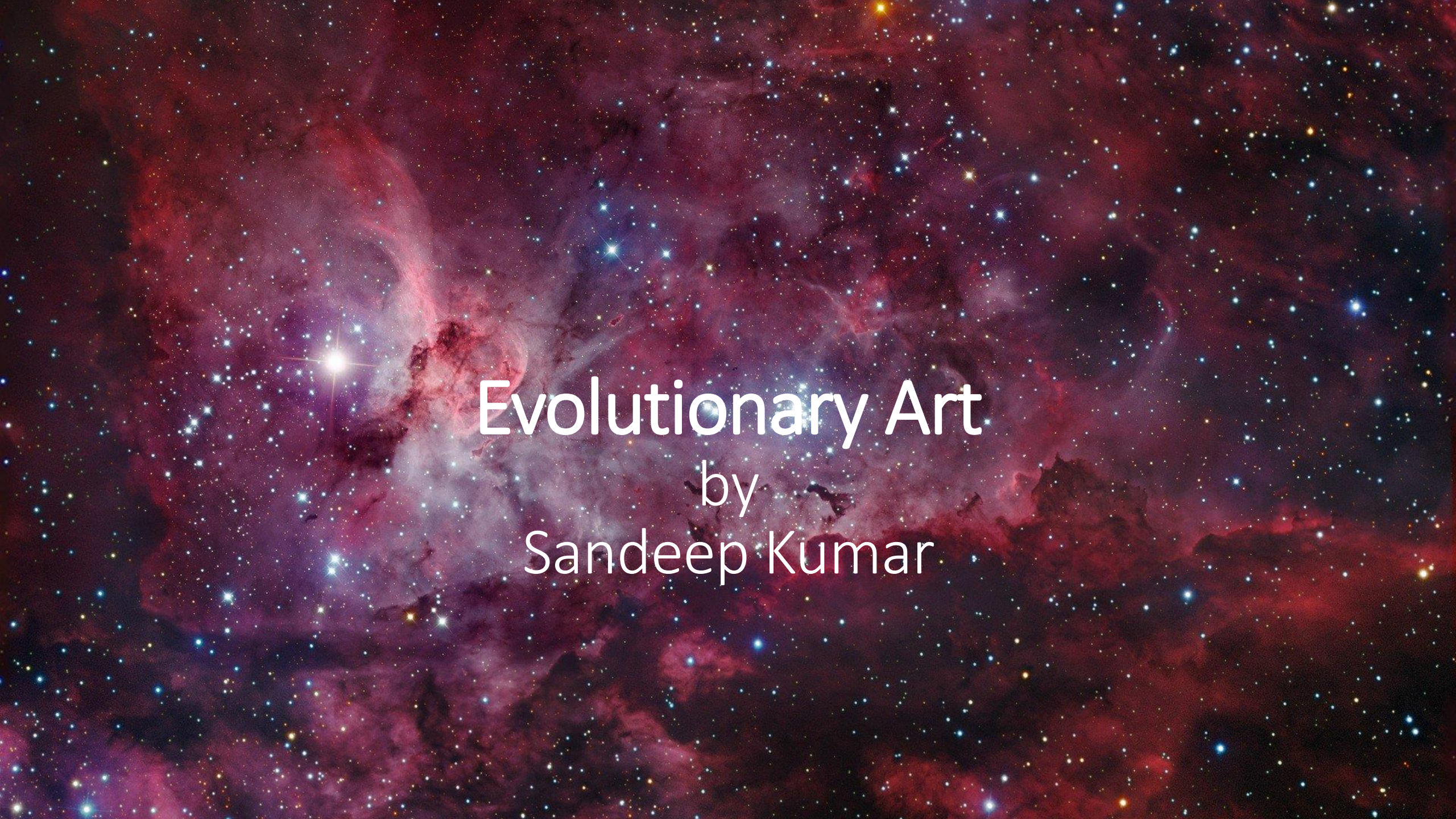
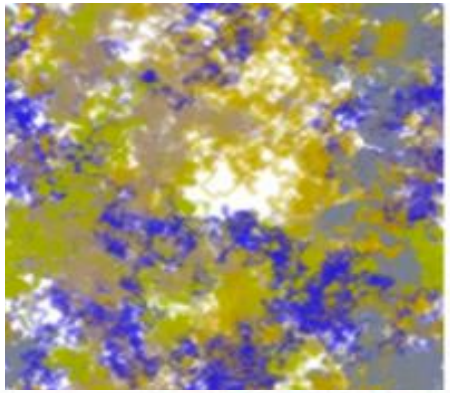
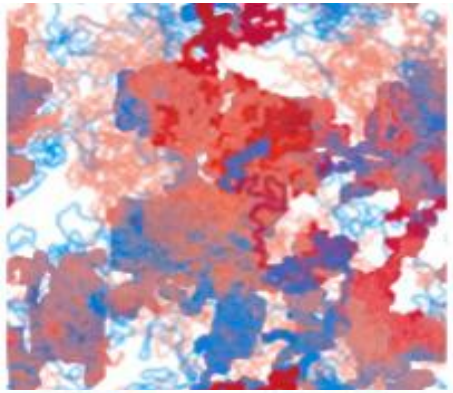

