

**QUELQUES APPLICATIONS DE L'EVOLUTION  
ARTIFICIELLE  
AU TRAITEMENT D'IMAGES:  
L'ALGORITHME DES MOUCHES**

Jean LOUCHET  
ARTENIA SARL

et

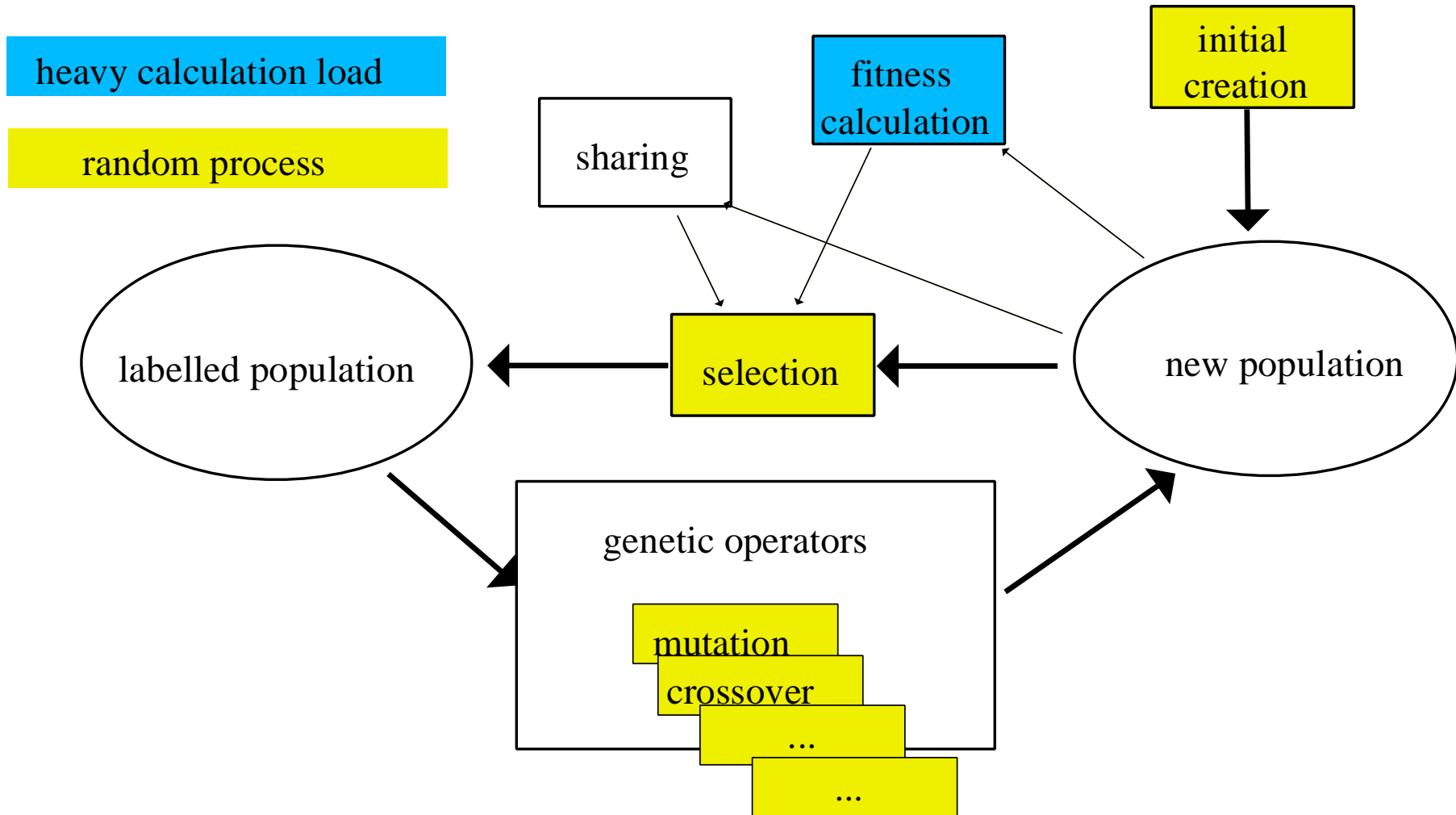
INRIA, Complex team, Rocquencourt (France)  
Jean.Louchet@gmail.com

1	Introduction: exploration d'espaces de paramètres par évolution artificielle	3
2	Retrouver des formes dans une image: revisitons la transformation de Hough.	11
	Evolutionnarisation de la transformation de Hough	16
	Houghvolution généralisée	20
3	Evolution parisienne, ou comment diviser le travail: l'algorithme des mouches.	22
	Une application en robotique mobile	26
	Une application en imagerie médicale	32
4	Conclusion	39

# 1 Introduction: exploration d'espaces de paramètres par évolution artificielle



*A la recherche des sommets: opérateurs fonctionnant en parallèle.*



*L'un des schémas possibles pour l'évolution artificielle.*

## DARWIN

- le “meilleur” survit plus longtemps
- hérédité

Codage - génotype et phénotype

Nature de l'espace de recherche (génotype): discret (algorithmes génétiques) ou continu (stratégies d'évolution)

Fonction de fitness (évaluation, qualité...)

$$\text{fitness } F : E \rightarrow R$$

$E$ : espace de recherche;  $R$ : corps des réels

$$x = \text{Argmax}(f) \Leftrightarrow [x' \neq x \Rightarrow f(x') < f(x)]$$

**Créer et faire (laisser?) évoluer une *population* dans l'espace de recherche, en imitant les lois de la nature**

Maxima locaux multiples

Autres approches

Fonction de coût ou de fitness?

Sharing (partage)? ou la régulation démographique...

## Opérateurs génétiques

- sélection
  - ranking (classement)
  - roulette
  - tournoi
  - seuil simple (normalement ça ne marche pas...)
- mutation
- croisement
  - échange de segments
  - barycentrique (ES seulement!)
- sharing
- raffinements:
  - préférences dans le choix de partenaires (mating preferences)
  - niches
  - co-évolution
  - diploidie, sexualité, gènes dominants/récessifs, etc.



*Résultat d'un croisement. L'opérateur n'est pas toujours gagnant.*



*Hans-Paul Schwefel (“Evolutionstrategie”)*

Coder par des nombres réels - culture d'ingénieurs?





*Riccardo Poli, David Goldberg (“Genetic algorithms”)*

Coder discret, booléen - faut-il vraiment laisser l’informatique aux informaticiens?



*Les autres pionniers: Wolfgang Banzhaf, William Langdon, Michèle Sebag.*

## 2 Retrouver des formes dans une image: revisitons la transformation de Hough.

$$y = ax + b$$

Dans le repère de coordonnées  $(a, b)$ , l'ensemble des droites qui contiennent le point  $(x, y)$  est une ligne droite d'équation:

$$b = (-x)a + y$$

L'espace repéré par les coordonnées  $(x, y)$  est souvent appelé le *plan*, et celui repéré par  $(a, b)$  le *plan dual* ou *espace des paramètres*.

Un petit tour au tableau noir...

ET OU EST PASSEE L'ISOTROPIE?

la réponse: voici une *bonne* équation de la droite:

$$\rho = x \cos\theta + y \sin\theta$$

Algorithme:

---

Initialiser le tableau  $H(\theta, \rho) = 0$

Pour chaque pixel  $(x, y)$

    si le pixel satisfait les critères (valeur, contraste, etc.) alors

        pour chaque  $\theta \in [0, 2\pi]$ :  $H(\theta, x \cos\theta + y \sin\theta) + = 1$

// chaque "pixel intéressant" crée une arche de sinusoïde dans l'espace d'accumulation de Hough.

