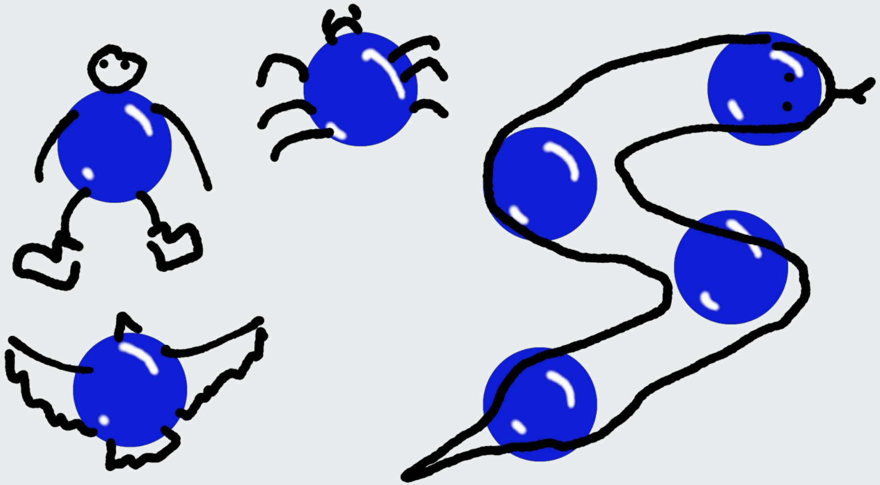
Birds, hikers, snakes and ants

Bio-inspired optimisation algorithms



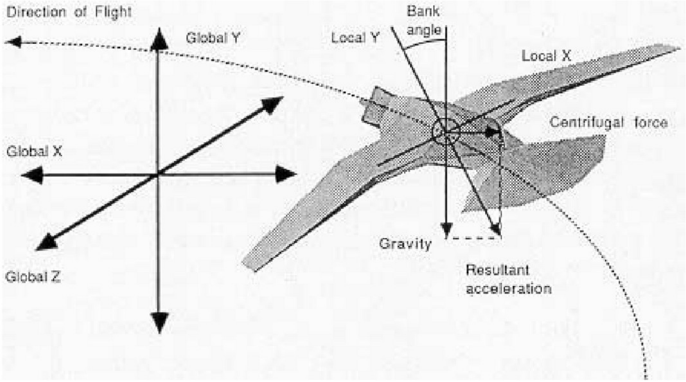
Animals are particles with simple behaviours

Reynolds introduced "Boids", probably the first biological use of particles

Boids are particles (3D points in R^3 space)

They follow a simple logic:

- Follow,
- Avoid,
- Group

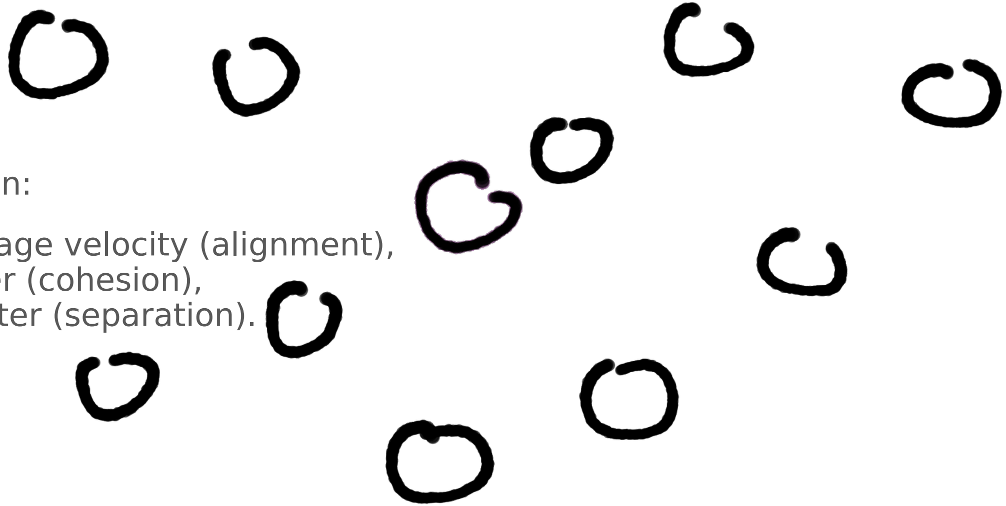


Reynolds, C. W. (1987). Flocks, Herds, and Schools: A Distributed Behavioral Model. In *Computer Graphics* (Vol. 21, Issue 4).
Reynolds, C. W. (2002). *Steering Behaviors For Autonomous Characters Steering Behaviors For Autonomous Characters*.

Boids

3 forces to change the particle’s acceleration:

- a force in direction of neighbors’ average velocity (alignment),
- a force in direction of neighbors center (cohesion),
- a force in opposition of neighbors center (separation).



And that’s all!

[Little demo]

Local behaviour = global pattern

Reynolds, C. W. (1987). Flocks, Herds, and Schools: A Distributed Behavioral Model. In *Computer Graphics* (Vol. 21, Issue 4).
Reynolds, C. W. (2002). *Steering Behaviors For Autonomous Characters Steering Behaviors For Autonomous Characters*.

Boids

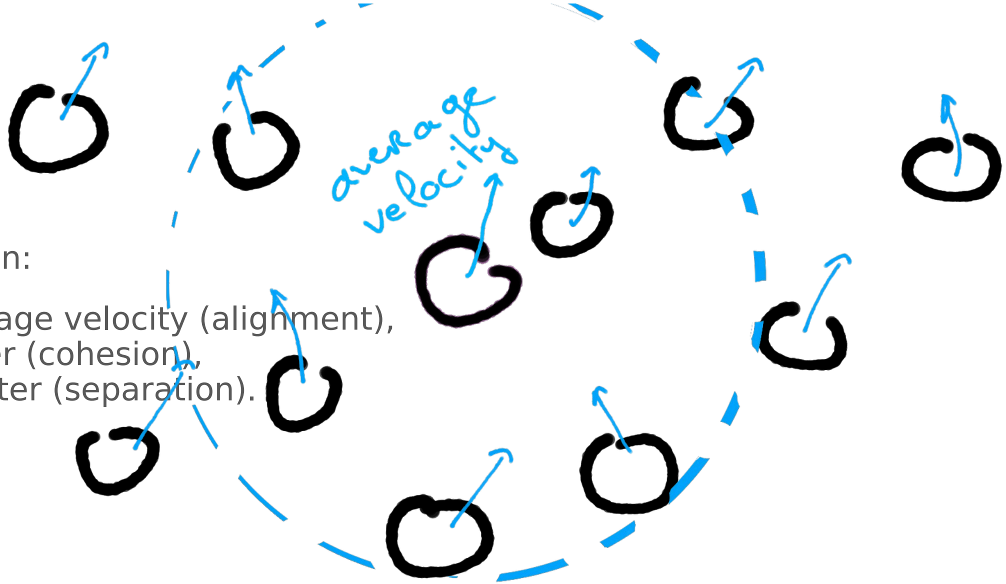
3 forces to change the particle's acceleration:

- a force in direction of neighbors' average velocity (alignment),
- a force in direction of neighbors center (cohesion),
- a force in opposition of neighbors center (separation).