Evolutionary Art
by
Sandeep Kumar

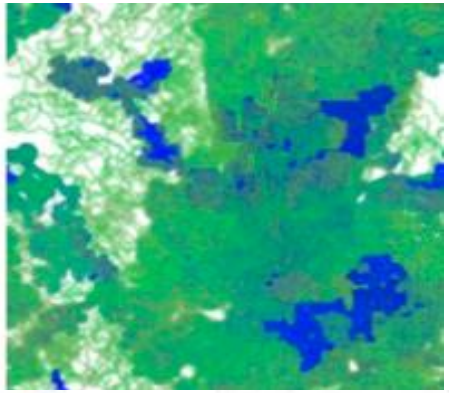
Evolutionary art categories

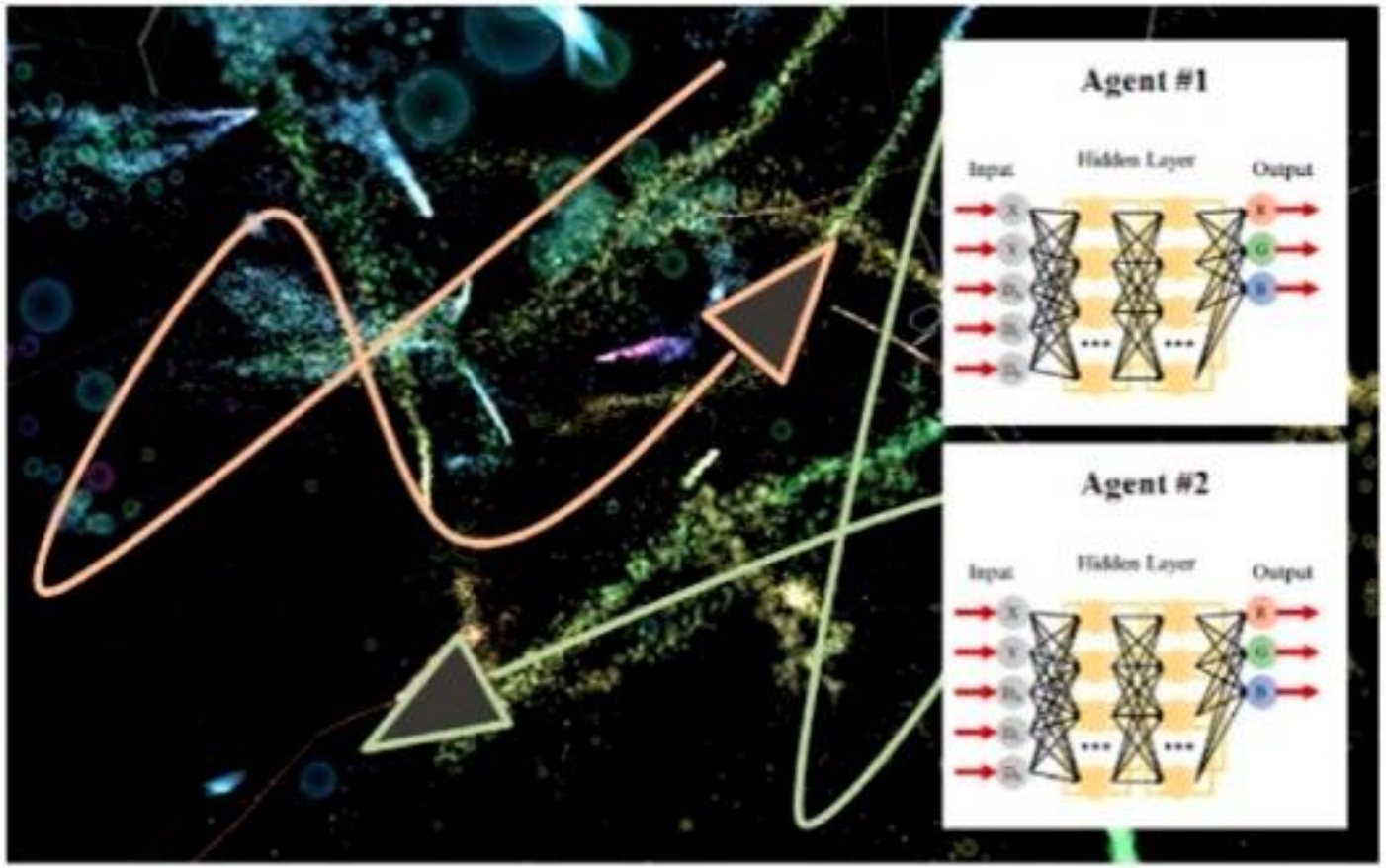
Classified into two categorical approaches:

- Agent-based
  - Genetic algorithm
  - Genetic programming
  - Ant colony optimization
  - Evolve evolutionary art with few agents (particles)
- Pixel- based Approach
  - Evolve with every single pixel.









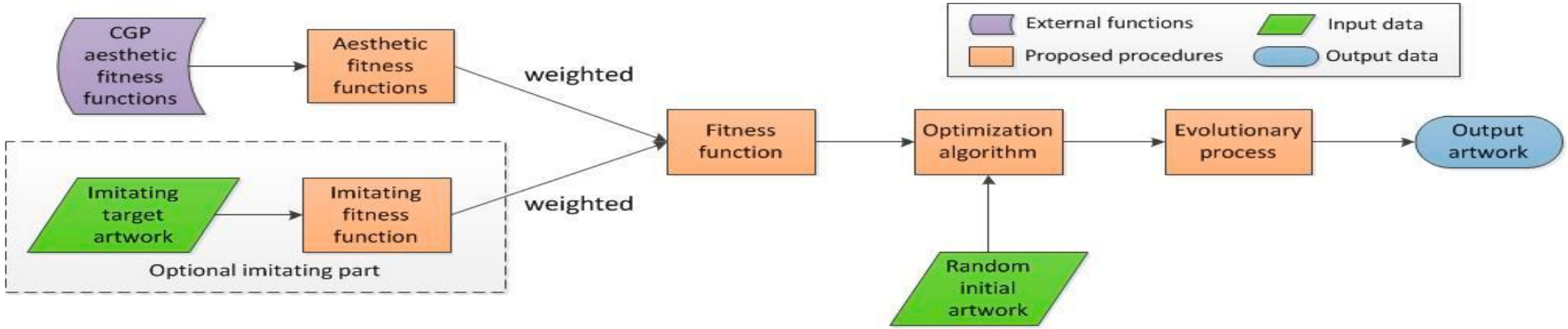
Agent-based approach:

Greenfield designed a series of ant painting methods, such as ant painting, using a multiple pheromone model. The ant colony mechanism was employed to generate evolutionary art.

