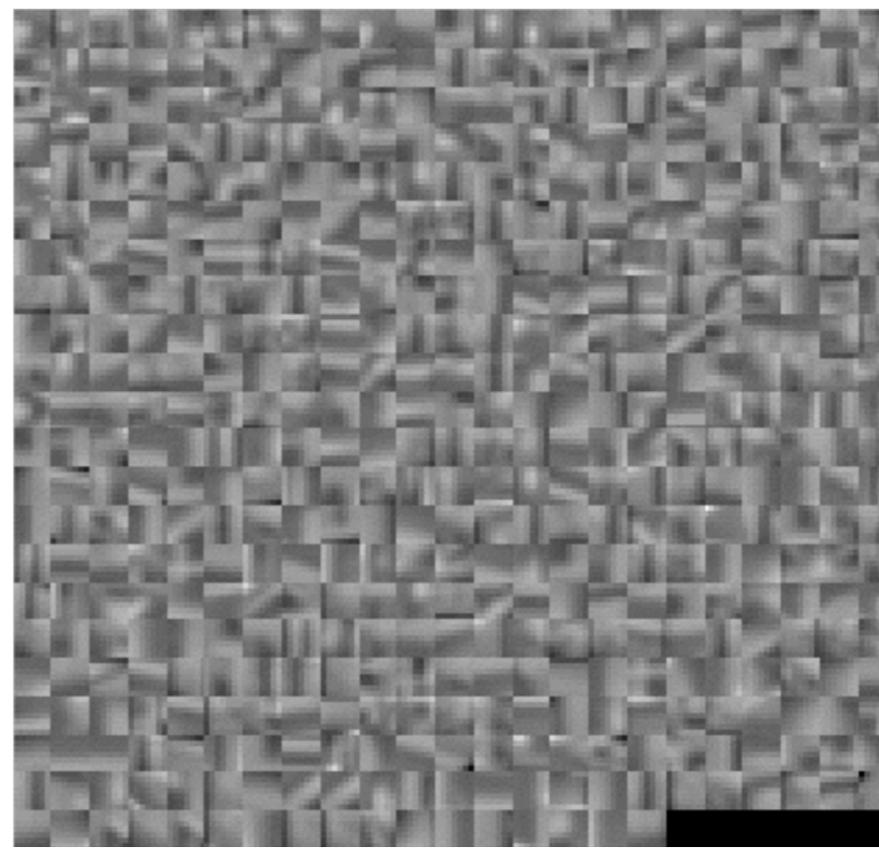
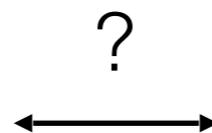
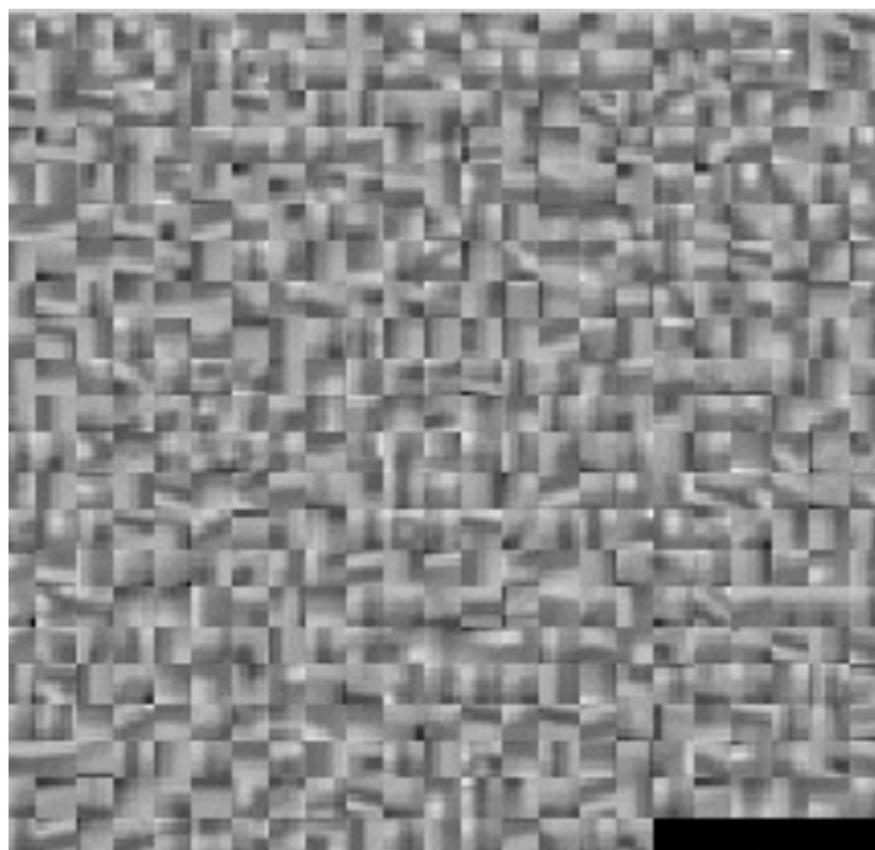