$(\theta, \rho) = \text{Argmax } H$

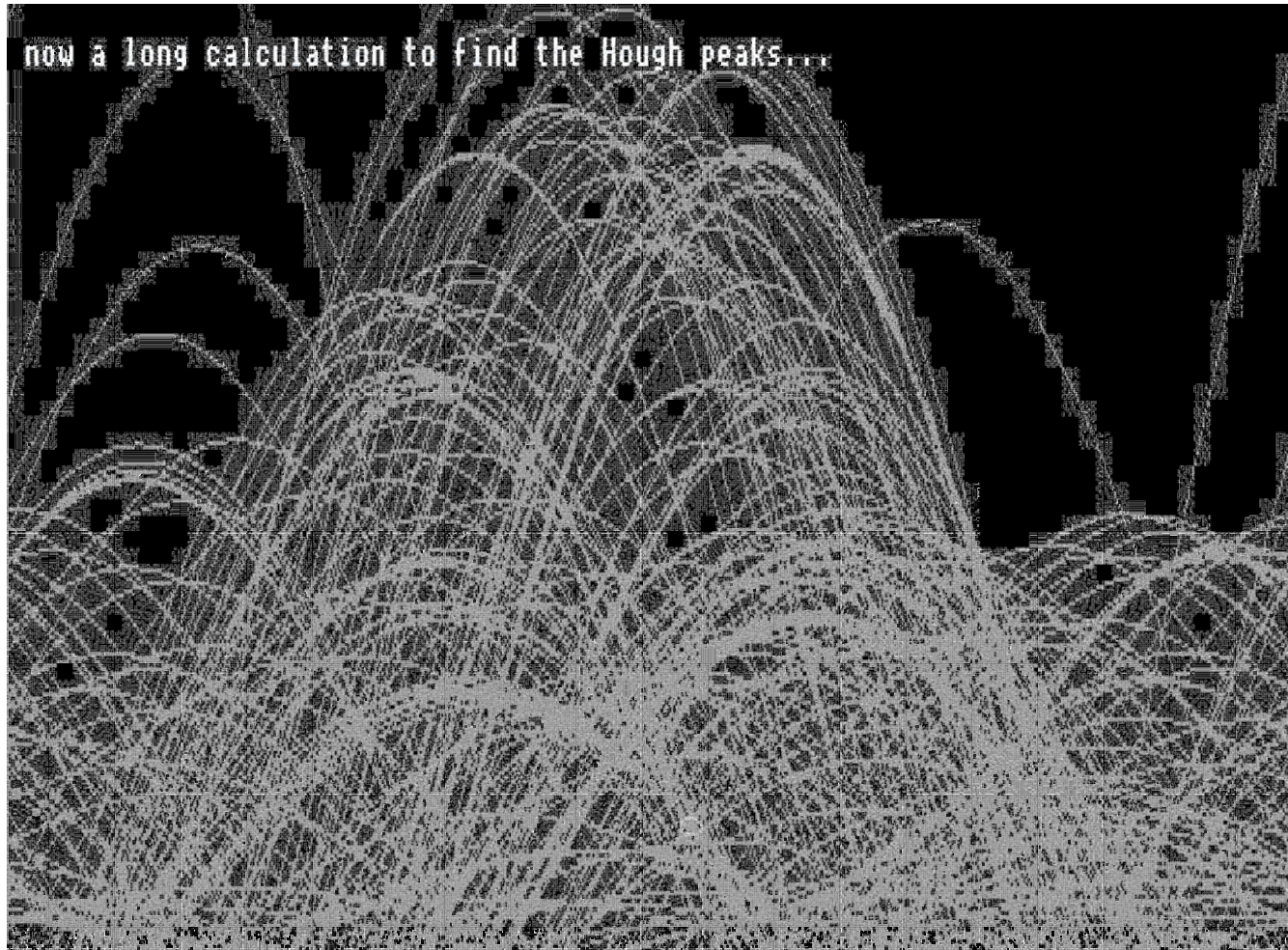
// donne l'équation de droite correspondant au plus important alignement.

---

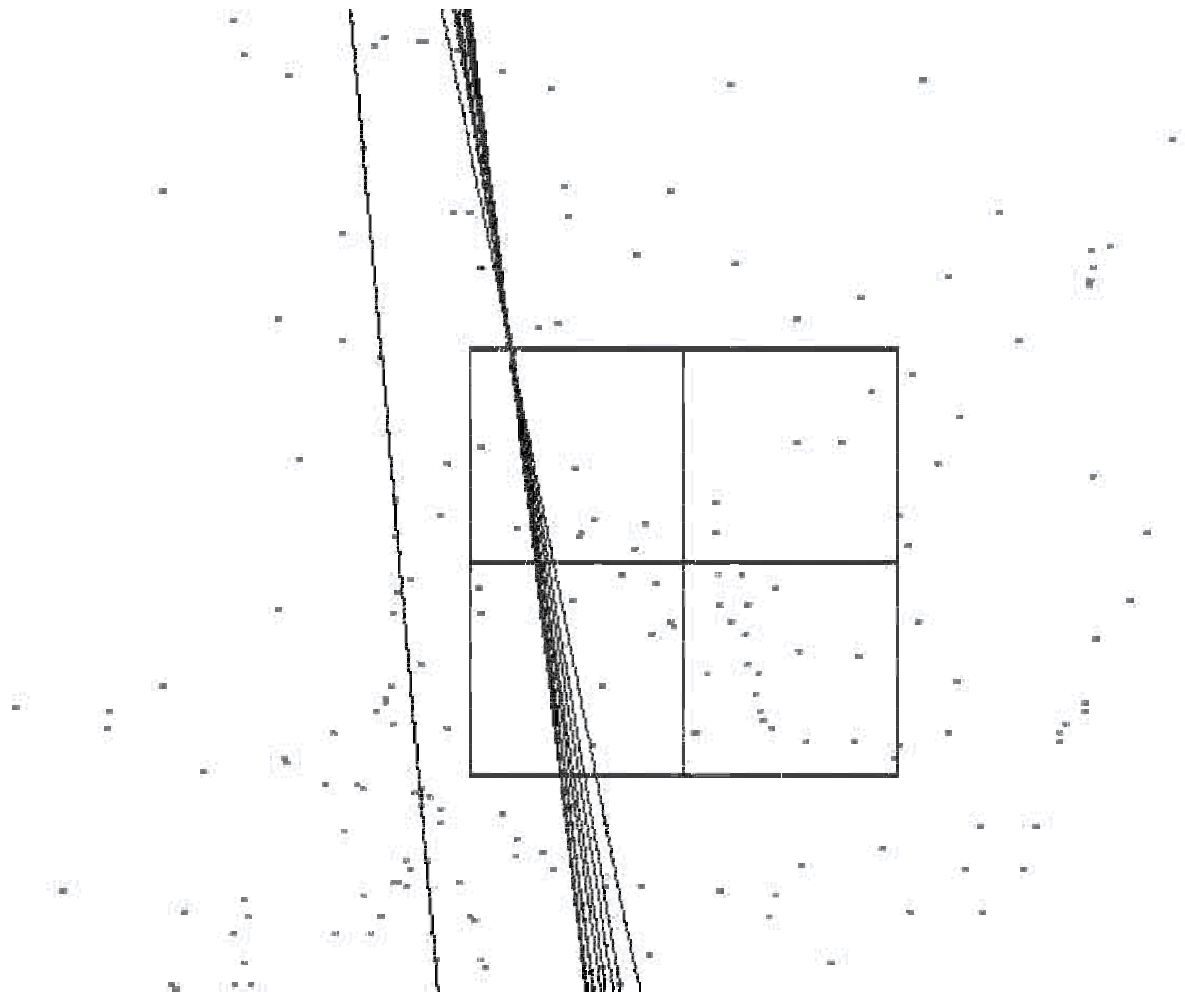


*Une application en archéologie: carte des monuments paléolithiques dans le Devonshire (Peter Brough)*





*L'accumulateur de Hough*



*Alignements détectés*

# Evolutionarisation de la transformation de Hough

Idée: pourquoi pas explorer directement (mais intelligemment) l'espace des paramètres, plutôt que l'utiliser comme un accumulateur?

Algorithme:

---

Créer une population d'individus avec les chromosomes  $(\theta, \rho)$

Initialiser la population au hasard, avec  $\theta \in [0, 2\pi]$  et  $\rho \leq \rho_{max}$

Lancer une ES standard avec la fitness suivante:

*fitness = nombre de pixels sur la droite correspondante, vérifiant le critère.*

Mutation: Gaussienne

Croisement: barycentrique

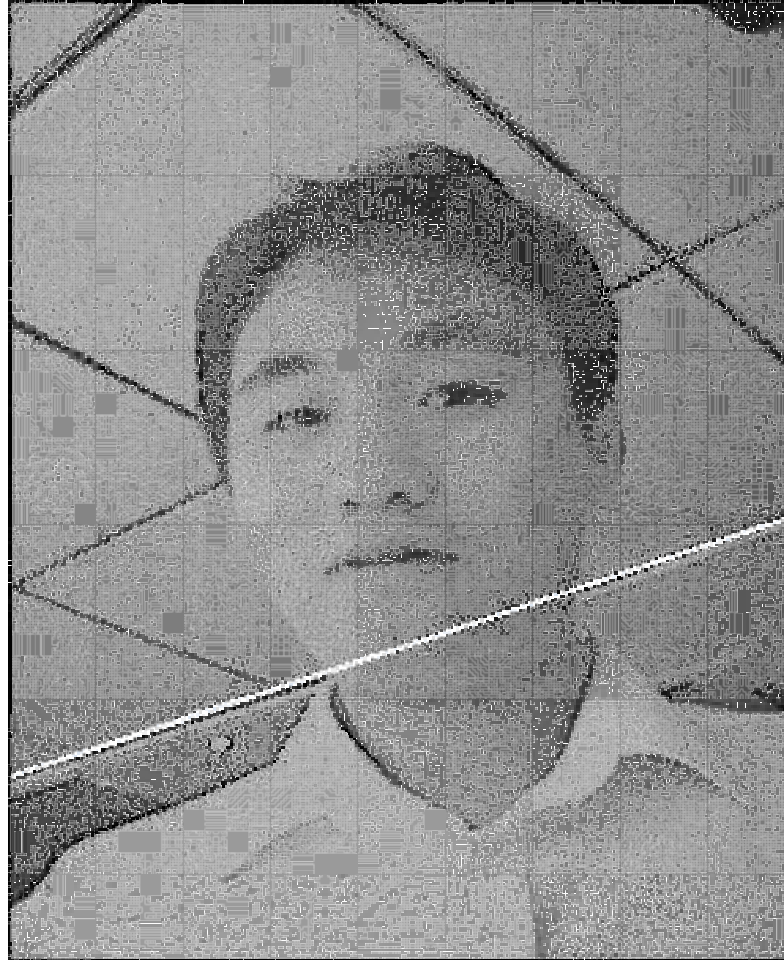
Pas de sharing

Sélection: tournoi

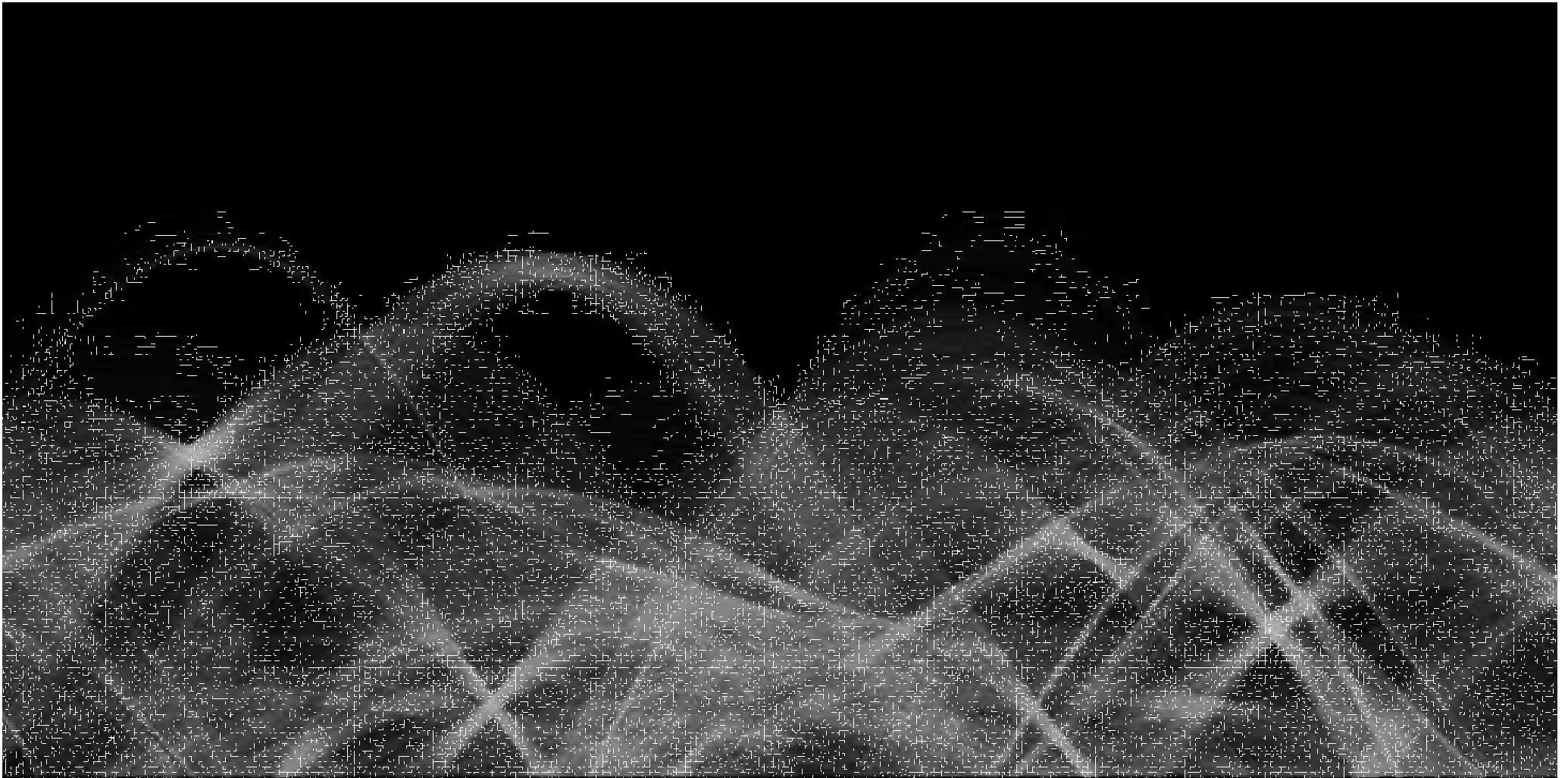
Raffinement: fitness rapide à base de sondage

//fonction de fitness indéterministe: un avantage...

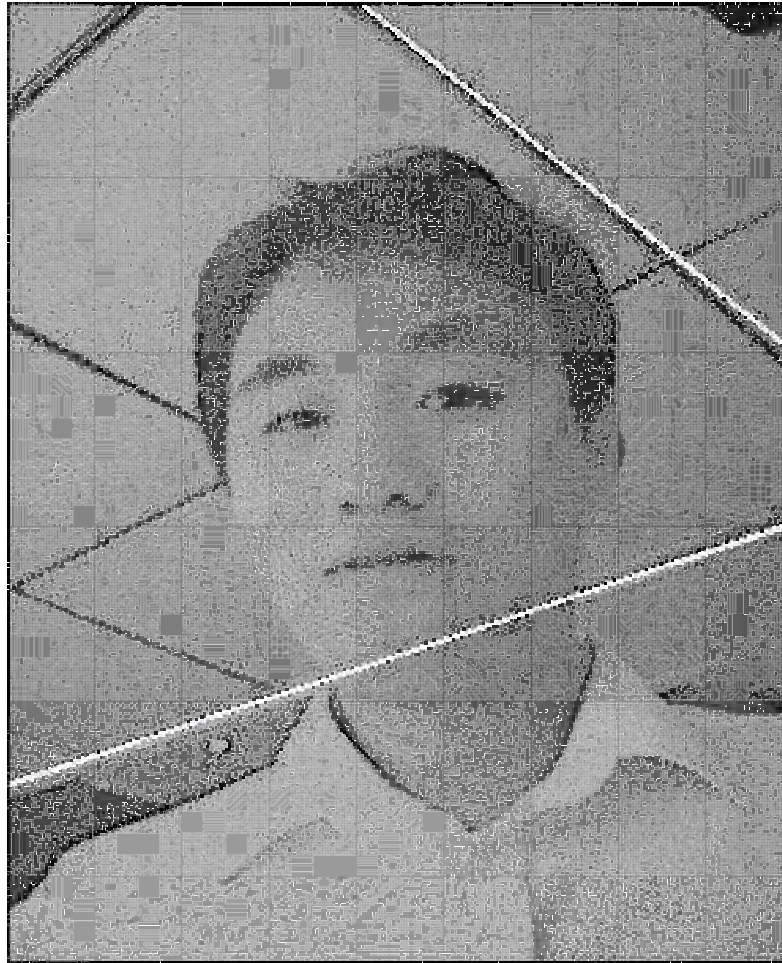




*Détection d'alignements façon classique: M. Cheng...*



*et l'accumulateur de Hough  $(\theta, \rho)$  de M. Cheng.*



*Hough évolutionnaire: les deux individus dominants.*

# Houghvolution généralisée

La transformation originale de Hough est limitée à 2 (peut-être 3) paramètres.

La version évolutionnaire est virtuellement *illimitée* en termes de dimension de l'espace de recherche. Algorithme:

Image  $I(x, y)$

Forme contrôlée par  $n$  paramètres  $(x_1, x_2, \dots, x_n)$  et définie par la fonction  $P$  :

$$P((x, y)(x_1, x_2, \dots, x_n)) \in \{0, 1\}$$

---

Créer une population d'individus, chaque chromosome est de la forme  $(x_1, x_2, \dots, x_n)$

Initialiser aléatoirement la population, avec chaque  $x_i \in [min_i, max_i]$  (intervalles de validité)

Lancer une stratégie d'évolution (ES) standard, avec la fitness suivante:

$$Fitness(x_1, x_2, \dots, x_n) = \text{nombre de pixels } (x, y) \text{ tels que}$$