Several ants moved on the canvas according to ant colony optimization rules and left colorful trajectories. Different fitness criteria were used to evaluate the aesthetic contributions

EVOLUTIONARY ARTWORKS GENERATED BY GREENFIELD'S ANT PAINTING METHOD

Pixel-Based Evolution Method - Flowchart

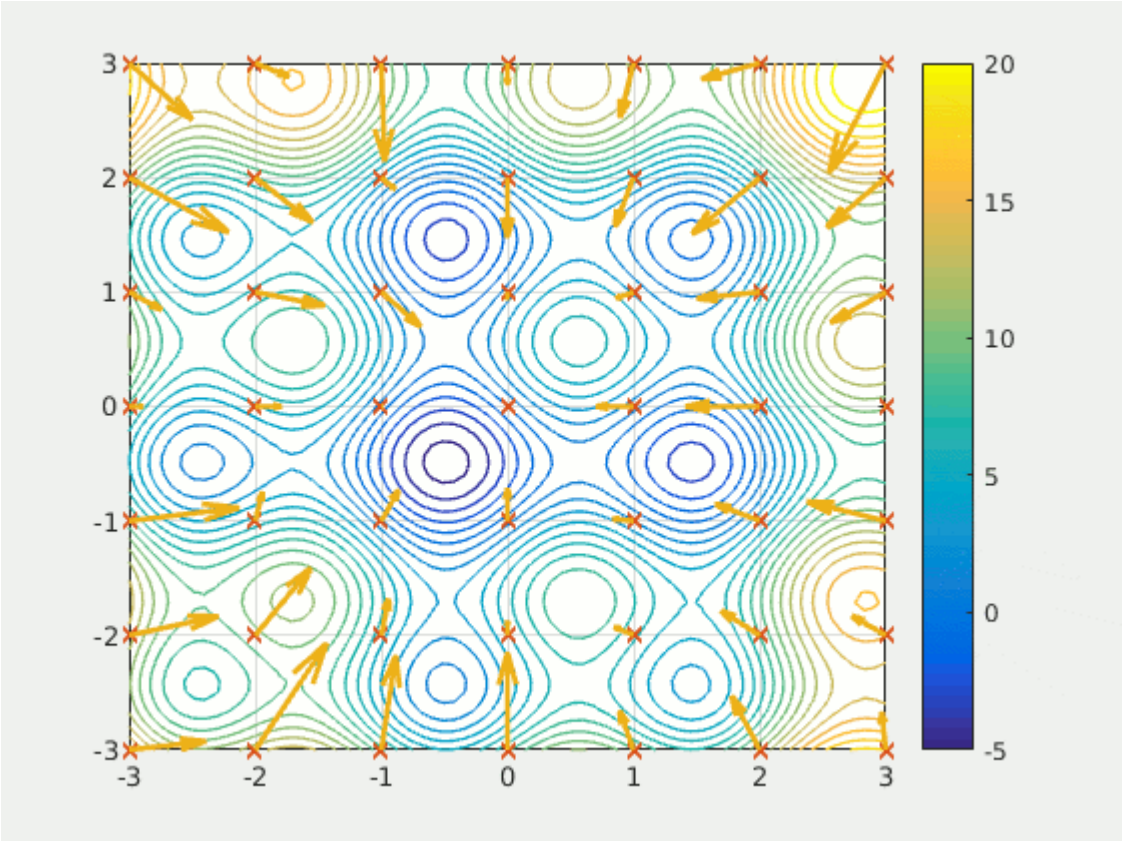


Particle Swarm Optimization

A computational method that optimizes a problem by iteratively trying to improve a candidate solution with regard to a given measure of quality. It solves a problem by having a population of candidate solutions, here dubbed particles, and moving these particles around in the search-space according to simple mathematical formula over the particle's position and velocity. Each particle's movement is influenced by its local best known position, but is also guided toward the best known positions in the search-space, which are updated as better positions are found by other particles. This is expected to move the swarm toward the best solutions.

Particle Swarm Optimization Algorithm:

for each particle $i = 1, \ldots, S$ **do**
    Initialize the particle's position with a uniformly distributed random vector: $\mathbf{x}_i \sim U(\mathbf{b}_{lo}, \mathbf{b}_{up})$
    Initialize the particle's best known position to its initial position: $\mathbf{p}_i \leftarrow \mathbf{x}_i$
    if $f(\mathbf{p}_i) < f(\mathbf{g})$ **then**
        update the swarm's best known position: $\mathbf{g} \leftarrow \mathbf{p}_i$
    Initialize the particle's velocity: $\mathbf{v}_i \sim U(-|\mathbf{b}_{up}-\mathbf{b}_{lo}|, |\mathbf{b}_{up}-\mathbf{b}_{lo}|)$
while a termination criterion is not met **do**:
    for each particle $i = 1, \ldots, S$ **do**
        for each dimension $d = 1, \ldots, n$ **do**
            Pick random numbers: $r_p, r_g \sim U(0,1)$
            Update the particle's velocity: $\mathbf{v}_{i,d} \leftarrow w\ \mathbf{v}_{i,d} + \varphi_p\ r_p\ (\mathbf{p}_{i,d}-\mathbf{x}_{i,d}) + \varphi_g\ r_g\ (\mathbf{g}_d-\mathbf{x}_{i,d})$
        Update the particle's position: $\mathbf{x}_i \leftarrow \mathbf{x}_i + \mathbf{v}_i$
        if $f(\mathbf{x}_i) < f(\mathbf{p}_i)$ **then**
            Update the particle's best known position: $\mathbf{p}_i \leftarrow \mathbf{x}_i$
            if $f(\mathbf{p}_i) < f(\mathbf{g})$ **then**
                Update the swarm's best known position: $\mathbf{g} \leftarrow \mathbf{p}_i$



$$\vec{x} = (x_1, x_2, x_h, x_s, x_v) \tag{1}$$

$$\min \quad f_h(\vec{x}), f_s(\vec{x}), f_v(\vec{x}), f_t(\vec{x}) \tag{2}$$

$$D = \left\{ \vec{x} \in R^5 \middle| 0 \le x_1 < width,\ 0 \le x_2 < height,\ 0 \le x_h < 256,\ 0 \le x_s < 256,\ 0 \le x_v < 256 \right\} \tag{3}$$

The vector $\vec{x}$ contains a set of attributes for each pixel. x_1 and x_2 are constant variable for current pixel. They indicate the coordinates of current pixel and participate in the fitness functions. x_h, x_s, x_v are the hue, saturation and value (H, S, V) color channels of the current pixel.