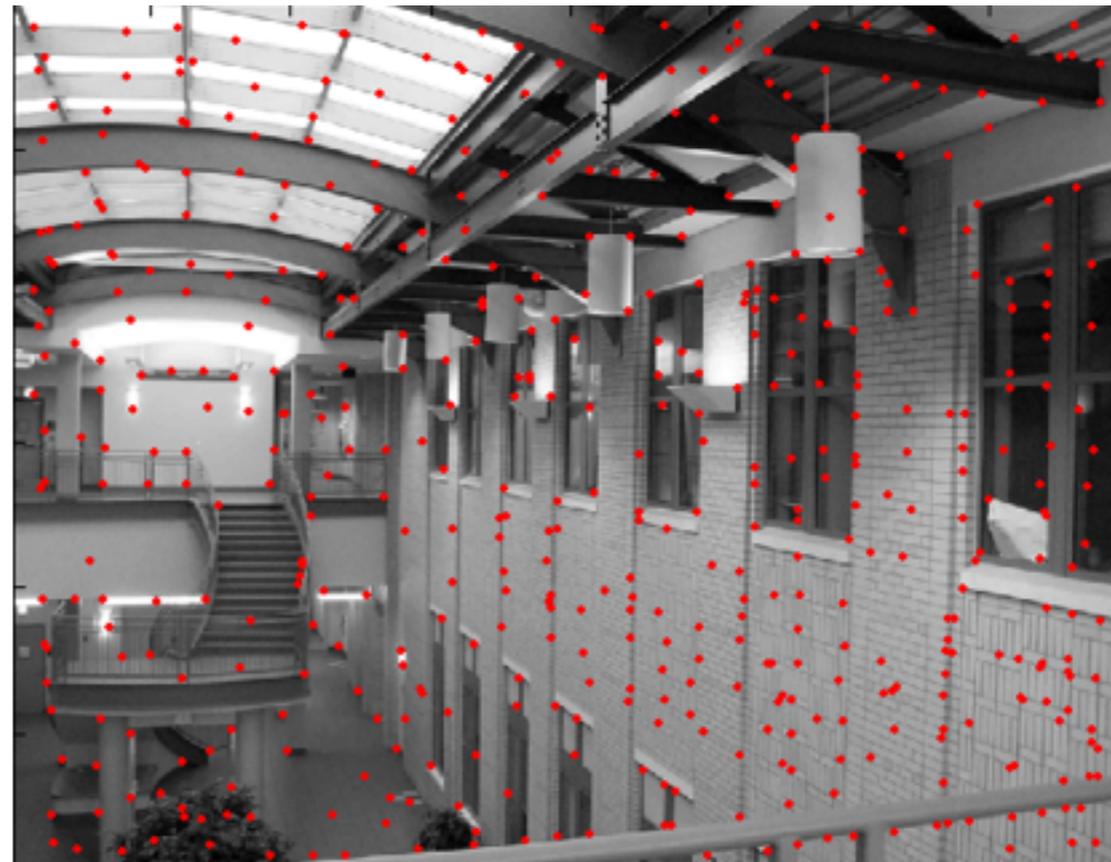
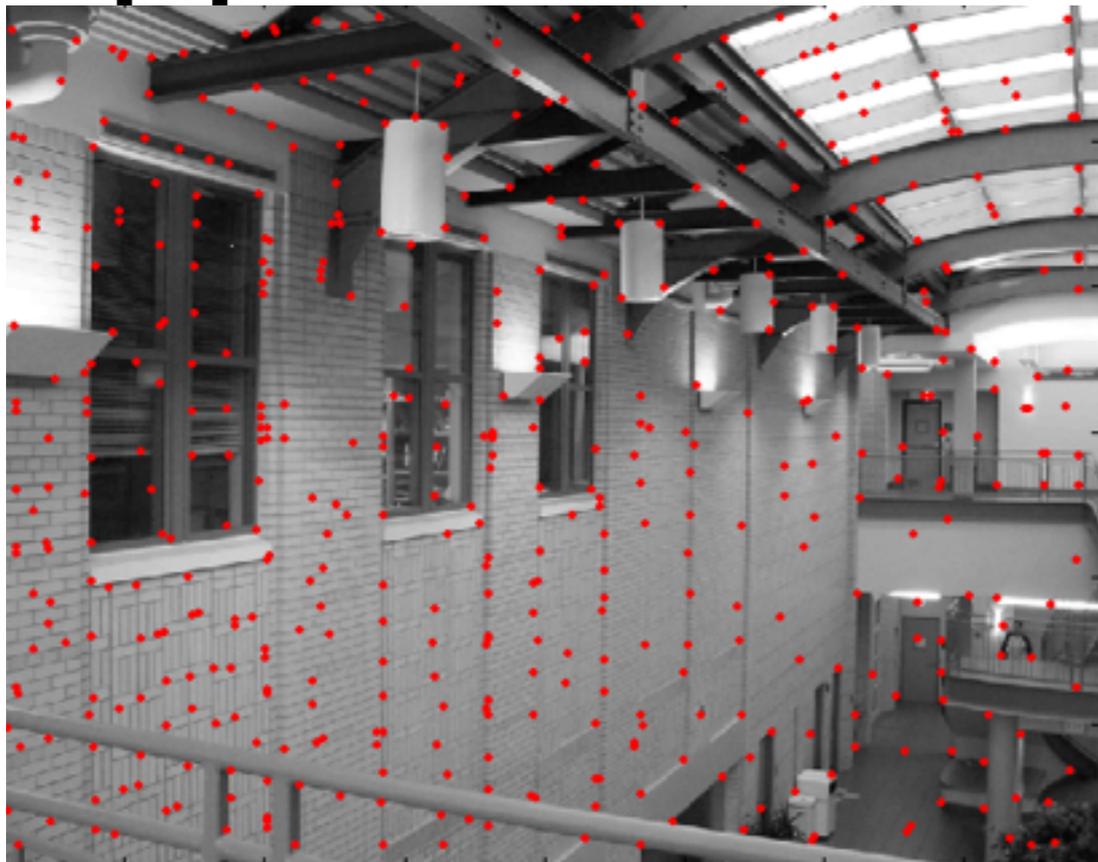