And that's all!

[Little demo]

Local behaviour = global pattern



Reynolds, C. W. (1987). Flocks, Herds, and Schools: A Distributed Behavioral Model. In *Computer Graphics* (Vol. 21, Issue 4).
Reynolds, C. W. (2002). *Steering Behaviors For Autonomous Characters Steering Behaviors For Autonomous Characters*.

Boids

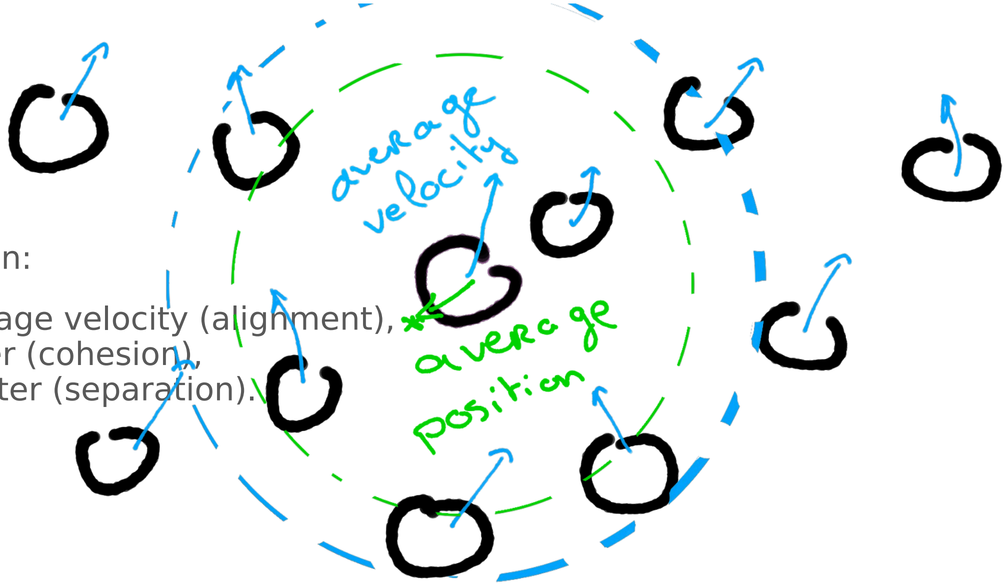
3 forces to change the particle's acceleration:

- a force in direction of neighbors' average velocity (alignment),
- a force in direction of neighbors center (cohesion),
- a force in opposition of neighbors center (separation).

And that's all!

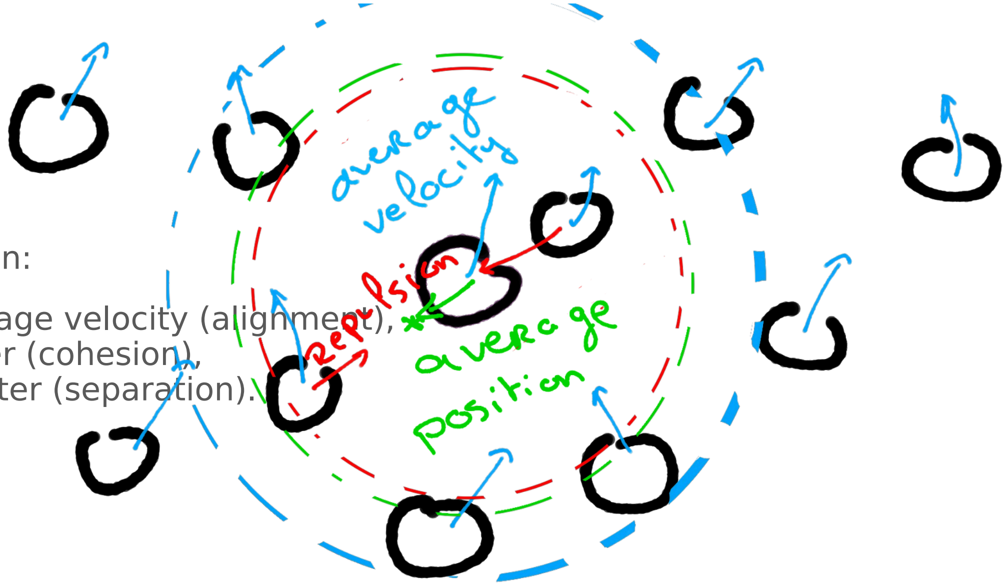
Little demo

Local behaviour = global pattern



Reynolds, C. W. (1987). Flocks, Herds, and Schools: A Distributed Behavioral Model. In *Computer Graphics* (Vol. 21, Issue 4).
Reynolds, C. W. (2002). *Steering Behaviors For Autonomous Characters Steering Behaviors For Autonomous Characters*.

Boids



3 forces to change the particle’s acceleration:

- a force in direction of neighbors’ average velocity (alignment),
- a force in direction of neighbors center (cohesion),
- a force in opposition of neighbors center (separation).

And that’s all!

[Little demo]

Local behaviour = global pattern

Reynolds, C. W. (1987). Flocks, Herds, and Schools: A Distributed Behavioral Model. In *Computer Graphics* (Vol. 21, Issue 4).
Reynolds, C. W. (2002). *Steering Behaviors For Autonomous Characters Steering Behaviors For Autonomous Characters*.