The fitness functions $f_h(\vec{x})$, $f_s(\vec{x})$, $f_v(\vec{x})$ are the aesthetic fitness functions.

The range of the CGP aesthetic fitness functions is [0, 255]

Table 1. CGP aesthetic fitness functions.

F_1	$(x_1 \mid x_2)\%255$
F_2	$(p\&x_1)\%255$
F_3	$\left(\frac{x_1}{1+x_2+p}\right)\%255$
F_4	$(x_1 \cdot x_2)\%255$
F_5	$(x_1 + x_2)\%255$
F_6	$\lvert x_1 - x_2\rvert\%255$
F_7	$255 - x_1\%255$
F_8	$\lvert 255 \cdot \cos x_1\rvert$
F_9	$\lvert 255 \cdot \tan((x_1\%45)\cdot\pi/180)\rvert$
F_{10}	$\lvert 255\cdot(\tan x_1)\%255\rvert$
F_{11}	$\sqrt{{x_1}^2 + {x_2}^2}\%255$
F_{12}	$x_1\%(p+1) + 255 - p$
F_{13}	$(x_1 + x_2)/2\%255$
F_{14}	$255\cdot\frac{(x_1+1)}{(x_2+1)}$, $(x_1 < x_2)$ *or* $255\cdot\frac{(x_2+1)}{(x_1+1)}$, *else*
F_{15}	$\left\lvert \sqrt{{x_1}^2 - p^2 + {x_2}^2 - p^2}\%255\right\rvert$

$$f_h\left(\vec{x}\right) = \left|x_h - F(x_1, x_2)\right| \tag{4}$$

$$f_s\left(\vec{x}\right) = \left|x_s - F(x_1, x_2)\right| \tag{5}$$

$$f_v\left(\vec{x}\right) = \left|x_v - F(x_1, x_2)\right| \tag{6}$$

The imitating fitness function $f_t\left(\vec{x}\right)$ is optional. There should be an artwork as the target art. We assume that the resolution of the target art is $W * H$. In this case, the coordinate attributes of vector $\vec{x}$, x_1 and x_2 are restricted to $x_1 \in [0,\ W-1]$ and $x_2 \in [0,\ H-1]$. The HSV values of a pixel for target art are represented by h, s, and v. The imitating fitness function $f_t\left(\vec{x}\right)$ is described as Equation (7):

$$f_t\left(\vec{x}\right) = \sum_{j=0}^{W-1}\sum_{k=0}^{H-1} \frac{x_h - h_{ij} + x_s - s_{ij} + x_v - v_{ij}}{3 \cdot W \cdot H \cdot \left((x_1 - j)^2 + (x_2 - k)^2 + 1\right)} \tag{7}$$

With all of the fitness functions set, the final SOP fitness function for pixel-based evolutionary art is described as Equation (8):

$$min\ \ f\left(\vec{x}\right) = f_h\left(\vec{x}\right) + f_s\left(\vec{x}\right) + f_v\left(\vec{x}\right) + t f_t\left(\vec{x}\right)$$

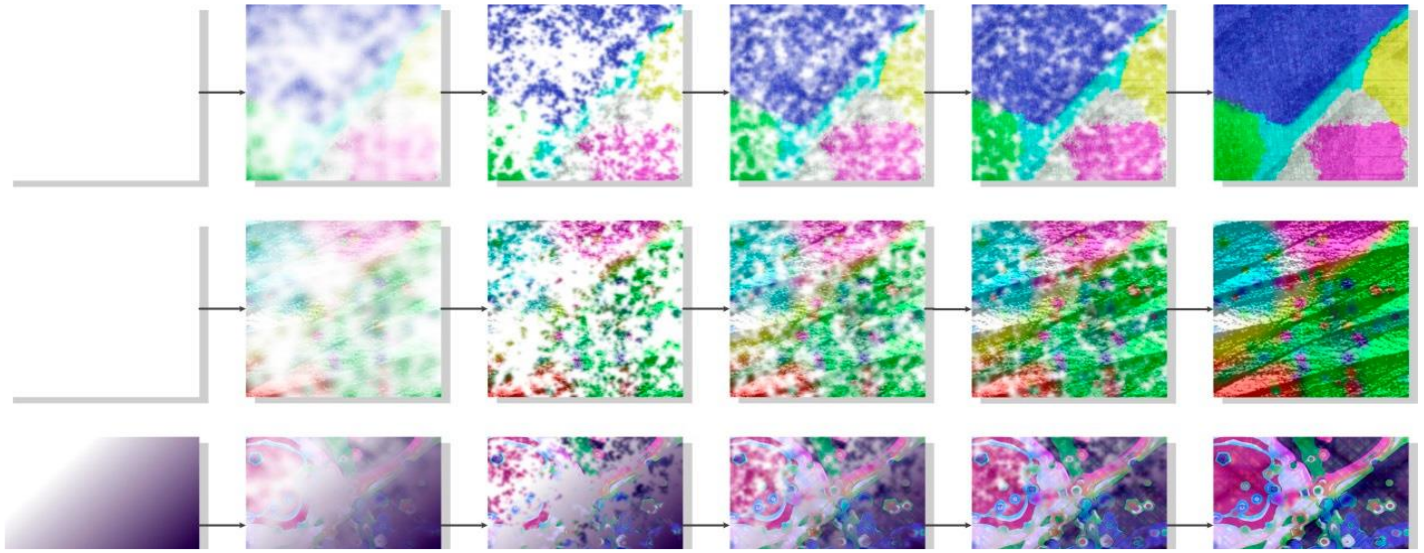
The pixel characteristic of a photograph is grid-organized. As the genetic algorithm has a lack of topology information, the PSO algorithm is a better choice. The original PSO algorithm has the global neighbor topology. However, it is unnecessary for pixel-based evolution and will result in the loss of local features. In the following part, we use PSO way to refer to things. The "pixels" will be called "particles"; The vector $\vec{x}$ will be called "location", since it is a set of values in the search space; The growth of the vector $\vec{x}$ will be called "velocity". We modified the global neighbor relationship into a von Neumann topology. The focal particle will only calculate the values of adjacent particles. The modified PSO algorithm is described as Equations (9) and (10):