# Descripteur

- Extraire une fenêtre orientée de dimensions 8x8
  - Échantillonnée à 5x l'échelle (donc 40x40)
- Normalisation:  $I' = (I - \mu) / \sigma$



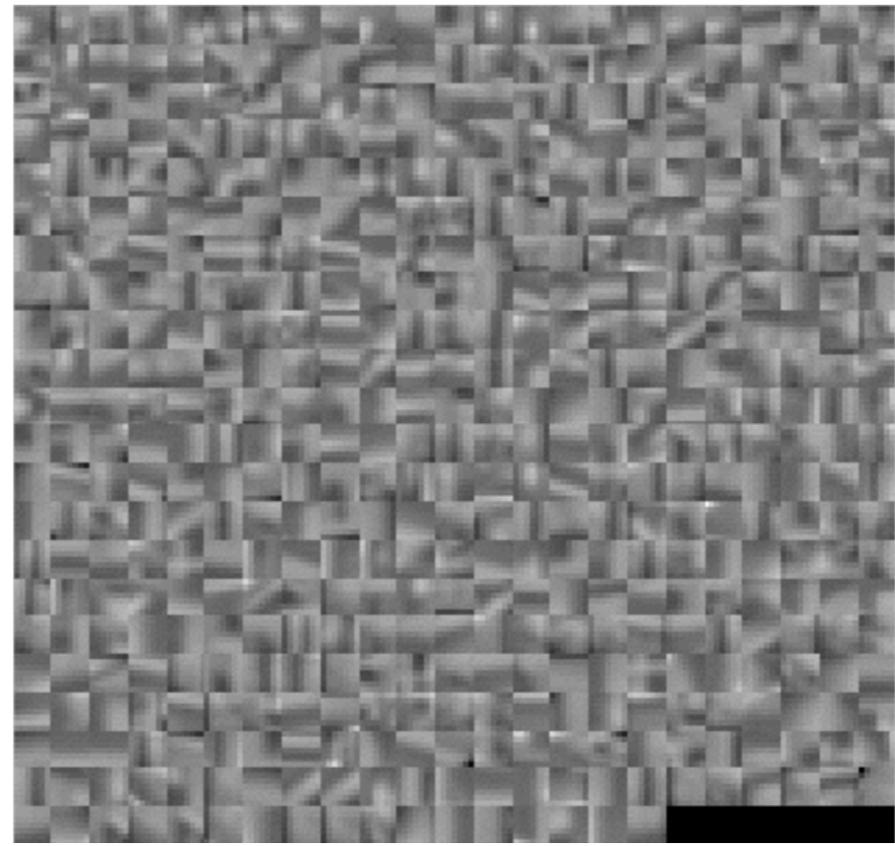