Boids

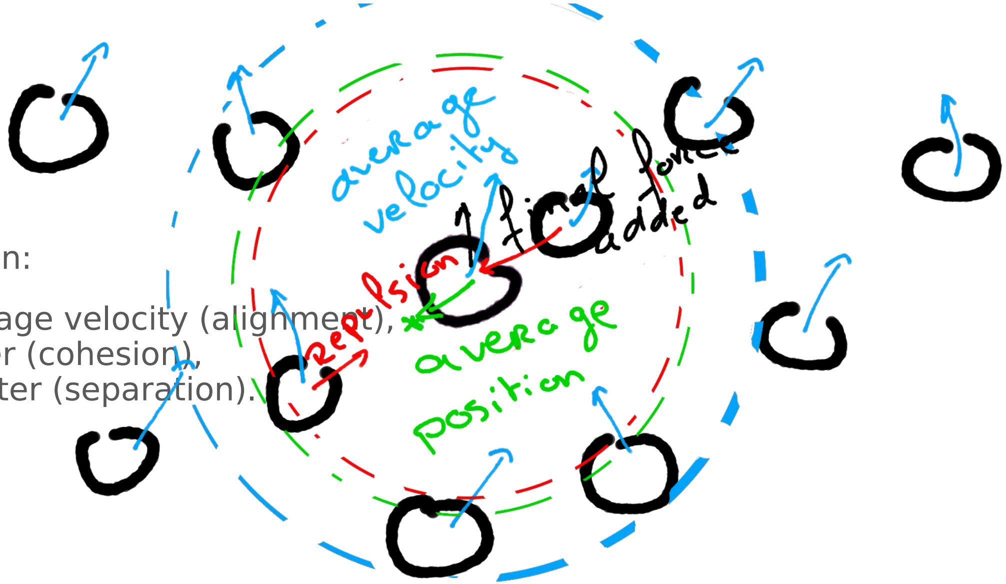
3 forces to change the particle's acceleration:

- a force in direction of neighbors' average velocity (alignment),
- a force in direction of neighbors center (cohesion),
- a force in opposition of neighbors center (separation).

And that's all!

[Little demo]

Local behaviour = global pattern



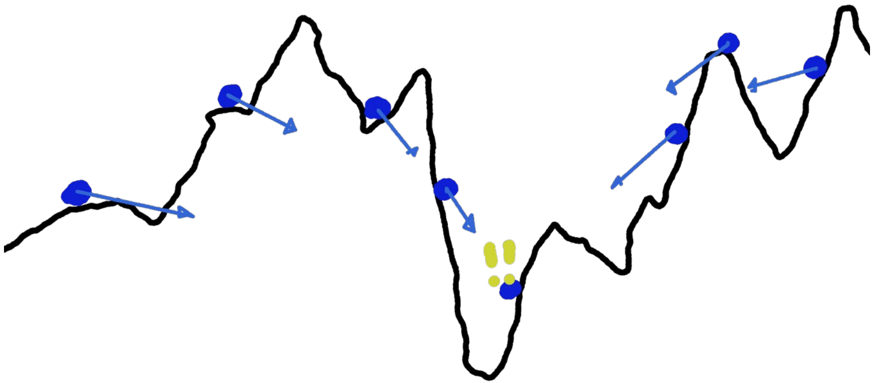
Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. *Proceedings of ICNN'95 - International Conference on Neural Networks*, *4*, 1942–1948.

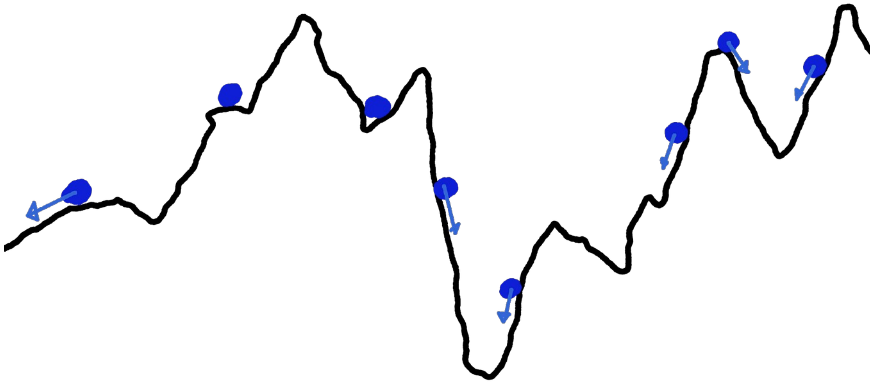
Social particles for optimisation

Particle Swarm Optimisation (PSO), a global minimum algorithm against Gradient Descent weaknesses.

No need for differentiation, no problem on non-differentiable function.

A single evaluation per step, even on N-dimensional spaces!





PSO idea

Gradient Descent idea

Algorithm:	Algorithm:
Each particle has a position	Each particle has a position
Evaluate the function	Evaluate the function and the gradient - You probably want to compute the finite difference method to compute the slope in each direction independently
Keep track of the best evaluation made so far	
Move a little bit toward the group’s best evaluation and your own best evaluation	Move in the direction of the gradient for an unknown distance - If the slope is null, you’re dead - If there is a discontinuity, you’re dead - If the slope is gentle, it’s going to be a long ride

Little demo

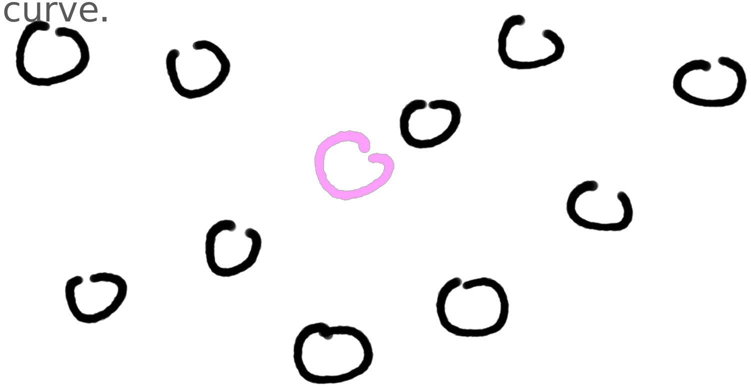
Little demo

Godoy, A., & von Zuben, F. J. (2009). A Complex Neighborhood based Particle Swarm Optimization. *2009 IEEE Congress on Evolutionary Computation*.
Cui, Z., & Shi, Z. (2009). Boid particle swarm optimisation. *International Journal of Innovative Computing and Applications*.

Introduce neighborhood

Increase searching area by having interconnected subgroups: “neighborhoods”

One of the simplest neighborhood: a simply connected curve.