$$\Delta \vec{x}_i^{k+1} = \omega \Delta \vec{x}_i^{k} + c_1 \cdot r_1 \cdot \left(\text{xbest}_i^k - \vec{x}_i^k \right) + c_2 \cdot r_2 \cdot \left(\text{vbest}_i^k - \vec{x}_i^k \right) \tag{9}$$

$$\vec{x}_i^{k+1} = \vec{x}_i^{k} + \cdot \vec{x}_i^{k+1} \tag{10}$$

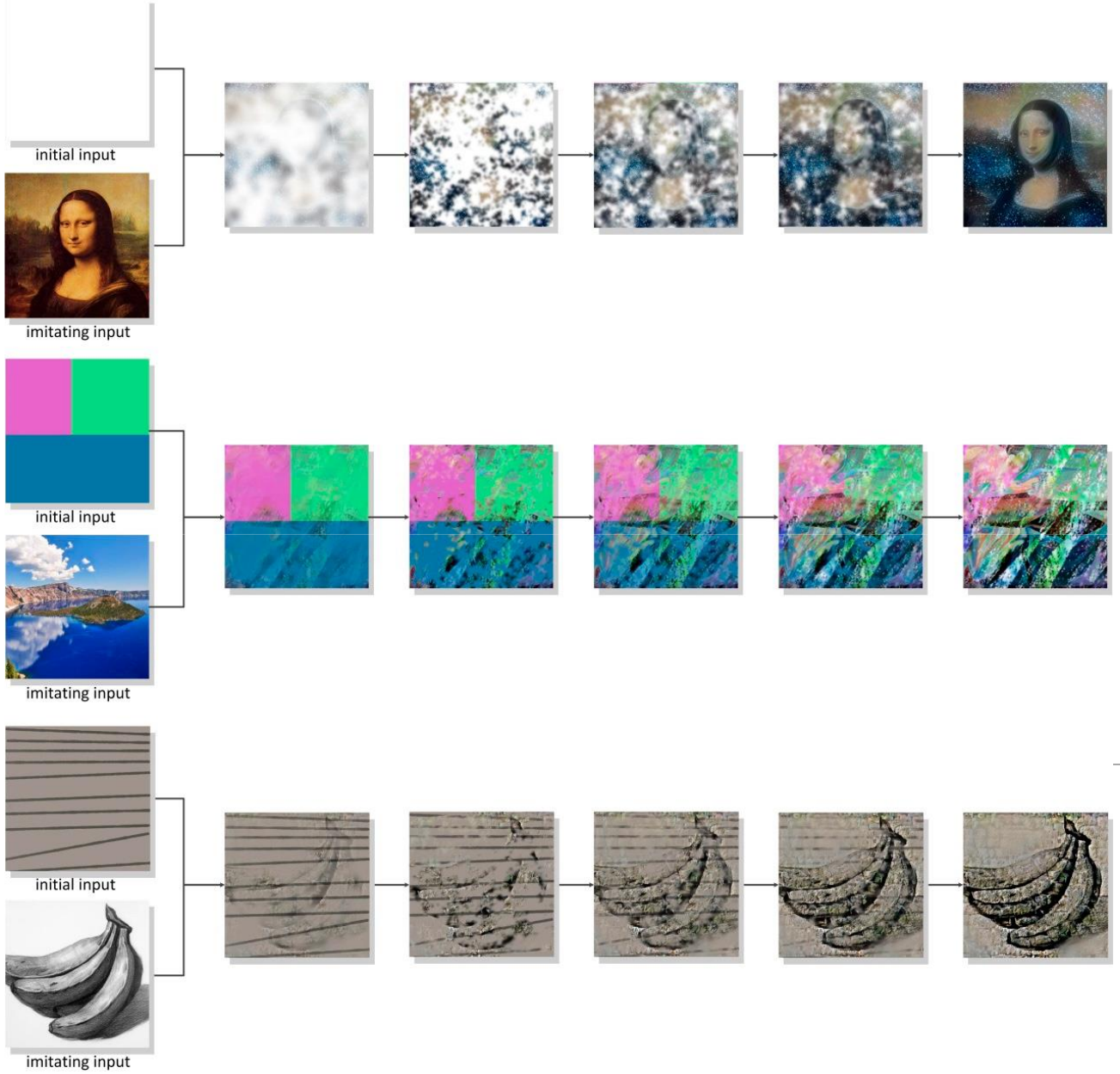
The inertial weight factor ω floats with the compactness of swarm. To prevent the particles become premature convergence, a greater inertial weight factor is needed. The inertial weight factor ω is self-adaptive and can be calculated by Equation (11).

$$\omega = \omega_{max} - \frac{k}{\text{iter}_{max}} (\omega_{max} - \omega_{min}) + r \tag{11}$$