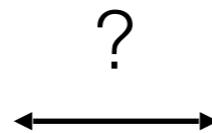
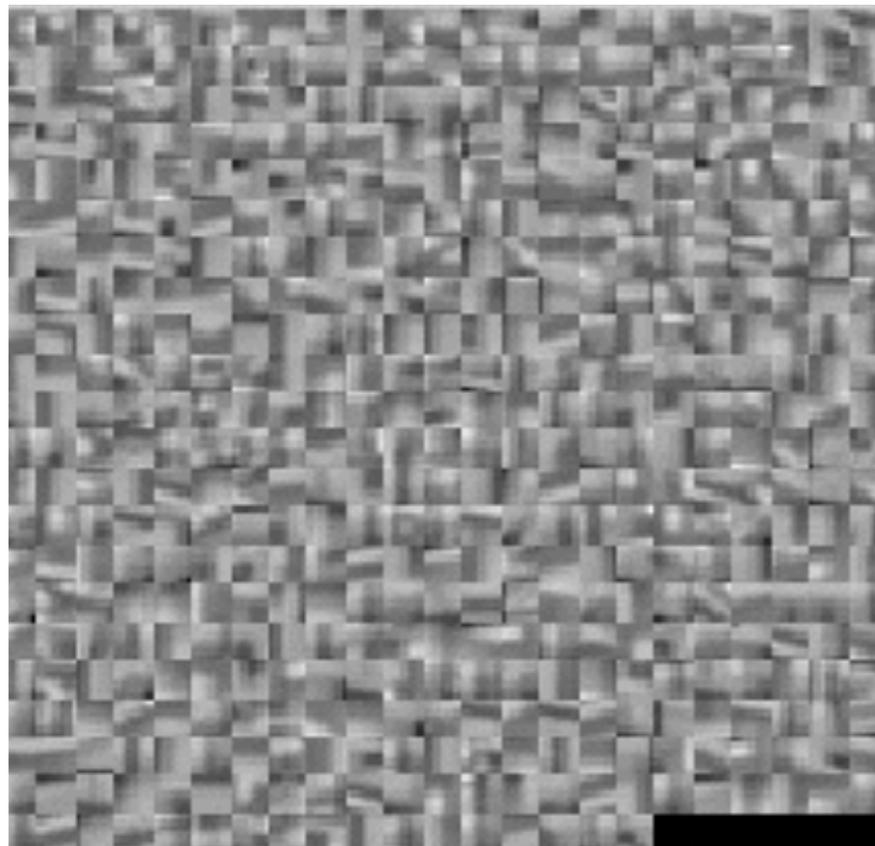
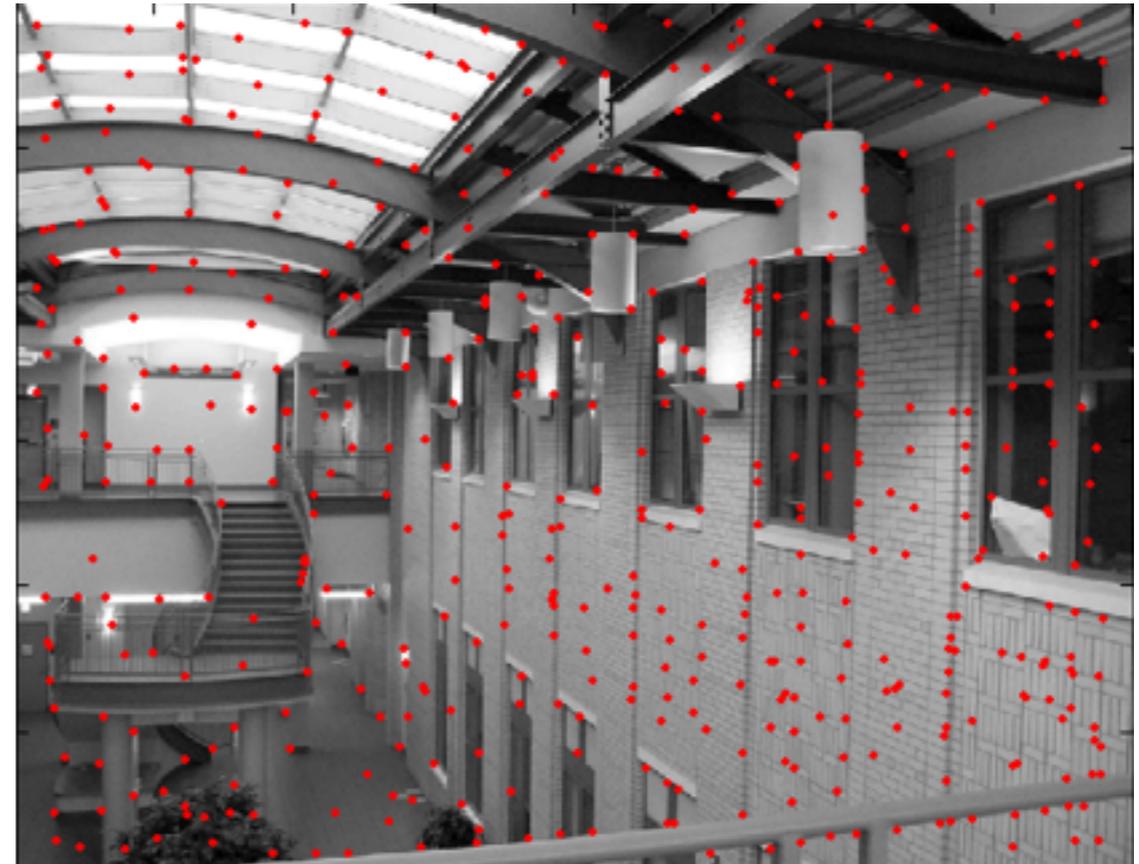
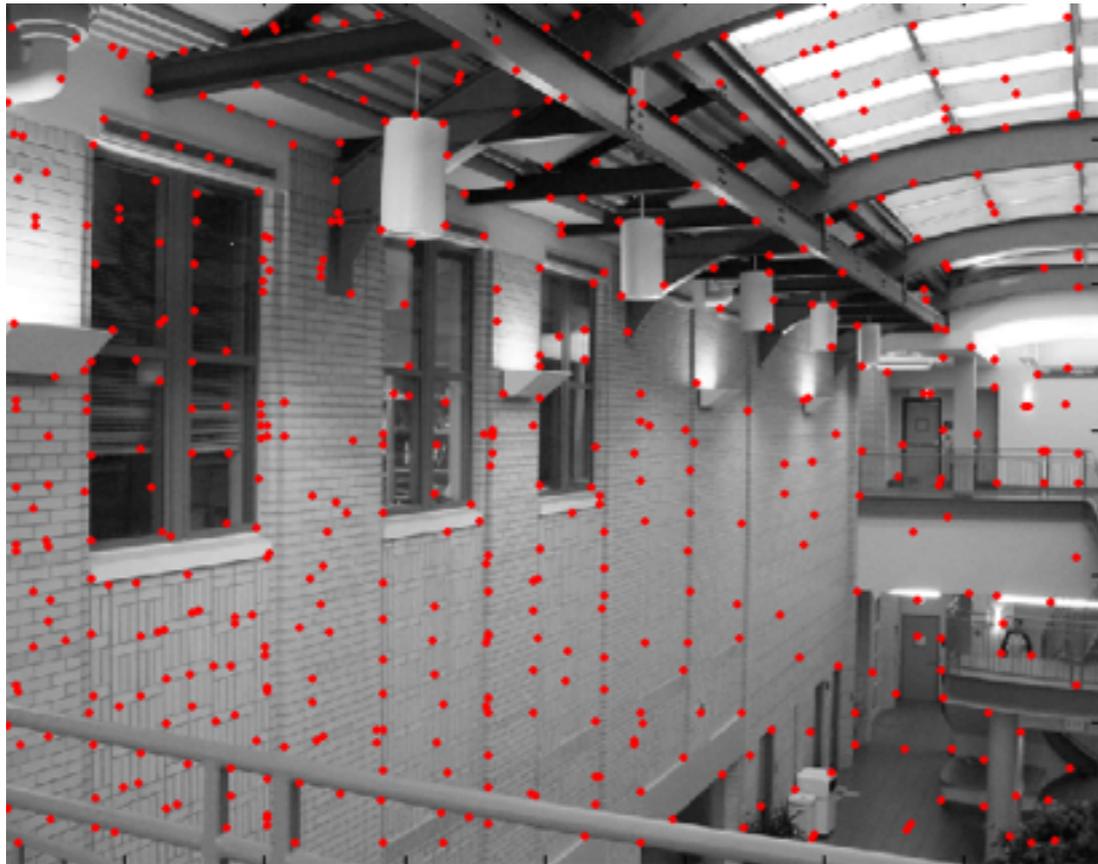
# Appariement