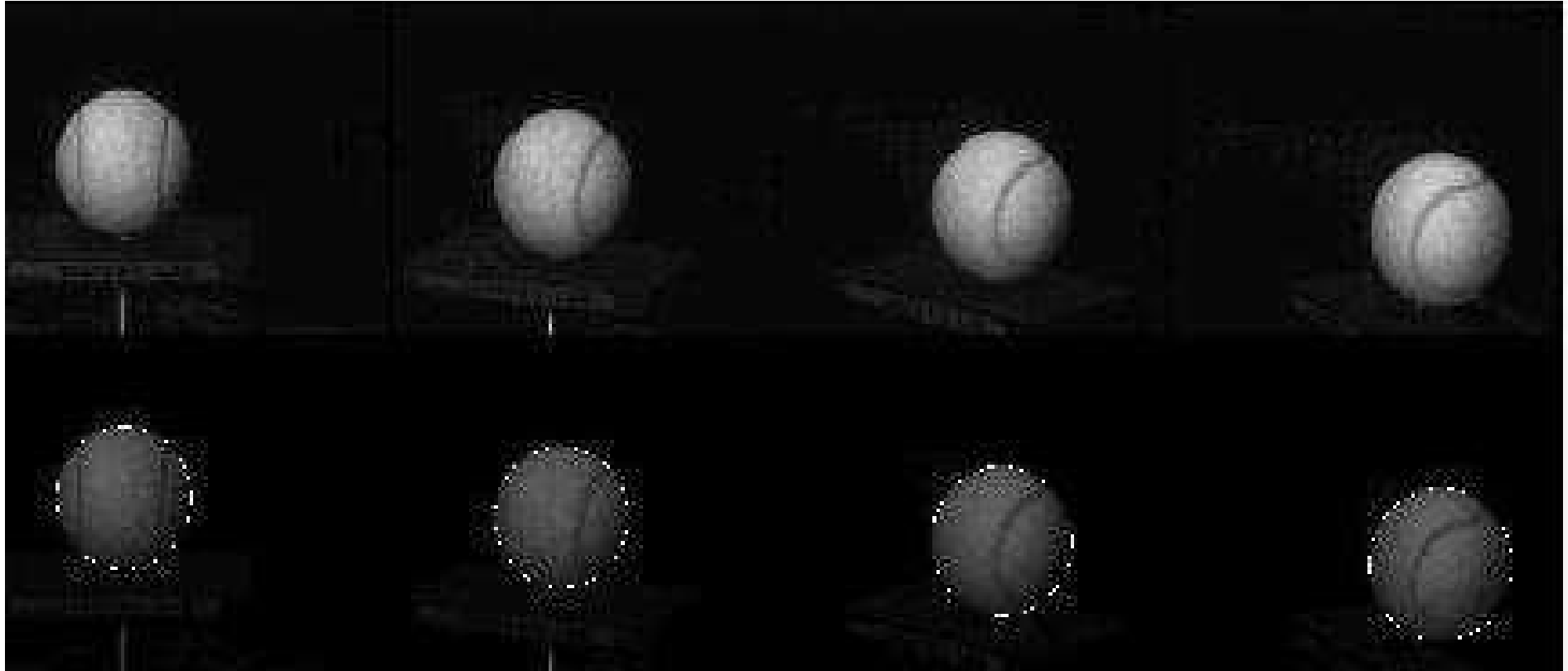
$$P((x, y)(x_1, x_2, \dots, x_n)) = 1$$

Mutation: Gaussienne

Croisement: barycentrique

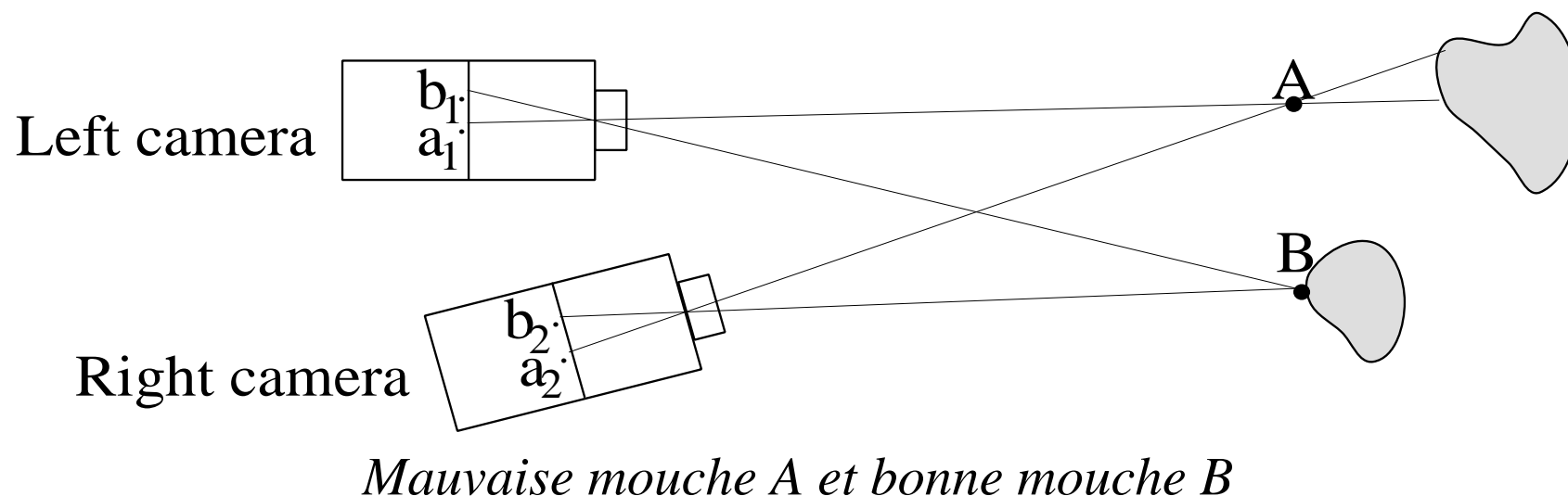
Pas de sharing

Selection: tournoi



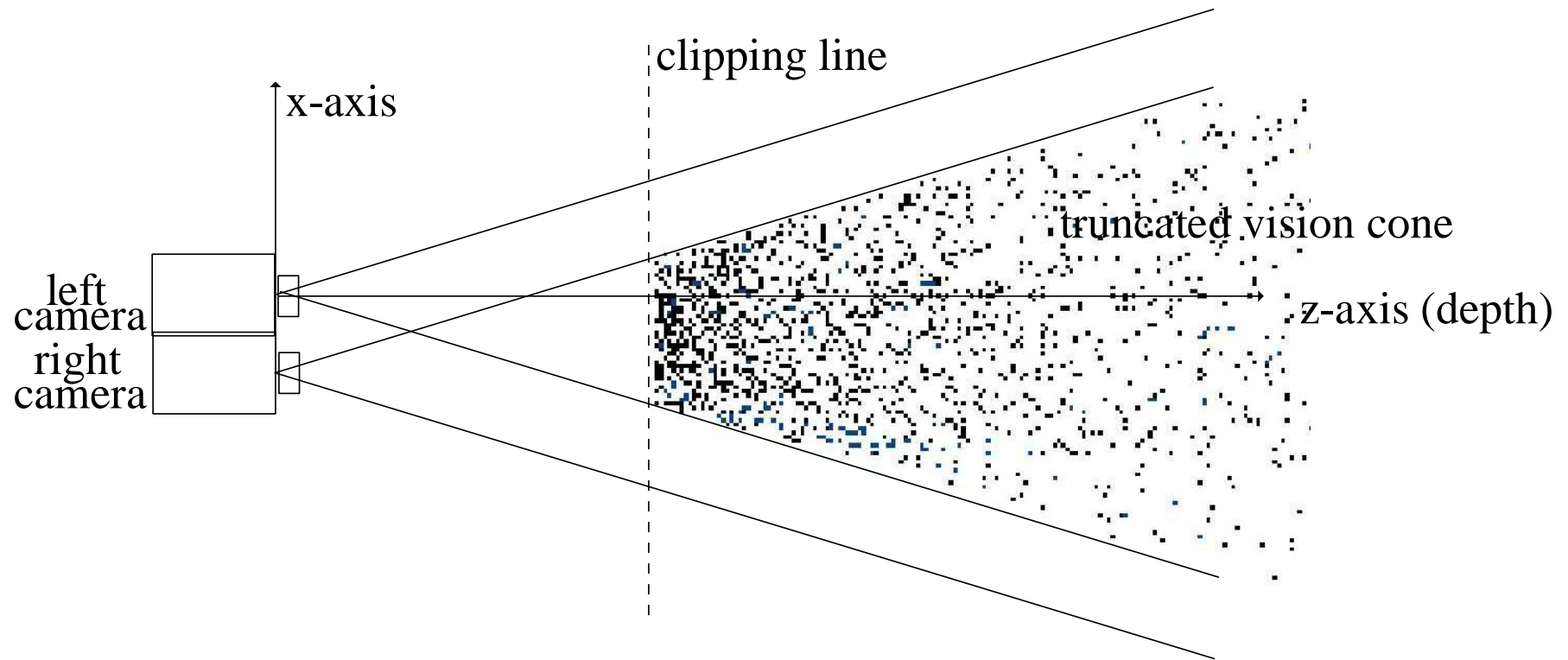
*Poursuite en temps réel d'une balle en 3-D (le diamètre apparent est inconnu), par transformation de Louchough.*