ORIGINAL EVOLUTIONARY ART WITHOUT A TARGET. EACH ROW REPRESENTS A SINGLE RUN

The resolution of the canvas was set to 300:300, therefore the coordinate attributes x1 and x2 were restricted to x1 2 [0, 299] and x2 2 [0, 299]. The parameter p of CGP aesthetic fitness functions was set to 10. The functions used in our fitness functions were randomly selected as usual. Next, for the evolution method parameters, there are many parameter selection methods for PSO. We set the parameters in PSO as follows. The starting value of inertia weight was 0.9 and the ending value of inertia weight was 0.4. The range of inertia weight was restricted to (0.4, 0.9). The maximum number of iterations was set to 150. The acceleration factors c1 and c2 were both set to 2. It has been proven that the initial parameter values we used grant convergence and good exploration, and are still used in many state-of-art optimization algorithms. es during the iterations. The aesthetic styles are different because the aesthetic fitness functions were randomly.

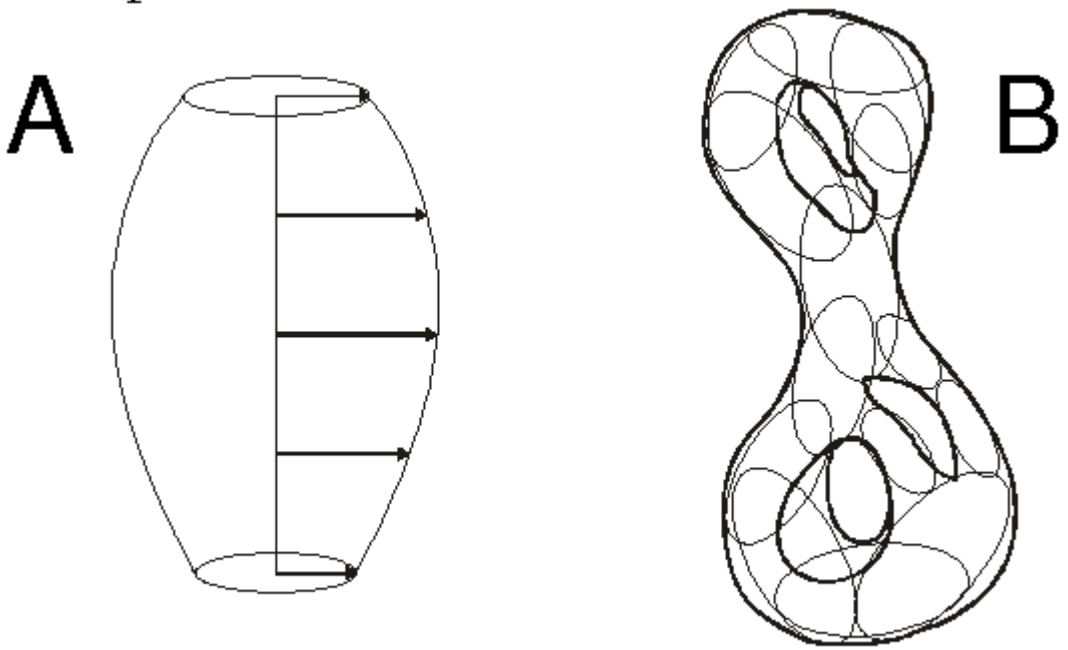


Imitating evolutionary art with a target:

The imitating runs as the previous original runs, except an additional target art input and the imitating factor **t = 0.2**. It shows the results of three independent runs. Every single run has two input graphs, four intermediate generations and one final imitating evolutionary art. Our method does not intend to reproduce the target art precisely.

The results demonstrate the tendency to evolve towards the target art, but also retain some original characteristics. Subjectively, we can feel the aesthetic feeling of final imitating evolutionary art and some intermediate generation art