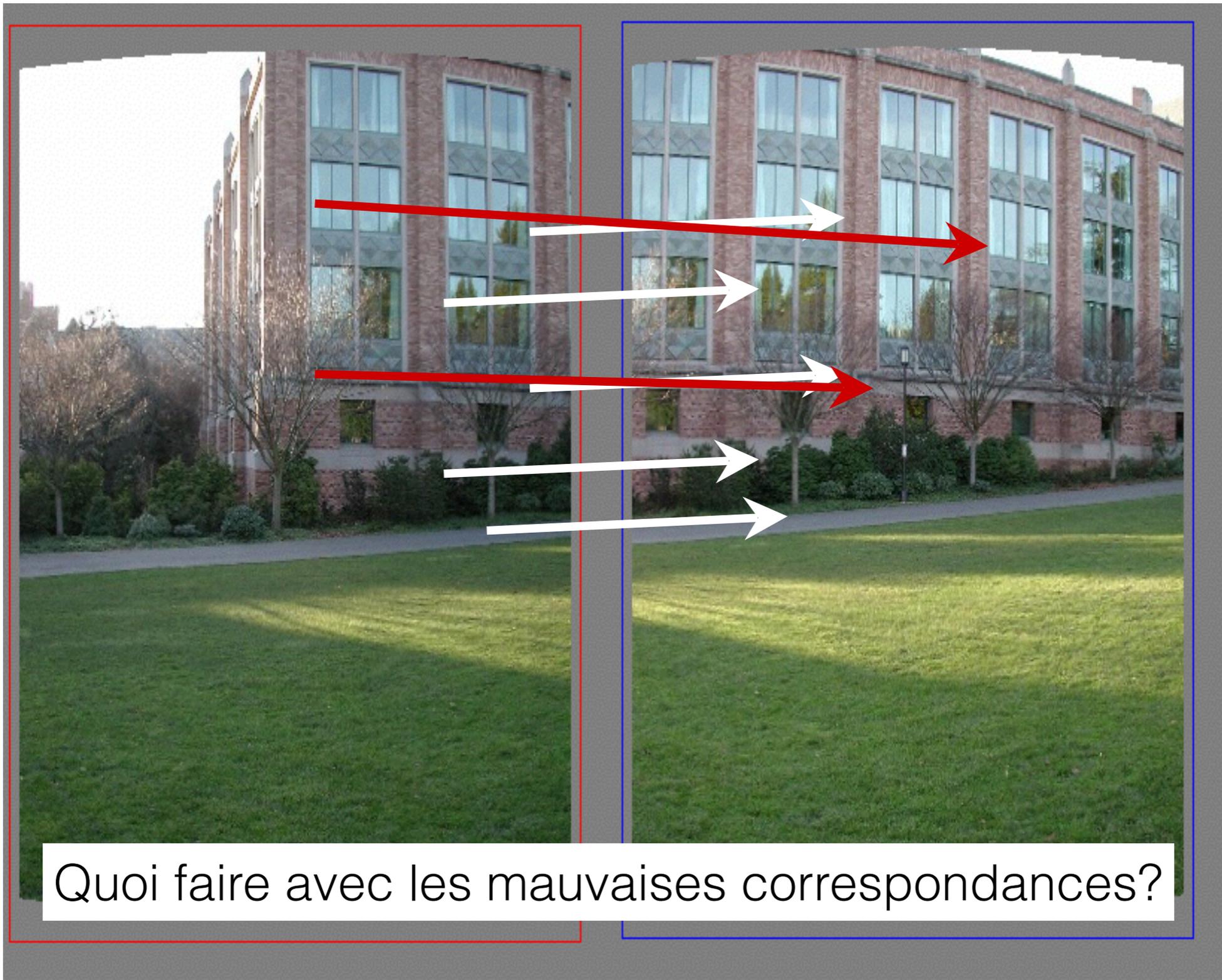
# Appariement

- Recherche exhaustive
  - Comparer chaque point à tous les points dans l'autre image et appliquer un seuil sur la différence
- Plus-proche-voisin
  - Exemple: "kd-tree" et variantes