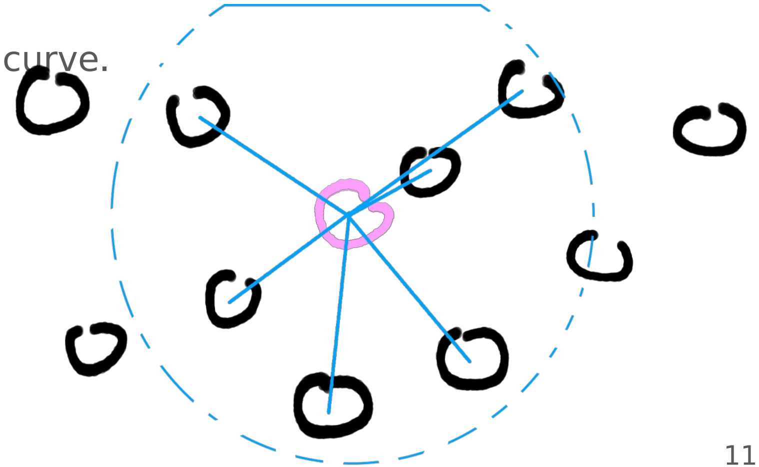


Godoy, A., & von Zuben, F. J. (2009). A Complex Neighborhood based Particle Swarm Optimization. *2009 IEEE Congress on Evolutionary Computation*.
Cui, Z., & Shi, Z. (2009). Boid particle swarm optimisation. *International Journal of Innovative Computing and Applications*.

Introduce neighborhood

Increase searching area by having interconnected subgroups: “neighborhoods”

One of the simplest neighborhood: a simply connected curve.

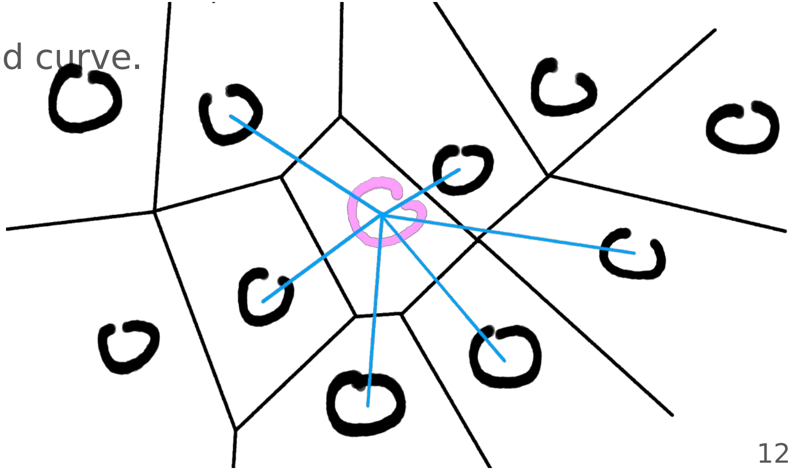


Godoy, A., & von Zuben, F. J. (2009). A Complex Neighborhood based Particle Swarm Optimization. *2009 IEEE Congress on Evolutionary Computation*.
Cui, Z., & Shi, Z. (2009). Boid particle swarm optimisation. *International Journal of Innovative Computing and Applications*.

Introduce neighborhood

Increase searching area by having interconnected subgroups: “neighborhoods”

One of the simplest neighborhood: a simply connected curve.

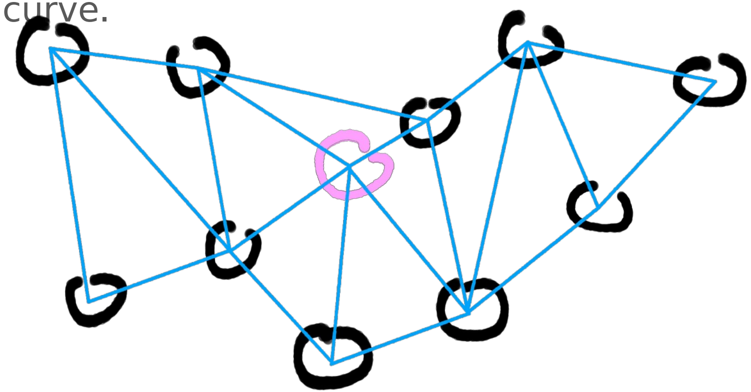


Godoy, A., & von Zuben, F. J. (2009). A Complex Neighborhood based Particle Swarm Optimization. *2009 IEEE Congress on Evolutionary Computation*.
Cui, Z., & Shi, Z. (2009). Boid particle swarm optimisation. *International Journal of Innovative Computing and Applications*.

Introduce neighborhood

Increase searching area by having interconnected subgroups: “neighborhoods”

One of the simplest neighborhood: a simply connected curve.

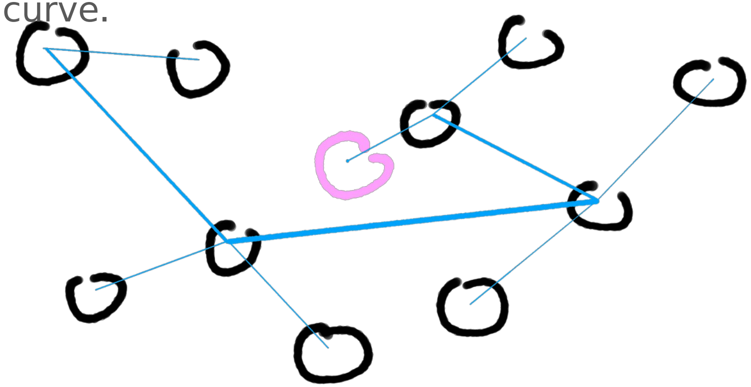


Godoy, A., & von Zuben, F. J. (2009). A Complex Neighborhood based Particle Swarm Optimization. *2009 IEEE Congress on Evolutionary Computation*.
Cui, Z., & Shi, Z. (2009). Boid particle swarm optimisation. *International Journal of Innovative Computing and Applications*.

Introduce neighborhood

Increase searching area by having interconnected subgroups: “neighborhoods”

One of the simplest neighborhood: a simply connected curve.