low components allow increased freedom for evolutk



imisation of fixed parameterisation versus component-base

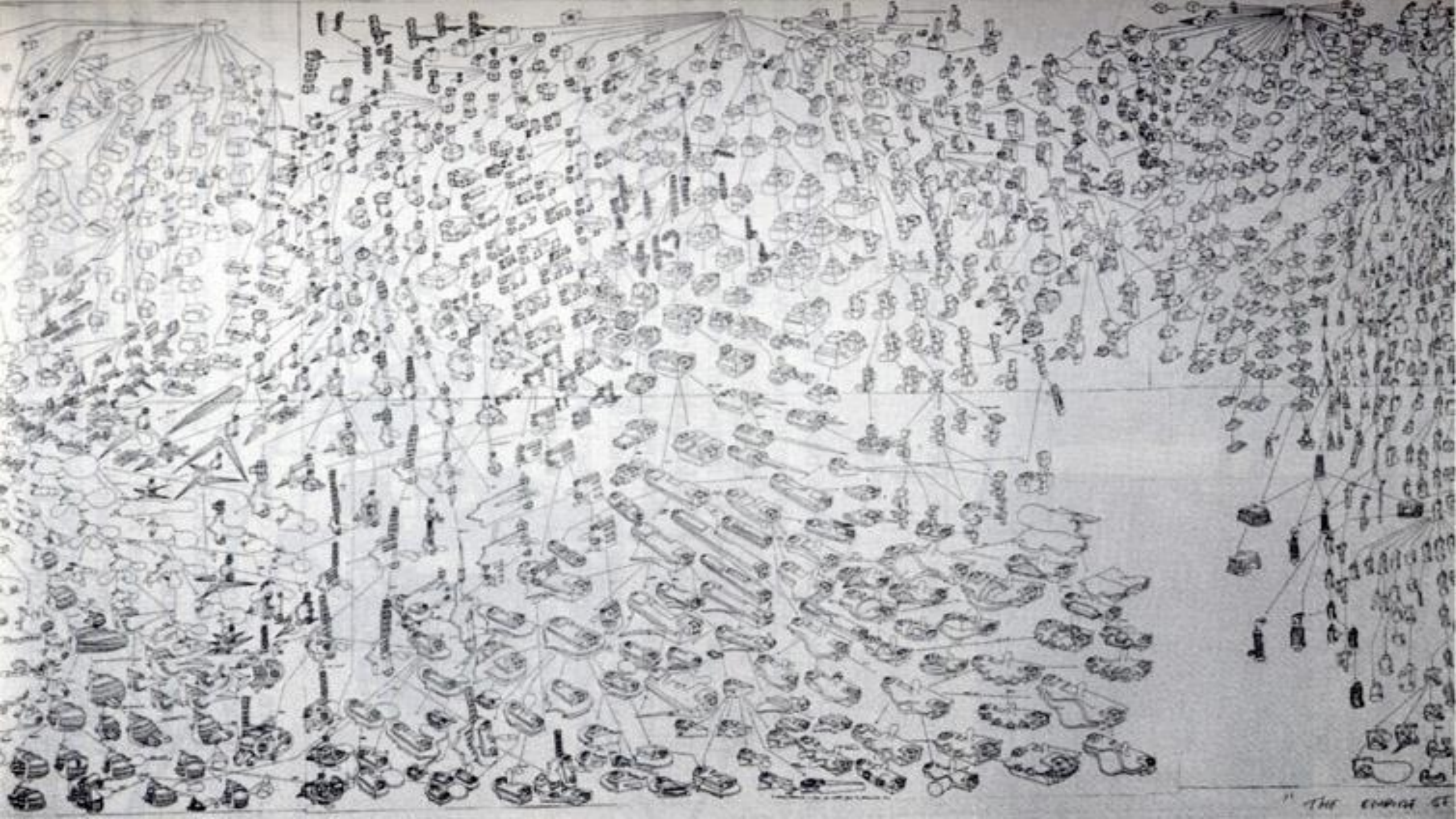
Component based representation

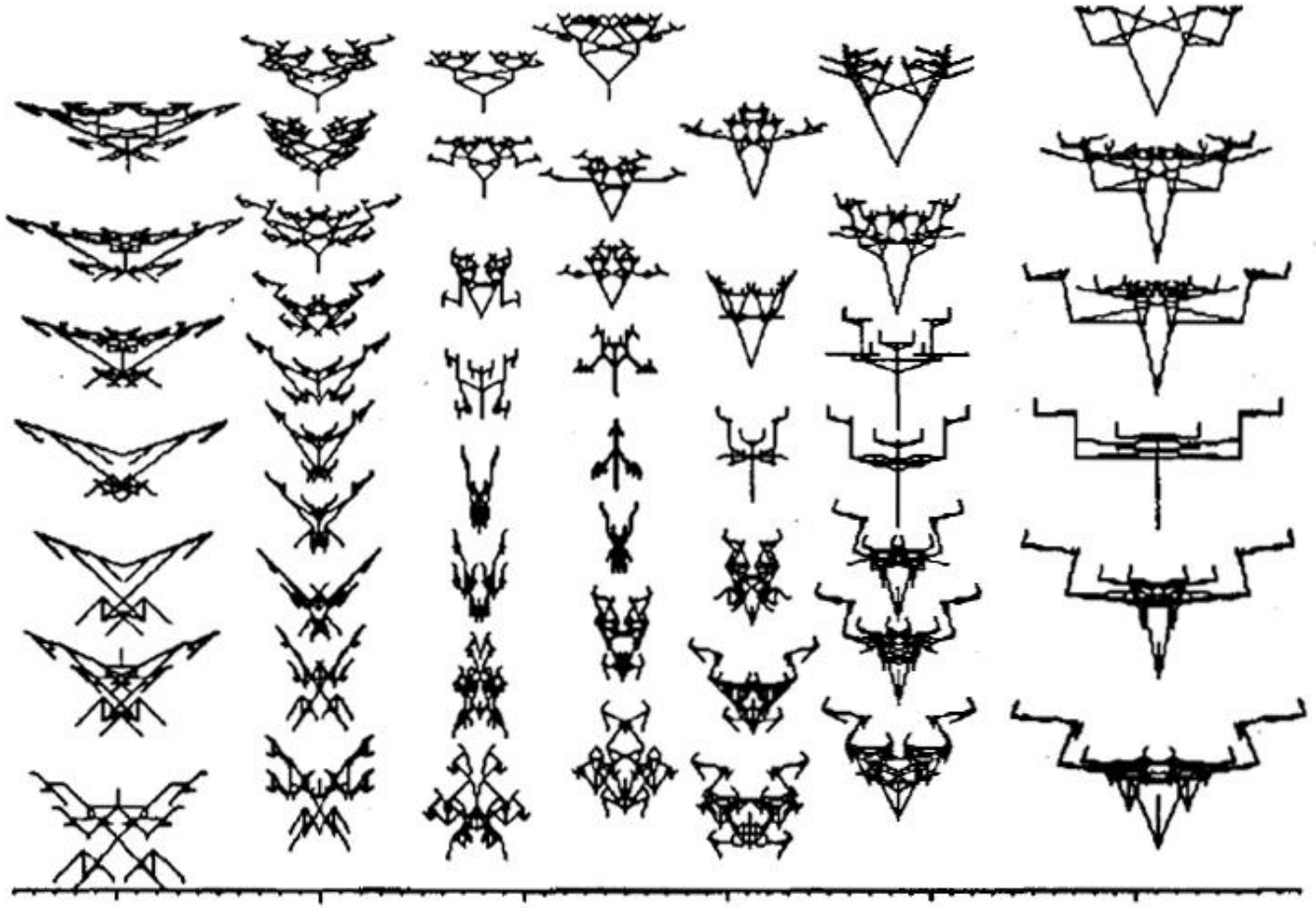
Fixed parameterization versus component-based exploration.