### 3 Evolution parisienne, ou comment diviser le travail. Exemple de l'algorithme des mouches.

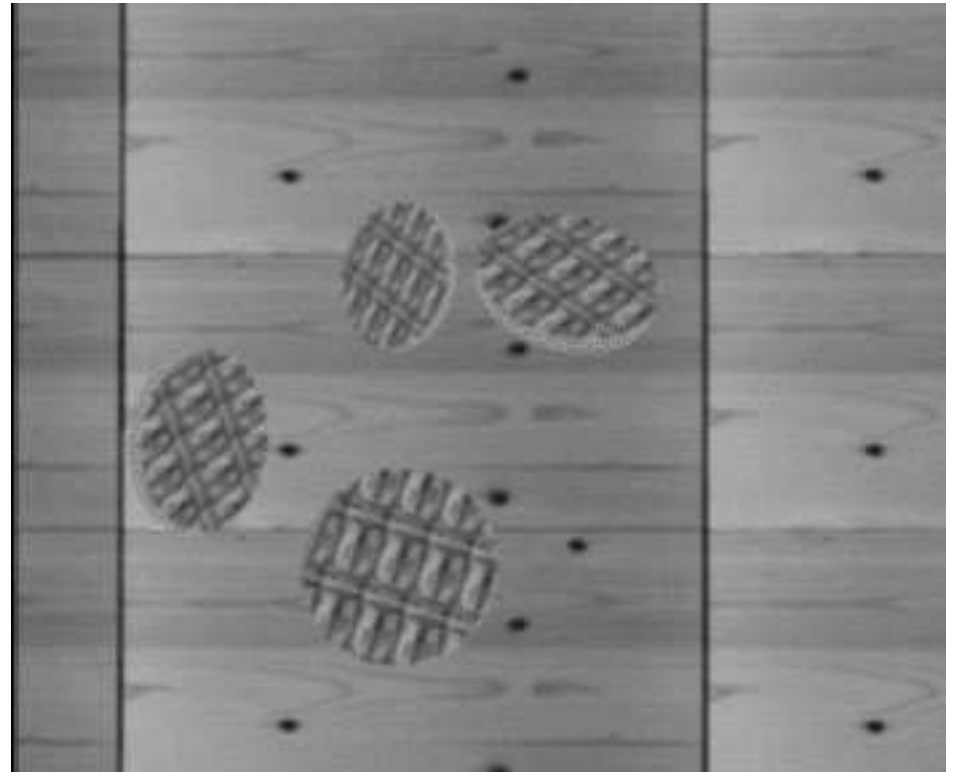
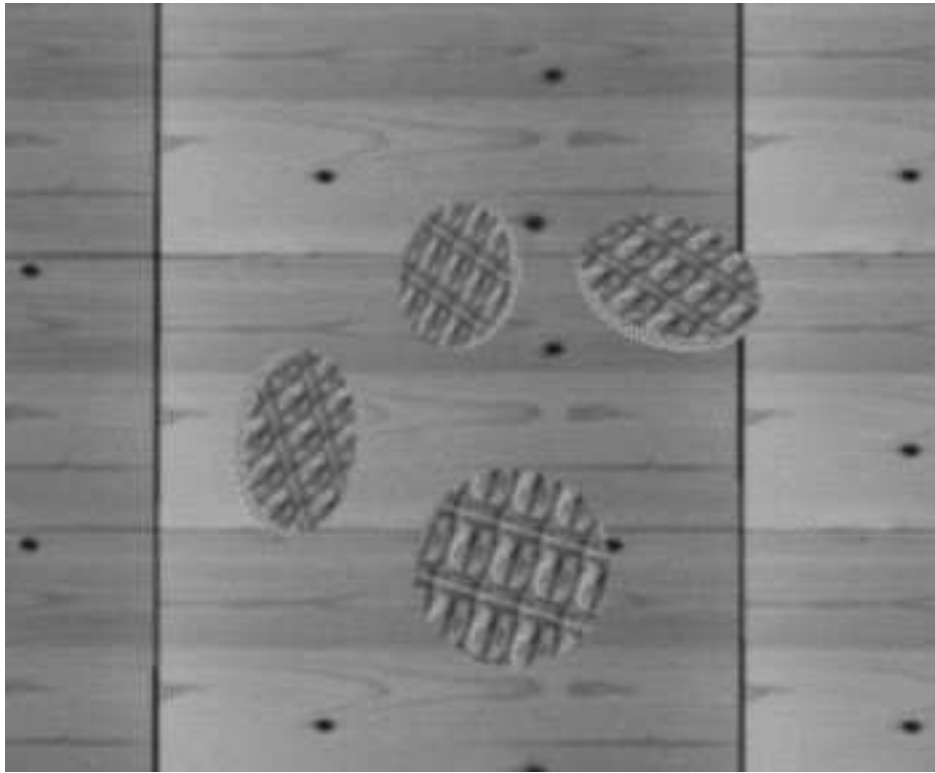


La fonction de fitness évalue le degré de cohérence de la mouche (- l'hypothèse) avec les données d'images issues des capteurs (ici, l'identité de leurs projections sur les deux caméras).

$$G = \sqrt{\sum_{(i,j) \in N} (L(x_L + i, y_L + j) - L(x_L, y_L))^2}$$

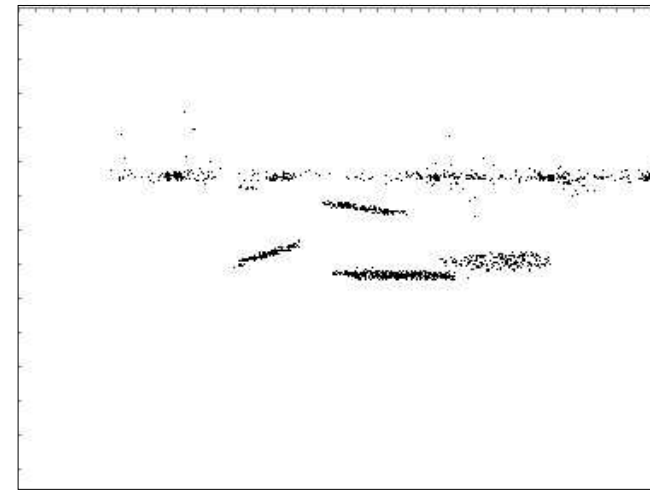
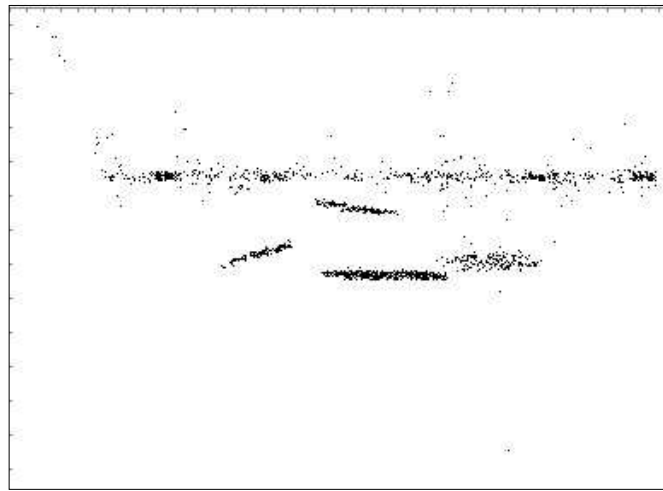
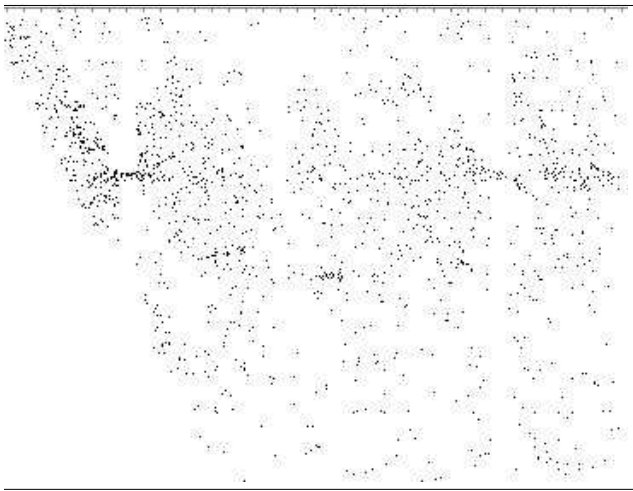


*La population initiale.*



*Une paire stereo (synthétique) qui va servir de test*





*Convergence après 10 , 50 et 100 générations  
(vue de dessus).*

# Une application en robotique mobile



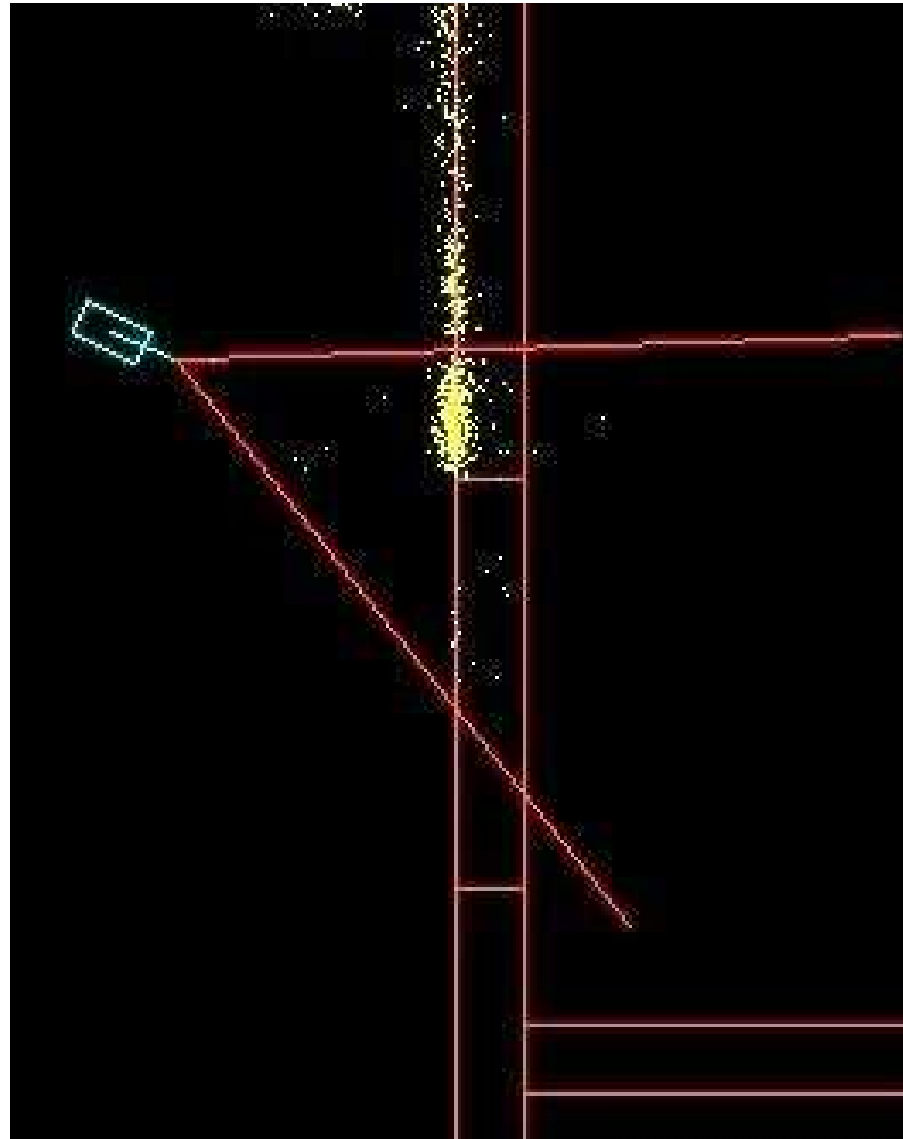
*Mouches calculées en temps réel dans un véhicule sur voie rapide*