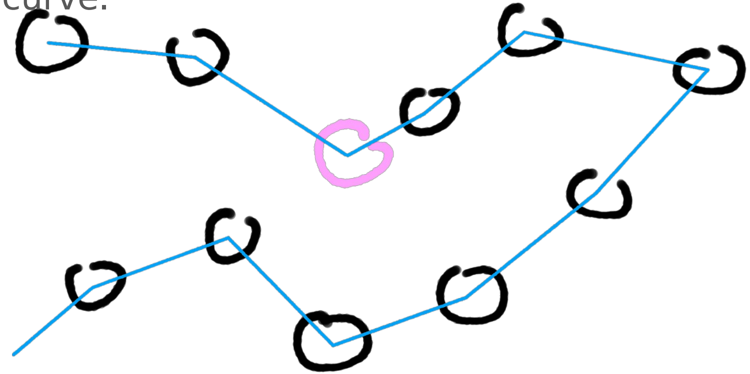


Godoy, A., & von Zuben, F. J. (2009). A Complex Neighborhood based Particle Swarm Optimization. *2009 IEEE Congress on Evolutionary Computation*.
Cui, Z., & Shi, Z. (2009). Boid particle swarm optimisation. *International Journal of Innovative Computing and Applications*.

Introduce neighborhood

Increase searching area by having interconnected subgroups: “neighborhoods”

One of the simplest neighborhood: a simply connected curve.



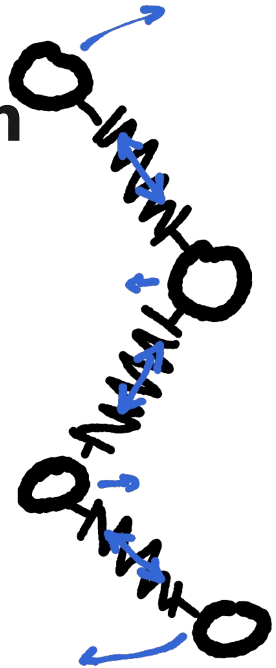
Local constraints instead of global constraints

We’ve seen that particles can locally create a global behaviour,

We’ve seen that particles can locally find optimal minimisations,

We’ll see that particles can do both at once.

Snake, the active contours algorithm



Algorithm:

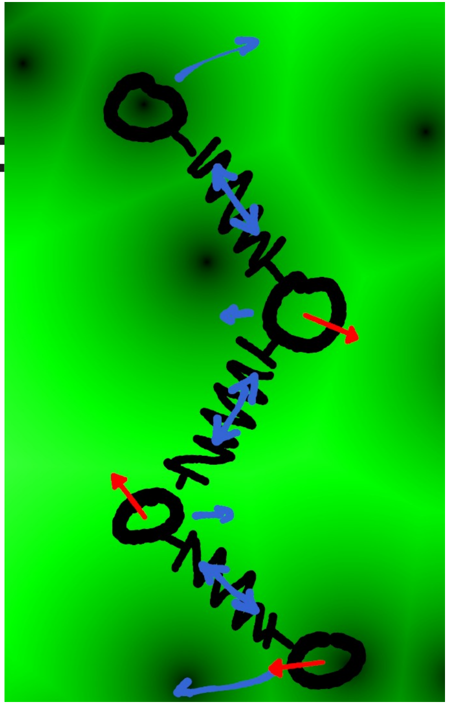
- Each particle is linked by springs with 2 neighbors
- Each particle follow the gradient of a function.

The end.

[Little demo]

Kass, M., Witkin, A., & Terzopoulos, D. (1988). Snakes: Active contour models. *International Journal of Computer Vision*, *1*(4), 321–331.

Snake, the active contours algorit



Algorithm:

- Each particle is linked by springs with 2 neighbors
- Each particle follow the gradient of a function.

The end.

[Little demo]

Kass, M., Witkin, A., & Terzopoulos, D. (1988). Snakes: Active contour models. *International Journal of Computer Vision*, *1*(4), 321–331.

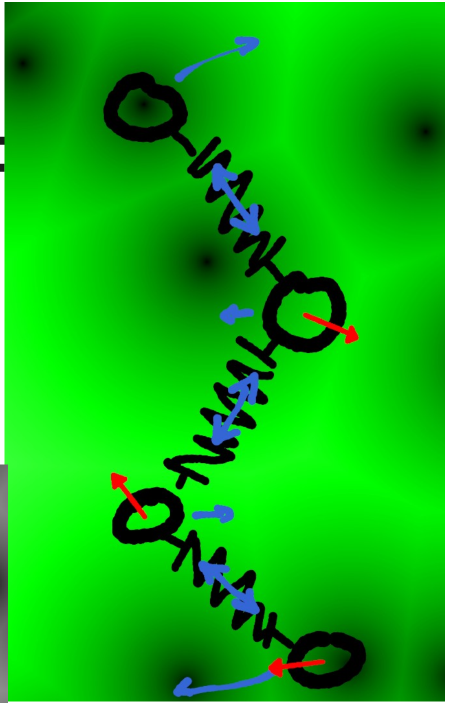
Snake, the active contours algorit

Algorithm:

- Each particle is linked by springs with 2 neighbors
- Each particle follow the gradient of a function.

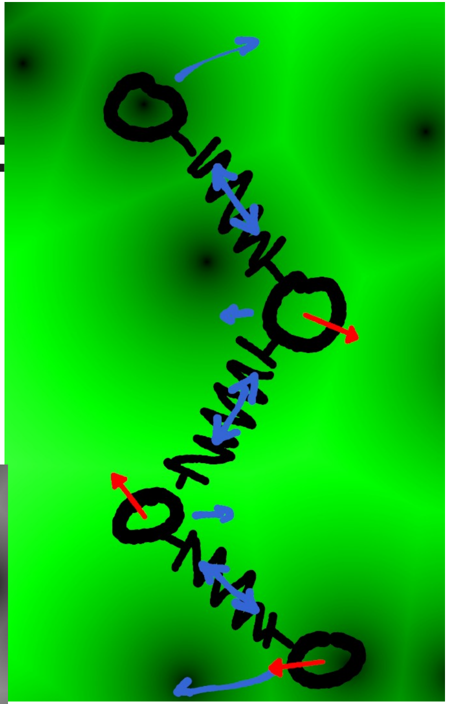
The end.

[Little demo]



Kass, M., Witkin, A., & Terzopoulos, D. (1988). Snakes: Active contour models. *International Journal of Computer Vision*, *1*(4), 321–331.