Shape A uses minimal parameters and is knowledge-rich (the height of the shape and the swept 2D outline is assumed by this representation).

Shape B is constructed by wrapping an envelope around a collection of 3D ellipses. By varying the relative positions and sizes of the ellipses, vastly more innovative and creative forms can be generated.

HTTPS://REDNUHT.ORG/GENETIC_CARS_2/

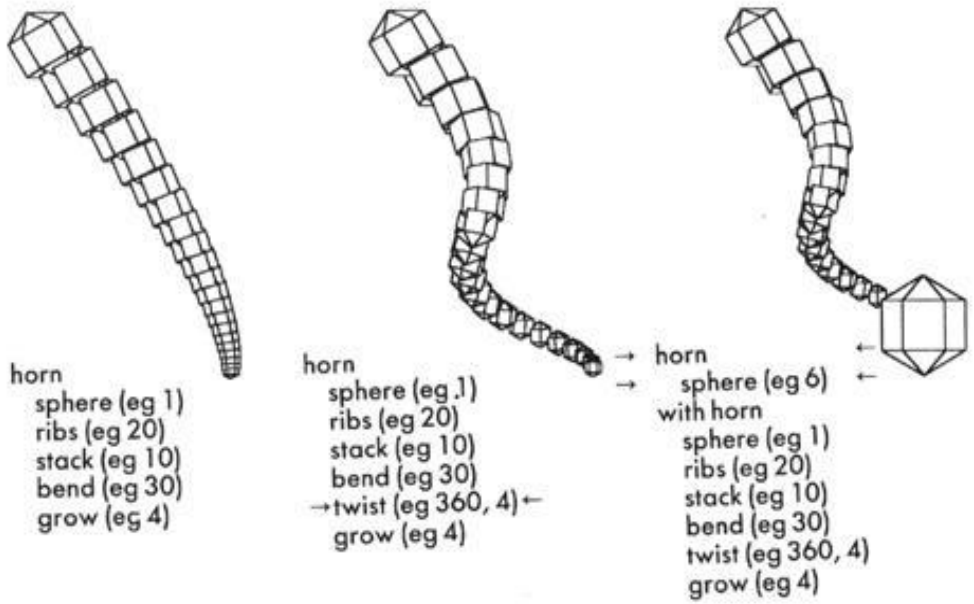




Richard Dawkins´ model shows the relation between genotypic codes ("genes") and phenotypic features ("biomorphs") realised by these codes in recursive procedures. The biologic developments proceed "bottom up" without a superior goal.

"Biomorphs" with forms similar to plants and animals grow out of combinations and repetitions of these genes.

DAWKINS, RICHARD: THE BLIND WATCHMAKER, 1986, EXAMPLES OF A MODEL FOR THE "EVOLUTION GAME" (DAWKINS: WATCHMAKER 1986, P.70, FIG.8).



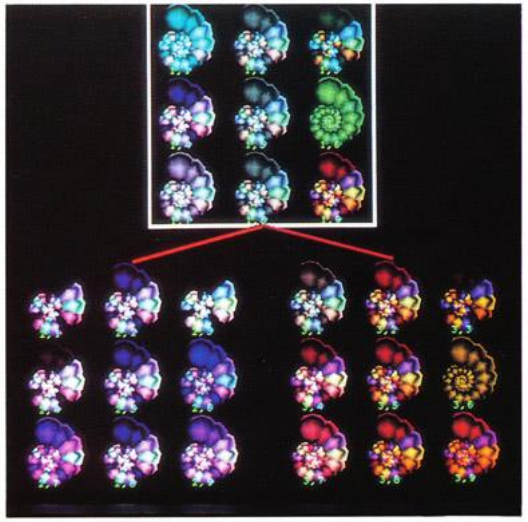
Latham presented these drawings of the series "Form Synth". "Form Synth" furnishes a vocabulary with three-dimensional elements being indicated by the CSG (Constructive Solid Geometry) program as line drawings. There are elements that can be combined as "horns". Via input to the editor Latham can determine the quantity and the combination manner of the elements. The element combinations in turn can build groups that are presented in the series "Mutations" (1991-92) in random order next to and beneath each other. "Continuous evolutions" can lead to "gene banks" for further selection phases and to animations with "life cycles" determining how long "genes" will reappear in an animation.

LATHAM, WILLIAM: HORNS, STRUCTURE MUTATION WITH THE SOFTWARE MUTATOR (TODD/LATHAM: EVOLUTIONARY ART 1992)

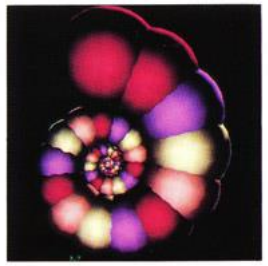
(31a) Sculpture Wire Frame.

(31b) Sculpture Rendered.

(31c) Sculptured Textured.



(31d,e) Evolution of Colour.



(31f.g.h.) Sequence of Frames from the Film "Mutations". Latham 1992.

Latham, William: Mutations

Latham describes himself as an "artist gardener" creating his "parody" of Artificial Life science by following aesthetic criteria.

The processing of forms in Latham´s and Todd´s "Evolutionary Art" doesn´t take care about biologic criteria. This indifference can be understood as a "parody" of the problems of the theoretical biology to design computer simulations as reconstructions of the laws of the natural cells´ development. "Form Synth" already demonstrates with its selection of basic elements and their combinations that it is not a biological model. Instead it only follows sugestions by "biological forms": "Our systems...often bear no relation to biological reality.“

But the computer cannot see and reward forms rich in content such as

- Griffins
- Witches faces
- Alien spaceships
- Lizards
- Surrealist Art

Griffins

Surrealistic Art

Conclusion

- Creative computers allow more innovative ideas to be explored in a short time
- Evolution is enabling our technology and arts to develop in suprising and exciting new ways.

Thank you!!