# Quoi faire avec les aberrations?



# Appariement

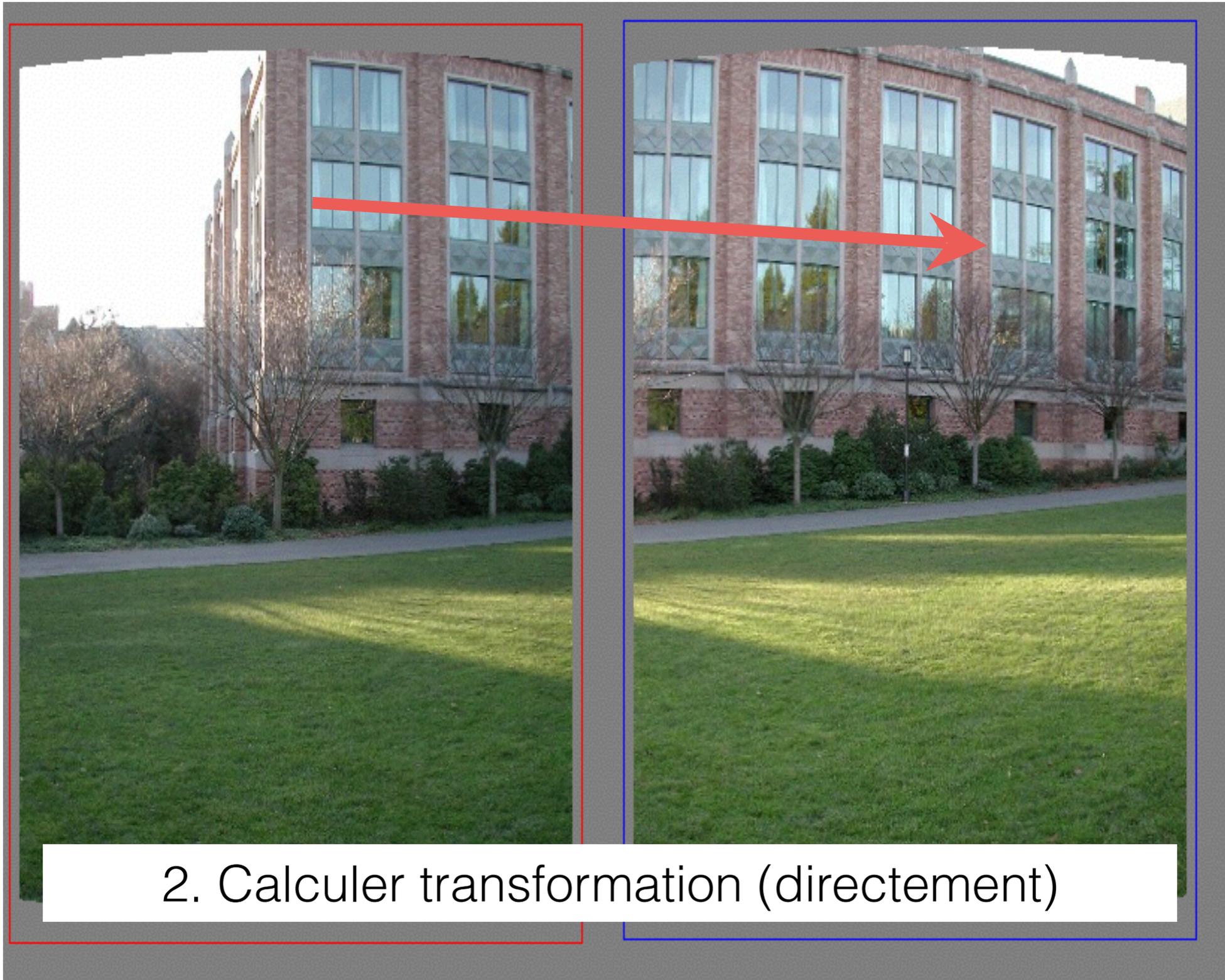


# “Random Sample Consensus”

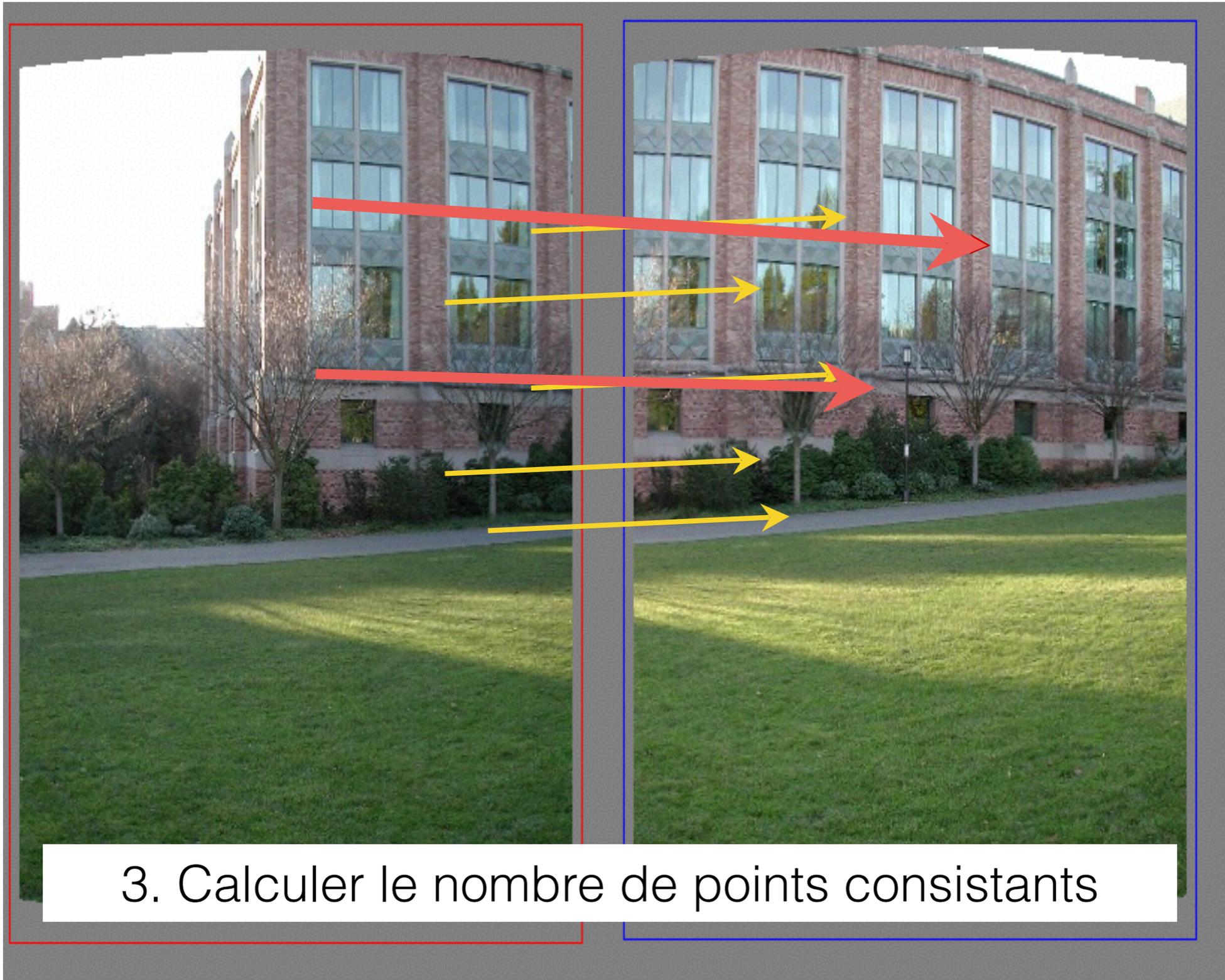


1. Sélectionner points au hasard (combien?)

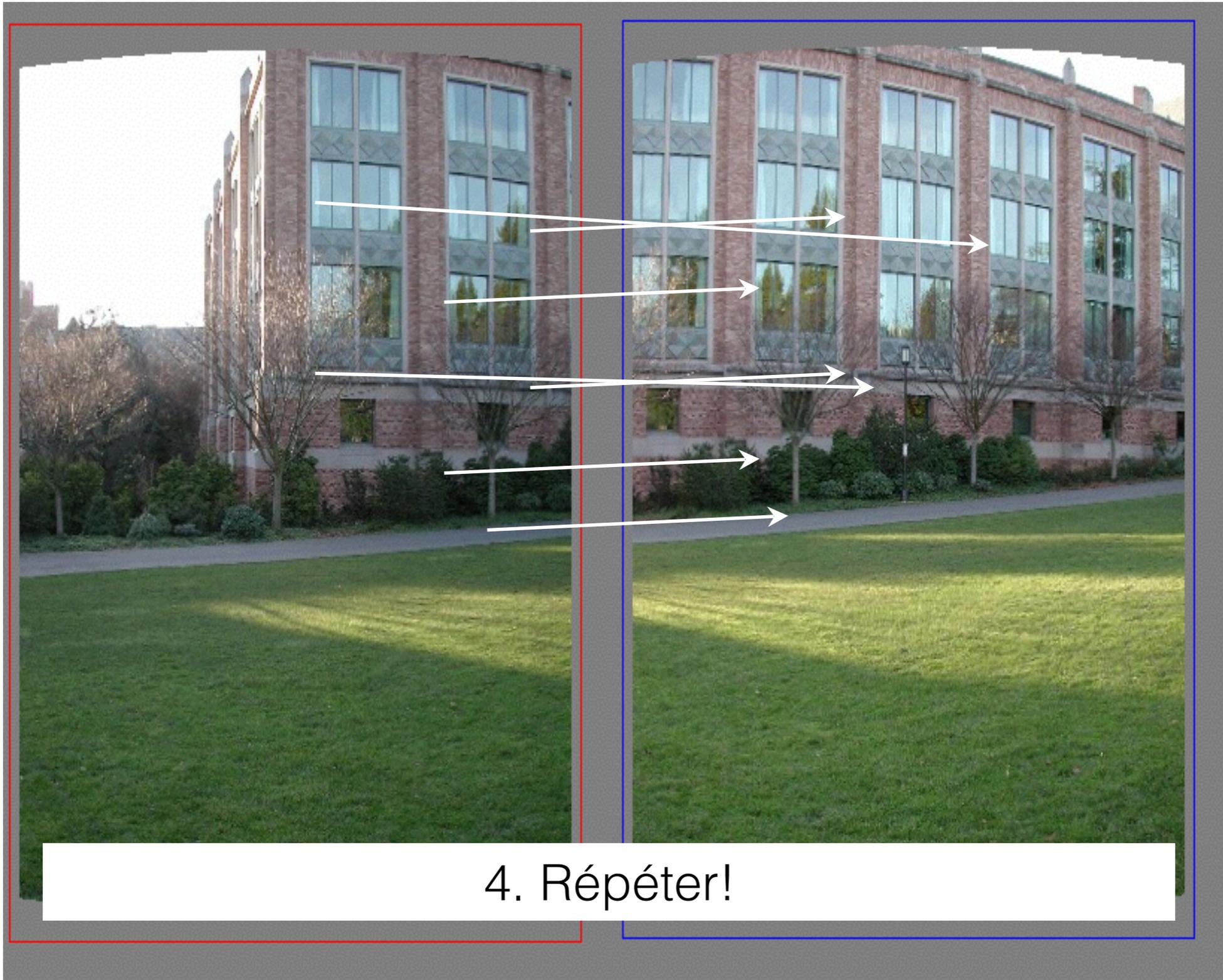
# “Random Sample Consensus”



# “Random Sample Consensus”



# “Random Sample Consensus”



# RANSAC pour homographies

- Pour N itérations:
  - Sélectionner points d'intérêt au hasard (combien?)
  - Calculer l'homographie H
  - Calculer le nombre de points où  $SSD(p', H p) < \epsilon$
- Garder l'itération qui correspond au plus grand nombre de points consistants
- Re-calculer H avec la méthode des moindres carrés avec tous les points consistants

# RANSAC