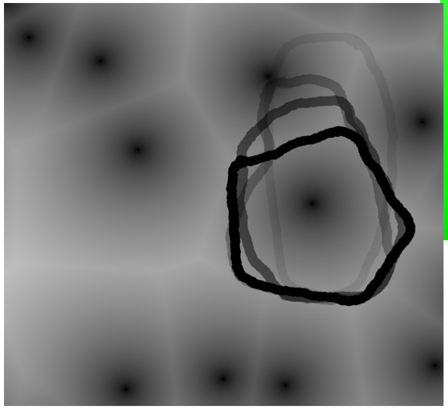
Snake, the active contours algorit



Algorithm:

- Each particle is linked by springs with 2 neighbors
- Each particle follow the gradient of a function.

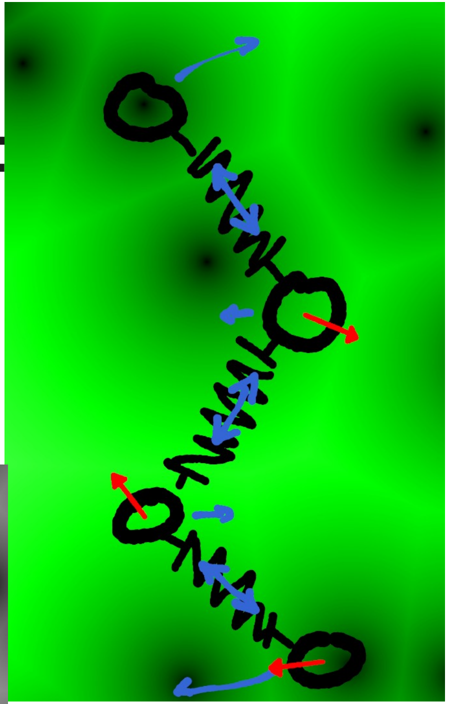
The end.



[Little demo]

Kass, M., Witkin, A., & Terzopoulos, D. (1988). Snakes: Active contour models. *International Journal of Computer Vision*, *1*(4), 321–331.

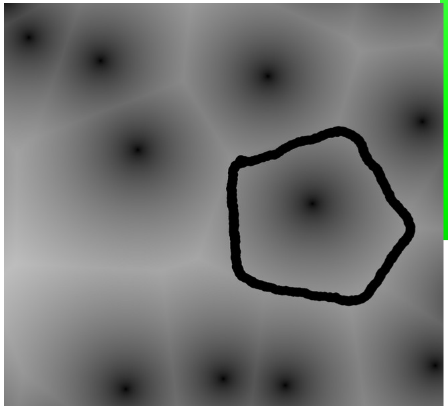
Snake, the active contours algorit



Algorithm:

- Each particle is linked by springs with 2 neighbors
- Each particle follow the gradient of a function.

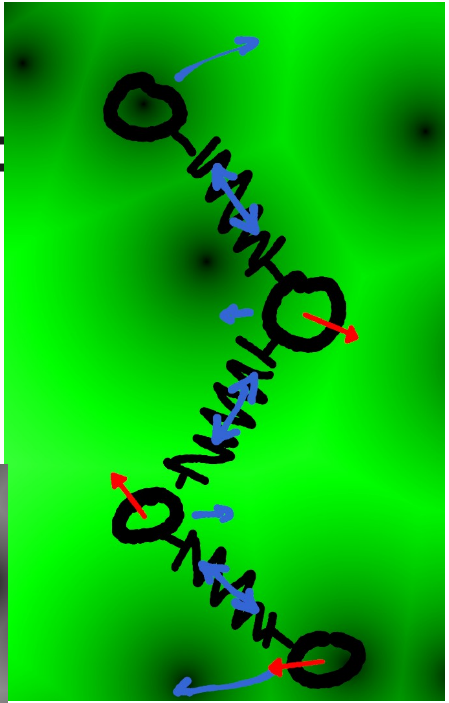
The end.



[Little demo]

Kass, M., Witkin, A., & Terzopoulos, D. (1988). Snakes: Active contour models. *International Journal of Computer Vision*, *1*(4), 321–331.

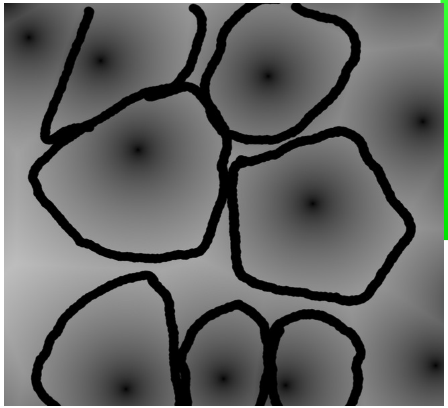
Snake, the active contours algorit



Algorithm:

- Each particle is linked by springs with 2 neighbors
- Each particle follow the gradient of a function.

The end.



[Little demo]

Snake, easy to compute

Minimisation of energy in a system

- Internal energy (controls locally the shape)
    - “Continuity”
    - “Curvature”
- External energy (controls the positioning of the shape, what we want to optimise)
    - “Image”

$$
\begin{aligned}
E^*_{\text{snake}} &= \int_0^1 E_{\text{snake}}(\mathbf{v}(s))\, ds \\
&= \int_0^1 E_{\text{int}}(\mathbf{v}(s)) + E_{\text{image}}(\mathbf{v}(s)) \\
&\quad + E_{\text{con}}(\mathbf{v}(s))\, ds
\end{aligned}
$$

Kass, M., Witkin, A., & Terzopoulos, D. (1988). Snakes: Active contour models. *International Journal of Computer Vision*, *1*(4), 321–331.

Snake, easy to compute… at first sight

Minimisation of energy in a system

- Internal energy (controls locally the shape)
    - “Continuity”
    - “Curvature”
- External energy (controls the positioning of the shape, what we want to optimise)
    - “Image”

$$
\begin{aligned}
E^*_{\text{snake}} &= \int_0^1 E_{\text{snake}}(\mathbf{v}(s))\,ds \\
&= \int_0^1 E_{\text{int}}(\mathbf{v}(s)) + E_{\text{image}}(\mathbf{v}(s)) \\
&\quad + E_{\text{con}}(\mathbf{v}(s))\,ds
\end{aligned}
$$

“Do whatever you want”

Kass, M., Witkin, A., & Terzopoulos, D. (1988). Snakes: Active contour models. *International Journal of Computer Vision*, *1*(4), 321–331.