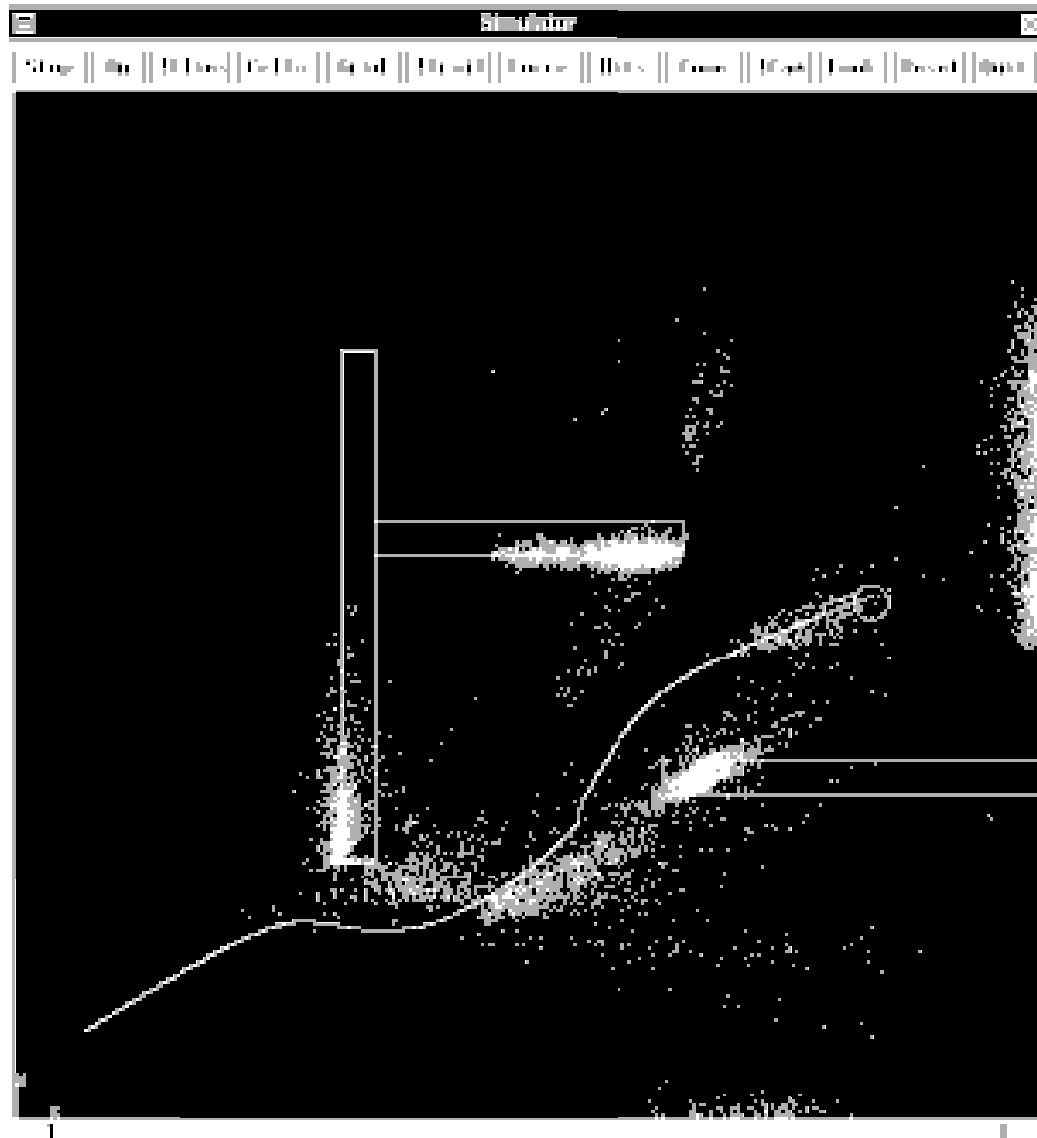
*Détection d'un piéton.*



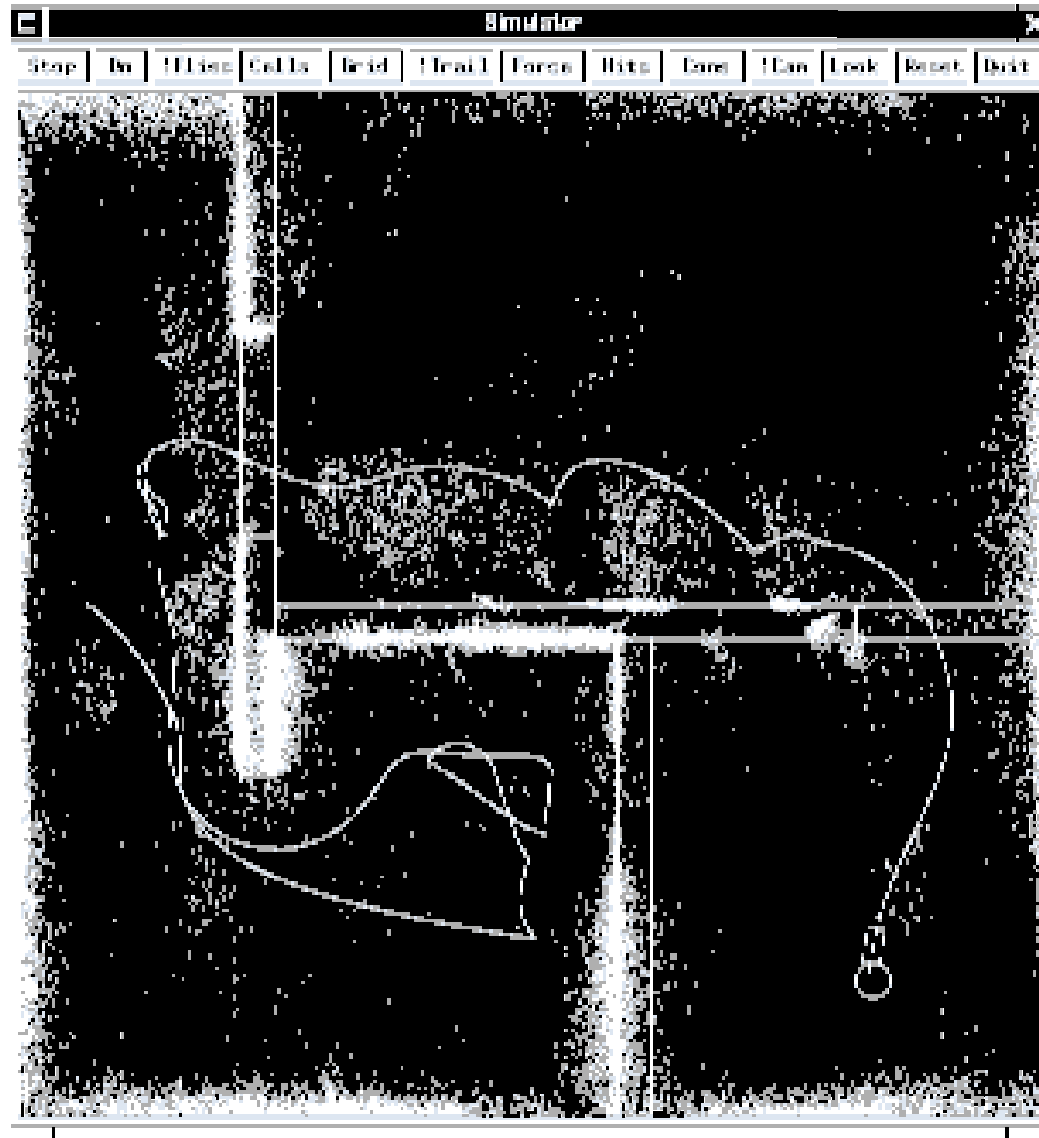
*Valeurs d'alerte (piéton)*



*Un robot mobile face à un mur et à une porte (dans un univers simulé). Les points brillants représentent les mouches mémorisées depuis les images précédentes.*