Snake, easy to compute... at first sight

Minimisation of energy in a system

- Internal energy (controls locally the shape)
  - "Continuity"
  - "Curvature"
- External energy (controls the positioning of the shape, what we want to optimise)
  - "Image"

$$E_{\text{int}} = (\alpha(s)|\mathbf{v}_s(s)|^2 + \beta(s)|\mathbf{v}_{ss}(s)|^2)/2$$

$$E^*_{\text{snake}} = \int_0^1 E_{\text{snake}}(\mathbf{v}(s))\,ds$$

$$= \int_0^1 E_{\text{int}}(\mathbf{v}(s)) + E_{\text{image}}(\mathbf{v}(s))$$

$$+ E_{\text{con}}(\mathbf{v}(s))\,ds$$

"Do whatever you want"

Snake, easy to compute… at first sight

Minimisation of energy in a system

- Internal energy (controls locally the shape)
  - “Continuity”
  - “Curvature”
- External energy (controls the positioning of the shape, what we want to optimise)
  - “Image”

$$E_{\text{int}} = (\alpha(s)|\mathbf{v}_s(s)|^2 + \beta(s)|\mathbf{v}_{ss}(s)|^2)/2$$

$$E_{\text{image}} = w_{\text{line}}E_{\text{line}} + w_{\text{edge}}E_{\text{edge}} + w_{\text{term}}E_{\text{term}}$$

$$\begin{aligned} E^*_{\text{snake}} &= \int_0^1 E_{\text{snake}}(\mathbf{v}(s))\, ds \\ &= \int_0^1 E_{\text{int}}(\mathbf{v}(s)) + E_{\text{image}}(\mathbf{v}(s)) \\ &\quad + E_{\text{con}}(\mathbf{v}(s))\, ds \end{aligned}$$

“Do whatever you want”

Snake, easy to compute... at first sight

Minimisation of energy in a system

- Internal energy (controls locally the shape)
  - “Continuity”
  - “Curvature”
- External energy (controls the positioning of the shape, what we want to optimise)
  - “Image”

$$E_{\text{int}} = (\alpha(s)|\mathbf{v}_s(s)|^2 + \beta(s)|\mathbf{v}_{ss}(s)|^2)/2$$

$$E^*_{\text{snake}} = \int_0^1 E_{\text{snake}}(\mathbf{v}(s))\, ds$$

$$= \int_0^1 E_{\text{int}}(\mathbf{v}(s)) + E_{\text{image}}(\mathbf{v}(s))$$

$$+ E_{\text{con}}(\mathbf{v}(s))\, ds$$

$$E_{\text{image}} = w_{\text{line}}E_{\text{line}} + w_{\text{edge}}E_{\text{edge}} + w_{\text{term}}E_{\text{term}}$$

$$E_{\text{line}} = -(G_\sigma * \nabla^2 I)^2$$

“Do whatever you want”

Snake, easy to compute… at first sight

Minimisation of energy in a system

- Internal energy (controls locally the shape)
  - "Continuity"
  - "Curvature"
- External energy (controls the positioning of the shape, what we want to optimise)
  - "Image"

$$E_{\text{int}} = (\alpha(s)|\mathbf{v}_s(s)|^2 + \beta(s)|\mathbf{v}_{ss}(s)|^2)/2$$

$$E^*_{\text{snake}} = \int_0^1 E_{\text{snake}}(\mathbf{v}(s))\, ds$$

$$= \int_0^1 E_{\text{int}}(\mathbf{v}(s)) + E_{\text{image}}(\mathbf{v}(s))$$

$$+ E_{\text{con}}(\mathbf{v}(s))\, ds$$

$$E_{\text{image}} = w_{\text{line}} E_{\text{line}} + w_{\text{edge}} E_{\text{edge}} + w_{\text{term}} E_{\text{term}}$$

$$E_{\text{line}} = -(G_\sigma * \nabla^2 I)^2$$

$$E_{\text{edge}} = -|\nabla I(x, y)|^2$$

"Do whatever you want"

Kass, M., Witkin, A., & Terzopoulos, D. (1988). Snakes: Active contour models. *International Journal of Computer Vision*, *1*(4), 321–331.

Snake, easy to compute… at first sight

Minimisation of energy in a system

- Internal energy (controls locally the shape)
  - “Continuity”
  - “Curvature”
- External energy (controls the positioning of the shape, what we want to optimise)
  - “Image”

$$E_{\text{int}} = (\alpha(s)|\mathbf{v}_s(s)|^2 + \beta(s)|\mathbf{v}_{ss}(s)|^2)/2$$

$$\begin{aligned} E^*_{\text{snake}} &= \int_0^1 E_{\text{snake}}(\mathbf{v}(s))\,ds \\ &= \int_0^1 E_{\text{int}}(\mathbf{v}(s)) + E_{\text{image}}(\mathbf{v}(s)) \\ &\quad + E_{\text{con}}(\mathbf{v}(s))\,ds \end{aligned}$$

$$E_{\text{image}} = w_{\text{line}} E_{\text{line}} + w_{\text{edge}} E_{\text{edge}} + w_{\text{term}} E_{\text{term}}$$

$$E_{\text{line}} = -(G_\sigma * \nabla^2 I)^2$$

$$E_{\text{edge}} = -|\nabla I(x, y)|^2$$

$$\begin{aligned} E_{\text{term}} &= \frac{\partial\theta}{\partial \mathbf{n}_\perp} \\ &= \frac{\partial^2 C/\partial \mathbf{n}_\perp^2}{\partial C/\partial \mathbf{n}} \\ &= \frac{C_{yy}C_x^2 - 2C_{xy}C_xC_y + C_{xx}C_y^2}{(C_x^2 + C_y^2)^{3/2}} \end{aligned}$$

“Do whatever you want”

Snake, awful to tune

Perfect when the “object” is at the center of the space.

Perfect when there is a “global gradient” towards the object.

Perfect when the initial curve is already in a “good initial position”.

But what if…

- the gradient is null around a vertex,
- the slope is too gentle in the whole area,
- we don’t have the good initial conditions,
- …?

Snake, awful to tune

$$E_{\text{int}} = \left(\boxed{\alpha}(s)|\mathbf{v}_s(s)|^2 + \boxed{\beta}(s)|\mathbf{v}_{ss}(s)|^2\right)/2$$

$$E_{\text{image}} = \boxed{w_{\text{line}}}E_{\text{line}} + \boxed{w_{\text{edge}}}E_{\text{edge}} + \boxed{w_{\text{term}}}E_{\text{term}}$$

Perfect when the “object” is at the center of the space.

Perfect when there is a “global gradient” towards the object.

Perfect when the initial curve is already in a “good initial position”.