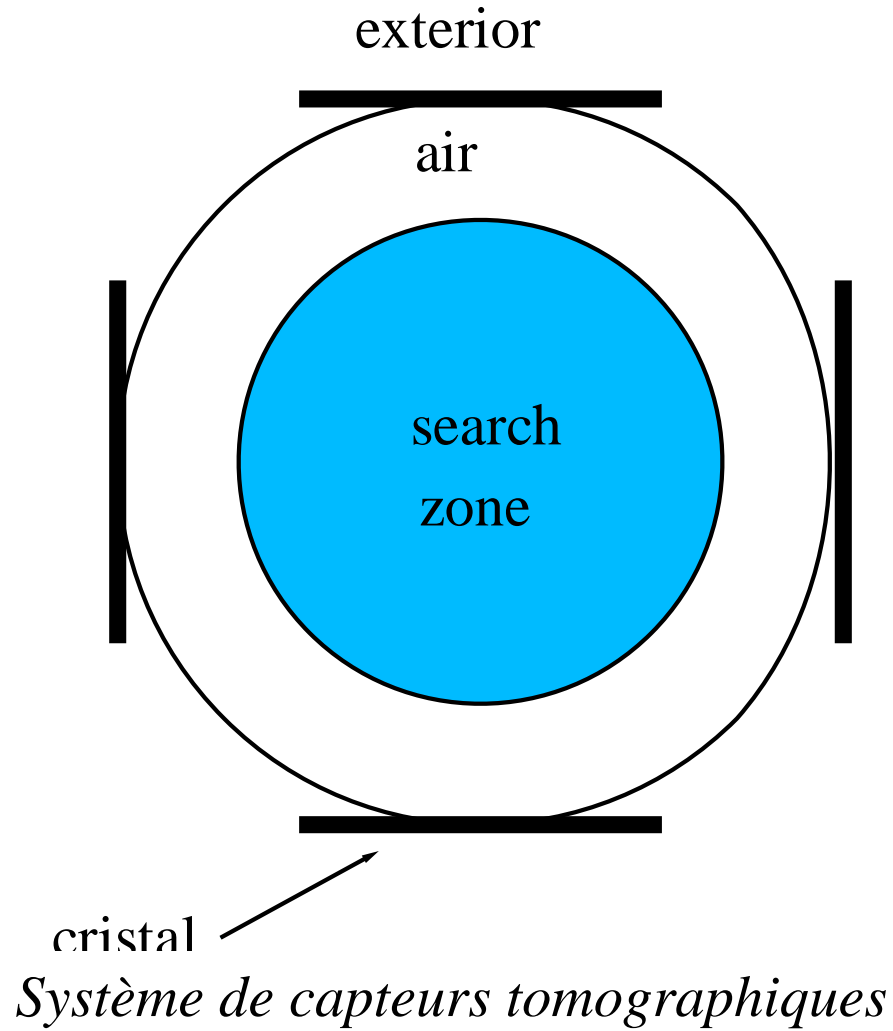


*Une trajectoire directe, sans situation de blocage (la cible est le cercle à droite).*



*Résolution d'un blocage difficile par création de cibles secondaires.*

# Une application en imagerie médicale





## Fitness par bonus

chaque mouche est un émetteur de photons

chaque photon qui atteint un pixel brillant de l'image réelle augmente le bonus de la mouche qui l'a créé.

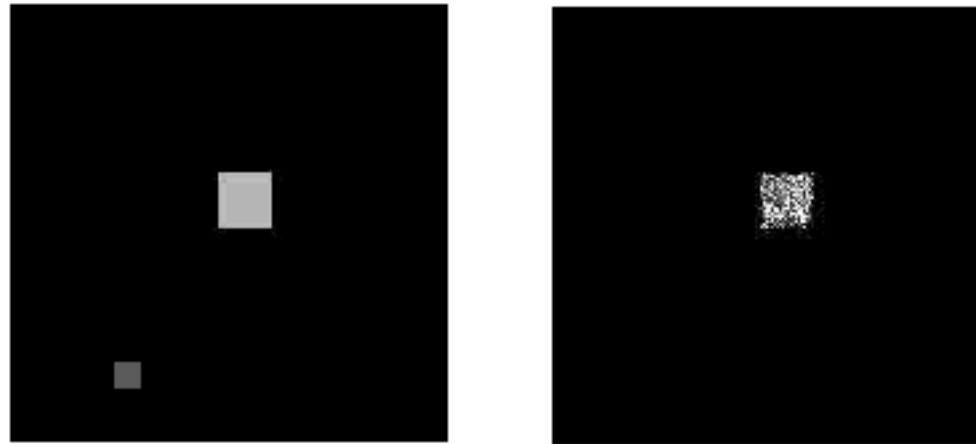


Figure 6: Bonus fitness: loss of smaller objects (left: ideal image; right: actual reconstruction), side view.

## Fitness marginale

Plutôt que d'évaluer chaque mouche indépendamment de son contexte (comme ci-dessus), on définit la fitness d'une mouche comme sa contribution (positive ou négative) à la fitness de la population totale:

$$fitness(i) = Fitness(population - \{i\}) - Fitness(population)$$



Figure 7: Marginal fitness: better detection of smaller objects (left: ideal image; right: actual reconstruction), side view.

## Comparaison de la fitness par bonus et de la fitness marginale

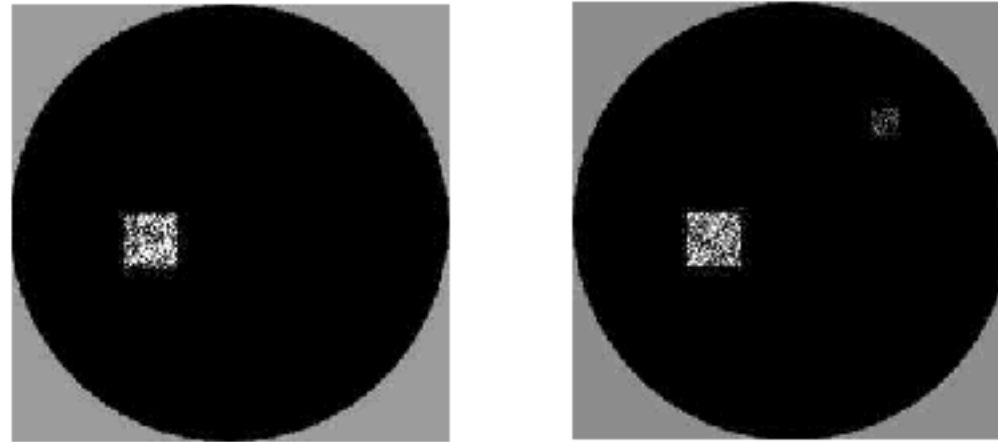
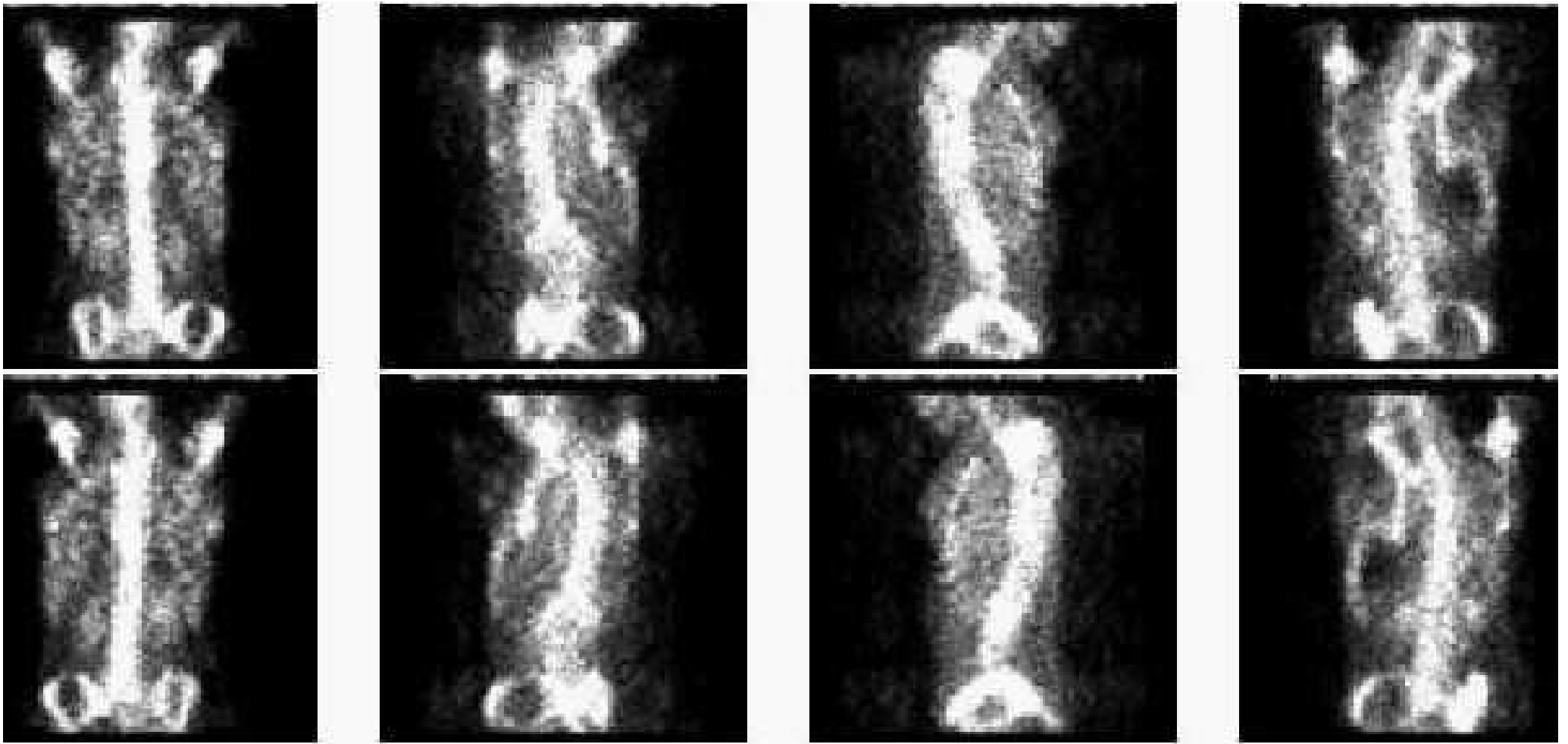
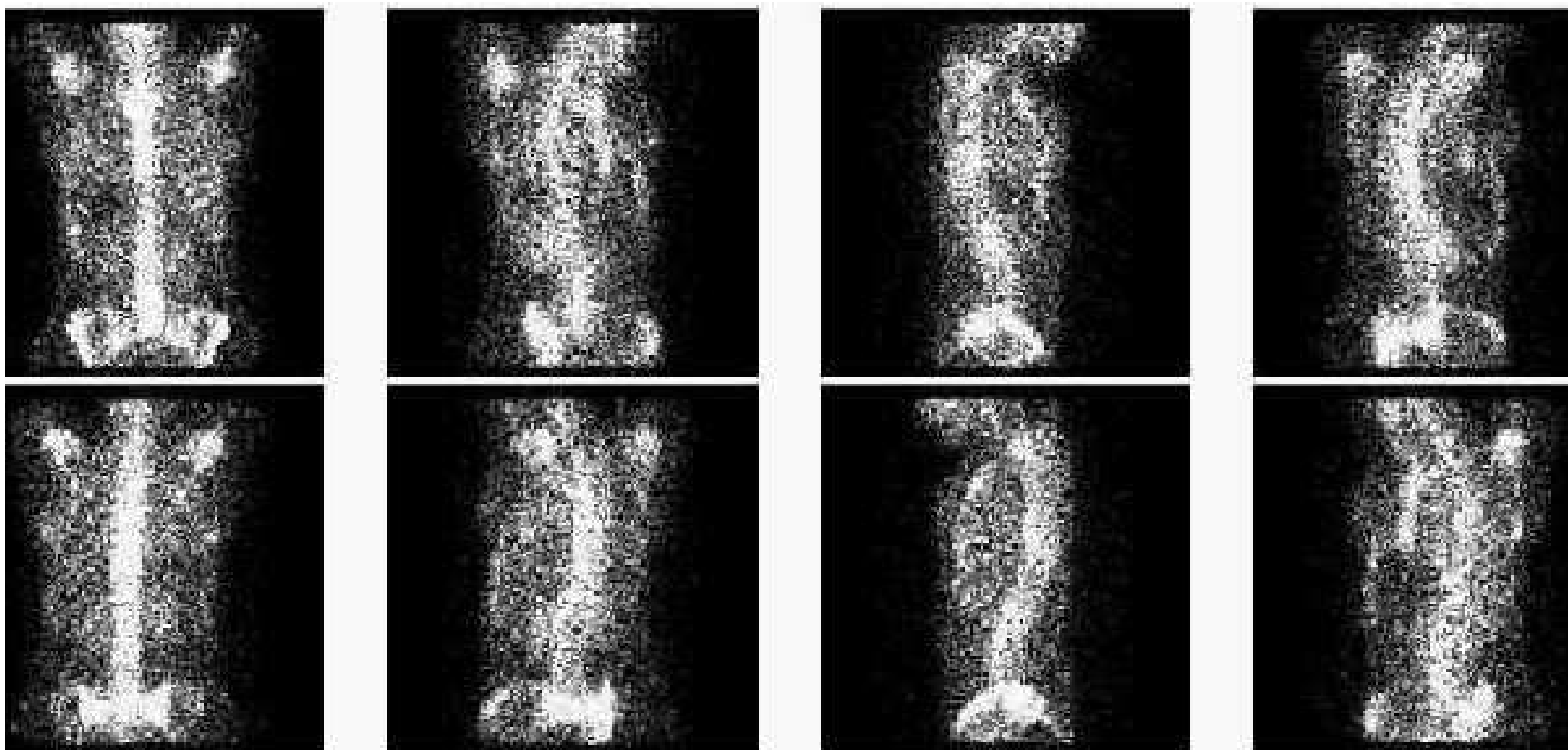


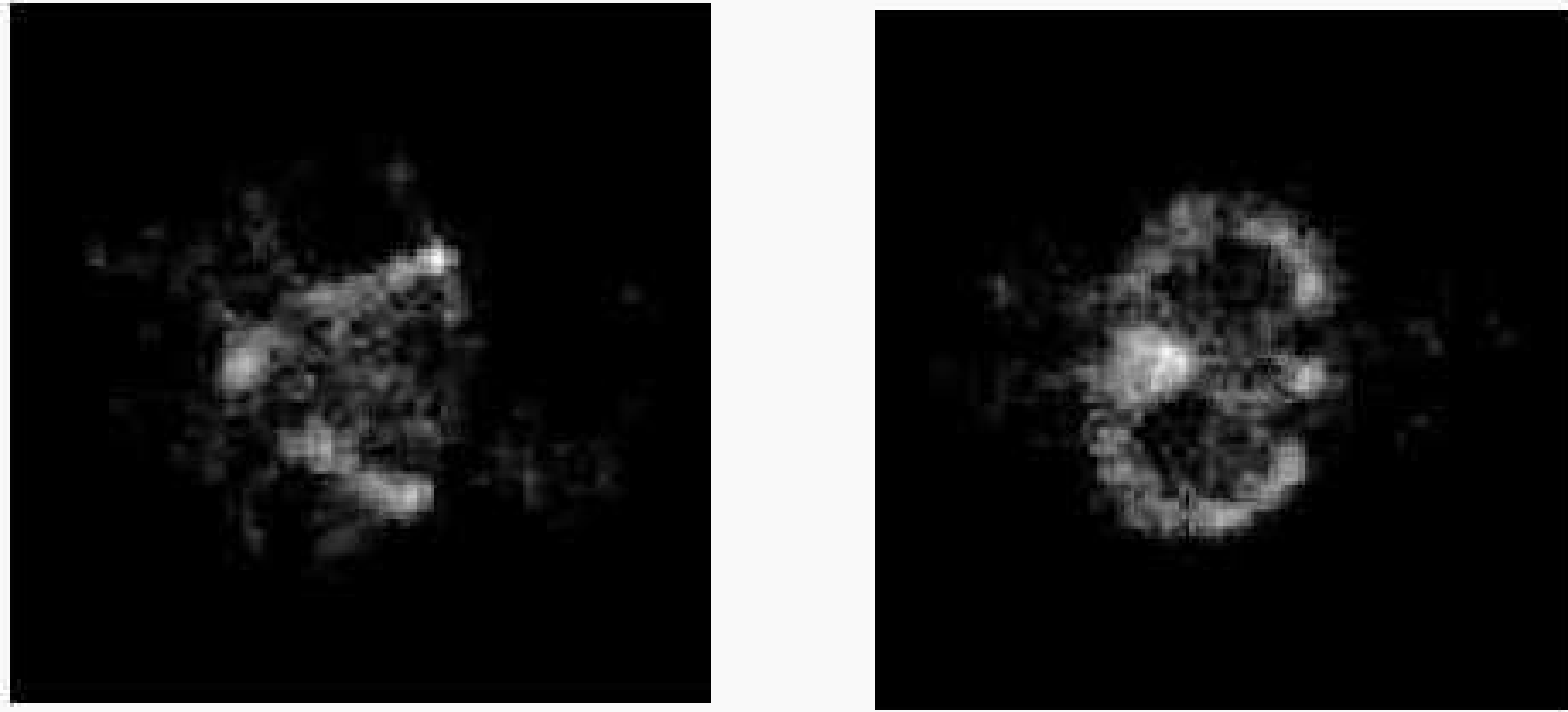
Figure 8: Top view of results with bonus (left) and marginal (right) fitness functions.



*8 images originales (projections sur 8 écrans détecteurs de photons)*



*3-D reconstruction 3-D par les mouches: le squelette est ici reconstruit à partir de seulement 2 vues, et voici les 8 projections de cette reconstruction 3-D. Ici on n'a pas tenu compte de l'effet Compton.*



*Coupes du pelvis, obtenues à partir de la même reconstruction 3-D faite à partir de 2 vues latérales (vue de dessus).*

## 4 Conclusion

- Optimiser de l'intérieur les paramètres d'un modèle pour ajuster aux résultats expérimentaux.
- les stratégies d'évolution (ES) permettent d'explorer des espaces de dimension élevée.
- décomposer les problèmes autant qu'il est nécessaire.
- pas de "boite noire" (l'évolution artificielle n'est pas de la magie). Fuir les algorithmes canoniques.
- 200 pages de doc contre une demi-page de code?
- $(A \Rightarrow B) \Rightarrow (\sim A \Rightarrow \sim B)$ . Se méfier des "théorèmes".
- la qualité du codage conditionne la réussite. Le pb. de l'épistasie.
- attention à l'Administratum: trop de raffinement tue l'efficacité. Raisonner en temps de calcul ou en générations?
- critère d'arrêt?

## **5 Acknowledgements**

Amine Boumaza

Aurélie Bousquet

Lionel Castillon

Pierre Collet

Anders Eckman

Evelyne Lutton

Sébastien Masclet

Olivier Pauplin

Xavier Provot

Jean-Marie Rocchisani

Maria Rodriguez Lopez

Bogdan Stanciulescu