But what if…

- the gradient is null around a vertex,
- the slope is too gentle in the whole area,
- we don’t have the good initial conditions,
- …?

Rivest-Hénault, D., Cheriet, M., Deschênes, S., & Lapierre, C. (2010). Length increasing active contour for the segmentation of small blood vessels. *Proceedings - International Conference on Pattern Recognition*, 2796–2799.
Gunn, S. R., & Nixon, M. S. (1995). *Improving snake performance via a dual active contour* (pp. 600–605)

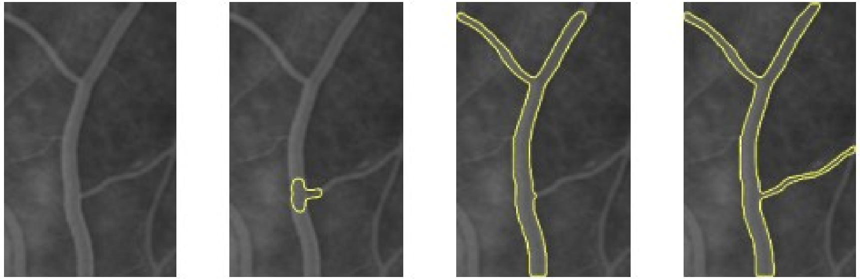
Snake, some improvements

- Null gradient around the shape:
    - Force the contours to grow

- We don’t know how to choose the initial conditions:
    - Use of dual-snakes

Rivest-Hénault, D., Cheriet, M., Deschênes, S., & Lapierre, C. (2010). Length increasing active contour for the segmentation of small blood vessels. *Proceedings - International Conference on Pattern Recognition*, 2796–2799.
Gunn, S. R., & Nixon, M. S. (1995). *Improving snake performance via a dual active contour* (pp. 600–605)

Snake, some improveme

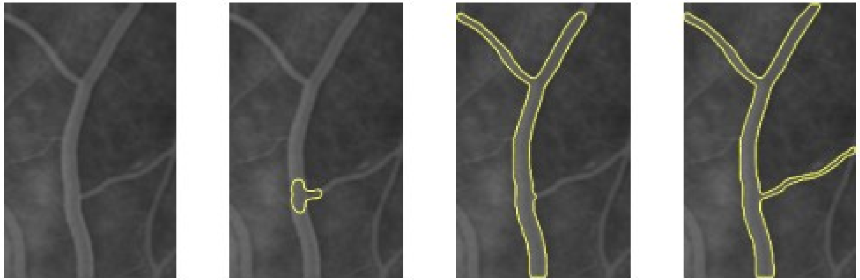

(a) (b) (c) (d)

- Null gradient around the shape:
  - Force the contours to grow

- We don't know how to choose the initial conditions:
  - Use of dual-snakes

Rivest-Hénault, D., Cheriet, M., Deschênes, S., & Lapierre, C. (2010). Length increasing active contour for the segmentation of small blood vessels. *Proceedings - International Conference on Pattern Recognition*, 2796–2799.
Gunn, S. R., & Nixon, M. S. (1995). *Improving snake performance via a dual active contour* (pp. 600–605)

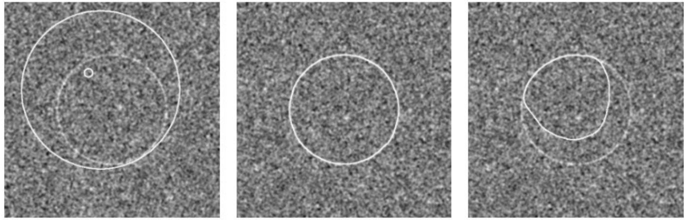
Snake, some improveme



(a) (b) (c) (d)

- Null gradient around the shape:
  - Force the contours to grow

- We don’t know how to choose the initial conditio
  - Use of dual-snakes

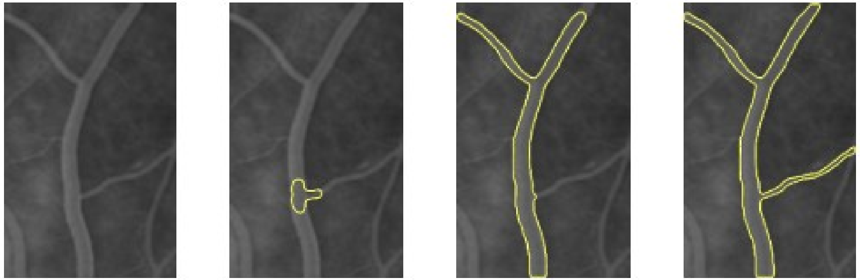


(a) Example Initialisation (b) Example Dual Result (c) Example Kass result

(d) Cup Initialisation (e) Dual Cup Result (f) Kass Cup Result

Rivest-Hénault, D., Cheriet, M., Deschênes, S., & Lapierre, C. (2010). Length increasing active contour for the segmentation of small blood vessels. *Proceedings - International Conference on Pattern Recognition*, 2796–2799.
Gunn, S. R., & Nixon, M. S. (1995). *Improving snake performance via a dual active contour* (pp. 600–605)

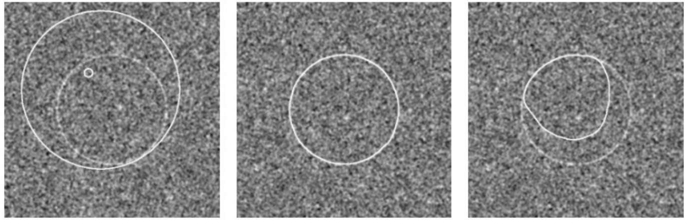
Snake, some improvem



(a) (b) (c) (d)

- Null gradient around the shape:
  - Force the contours to grow

- We don’t know how to choose the initial conditio
  - Use of dual-snakes



(a) Example Initialisation (b) Example Dual Result (c) Example Kass result

(d) Cup Initialisation (e) Dual Cup Result (f) Kass Cup Result

Snakes: **Active contour** models
M Kass, A Witkin, D Terzopoulos - International journal of computer vision, 1988 - Springer
… Figure 5 shows an example of such a **snake** exposed to a standard subjective **contour** illusion [7]. The shape of the **snake contour** between the edges and lines in the illusion is entirely …
☆ Enregistrer 99 Citer Cité 27441 fois Autres articles Les 41 versions

Snake, my Todo list

- Find a good initial curve placement using the Ant Colony Optimisation
- Mixing PSO and Snake for speeding up the convergence