Differential Evolution for Self-adaptive Triangular Brushstrokes

Uroš Mlakar, Janez Brest, Aleš Zamuda

University of Maribor

{uros.mlakar,janez.brest,ales.zamuda}@um.si

Student Workshop on Bioinspired Optimization Methods and their Applications (BIOMA 2014)

13th International Conference on Parallel Problem Solving from Nature (PPSN 2014)

Ljubljana, Slovenia, September 13, 2014
Overview

1 Motivation and Related Work

2 Differential Evolution

3 The Proposed Method
   - Encoding
   - Genotype → Phenotype Rendering
   - Fitness Evaluation

4 Results

5 Conclusion
Motivation: Line Strokes

- Riley et al. (WCCI Barcelona, July 2010) compared 2 representations
  - variable-length classic genetic algorithm and
  - tree-based genetic algorithm.

- Line strokes to generate evolved images
  - best fitness was $\sim 9.4\%$ of the original image,
  - using a tree-based algorithm.
Motivation: Triangular Brushstrokes

- Izadi et al. (AJCAI, Adelaide, December 2010) used GP for the creation of non-photorealistic animations
  - unguided and guided searches: the guided yields better results,
  - filled and empty brushstrokes,
  - reported results are:
    - unguided: $\sim 5\%$
    - guided: $\sim 2\%$ - requires the source image in the phenotype rendering.
Differential Evolution (DE)

- A floating-point encoding EA for global optimization over continuous spaces,
  - through generations,
  - the evolution process improves population of vectors,
  - iteratively by combining a parent individual and several other individuals of the same population.
- We choose the strategy jDE/rand/1/bin
  - mutation: \( v_{i,G+1} = x_{r_1,G} + F \times (x_{r_2,G} - x_{r_3,G}) \),
  - crossover: \( u_{i,j,G+1} = \begin{cases} v_{i,j,G+1} & \text{if } \text{rand}(0,1) \leq CR \text{ or } j = j_{rand} \\ x_{i,j,G} & \text{otherwise} \end{cases} \),
  - selection: \( x_{i,G+1} = \begin{cases} u_{i,G+1} & \text{if } f(u_{i,G+1}) < f(x_{i,G}) \\ x_{i,G} & \text{otherwise} \end{cases} \),
  - includes mechanism of \( F \) and \( CR \) control parameters self-adaptation.
The Proposed Method (Encoding)

An individual encoded image is stored into a DE vector:
\[ \mathbf{x} = (x_1, x_2, ..., x_{8T^{\text{max}}}, F, CR, T^L, T^U), \]
size is \( D + 4, \ D = 8T^{\text{max}}, \)
the scaling factor \( F \) and crossover rate \( CR \) as used by the jDE,
then \( T^L \) and \( T^U \) follow.
The parameters \( T^L \) and \( T^U \) define the number of triangles \( T_i \):
- \( T_i \) rendered in the evolved image,
- \( T^L \) and \( T^U \) updated similarly as the \( F \) control parameter.
The Proposed Method (Genotype → Phenotype Rendering) 1/3

- DE vector $x_i, \forall i \in \{1, \ldots, NP\}$ constituting a genotype rendered into a phenotype image $z_i$ (to be compared against $z^*$).

- Each brushstroke is represented as $(c_x, c_y, r, \alpha_1, \alpha_2, b^Y, b^{Cb}, b^{Cr})$:

$$c_x \in [0, \ldots, R_x), \quad c_y \in [0, \ldots, R_y), \quad r \in \left[0, \frac{R_x}{\sqrt{T_{\text{max}}}}\right), \quad \alpha_1 \in [0^\circ, 360^\circ),$$

$$\alpha_2 \in [0^\circ, 180^\circ), \quad b^Y \in [16, 236), \quad b^{Cb} \in [16, 241), \quad b^{Cr} \in [16, 241).$$

- $c_x$ and $c_y$ define the center of the triangle to be rendered,
- $r$ defines its circumscribed circle,
- $\alpha_1, \alpha_2$ define the points of the triangle on the circumscribed triangle,
- $b^Y, b^{Cb}, b^{Cr}$ are the color components of its brush.
The triangle vertices encoded by $\mathbf{x}_i$ construct $T_i$ triangles,
each triangle $T_k = (c_x, c_y, r, \alpha_1, \alpha_2)$ defines vertices as in Figure on the right, Eq. 1.
For optimization, the YCbCr color space is used.
For rendering, the brush color $b_k^{YCbCr}$ is transformed to the RGB color space using the Eq. 2.

$$P_{1,k} = [(c_{x,k}+r_k \cos \alpha_1, c_{y,k}+r_k \sin \alpha_1)]$$
$$P_{2,k} = [(c_{x,k}+r_k \cos(\alpha_1+\pi), c_{y,k}+r_k \sin(\alpha_1+\pi))]$$
$$P_{3,k} = [(c_{x,k}+r_k \cos \alpha_2, c_{y,k}+r_k \sin \alpha_2)]$$

$$P_1 = (c_x + r \cos \alpha_1, c_y + r \sin \alpha_1)$$
$$P_2 = (c_x + r \cos(\alpha_1+\pi), c_y + r \sin(\alpha_1+\pi))$$
$$P_3 = (c_x + r \cos \alpha_2, c_y + r \sin \alpha_2)$$

$$b_k^R = [1.164(b_k^Y -16)+1.596(b_k^{Cr} -128)]$$
$$b_k^G = [1.164(b_k^Y -16)-0.813(b_k^{Cr} -128)-0.391(b_k^{Cb} -128)]$$
$$b_k^B = [1.164(b_k^Y -16)+2.018(b_k^{Cb} -128)]$$
For each triangle $T_k$, a solid color is rendered, over the brush area with a transparency factor $\frac{1}{T_i}$, which makes the color of the brush: $b_k = \lfloor \frac{255}{T_i} b^k_{RGB} \rfloor$; this is analogous to blending each triangle as part-transparent triangle within the evolved image:

$$z^k_{x,y} = \sum_{T_k \text{ over } (x,y)} \lfloor \frac{255}{T_i} b^k_{RGB} \rfloor.$$

Triangles defined over the edges of the image canvas are drawn by clipping away pixels outside of the canvas area.
Fitness Evaluation

- After a phenotype image $z_i$ is rendered: it is compared to a reference image $z^*$ using the evaluation metric:
  
  $$f(z) = 100 \times \frac{\sum_{y=0}^{R_y-1} \sum_{x=0}^{R_x-1} |z_{x,y}^* - z_{x,y}^R| + |z_{x,y}^* - z_{x,y}^G| + |z_{x,y}^* - z_{x,y}^B|}{3 \times 255 \times R_x R_y}.$$ 

- The obtained result is the similarity of the evolved image and the reference image.

- The goal of the evolutionary process is to minimize the function value $f(z)$. 
Results (Experimental Setup)

- The parameter sets are:
  - $NP = \{25, 50, 100\}$,
  - $T_{max} = \{10, 20, ..., 150\}$,
  - $RN_i = \{0, 1, ...51\}$,
  - $MAXFES = 1e+5$.

- A total of 45 parameter settings, 2340 independent runs.
- Rendering: GDI+.
- The experiments conducted on 4 images of size $100 \times 100$ pixels.

- Additional experiment: all images evolved up to $MAXFES = 1e+6$. 

Baboon  Liberty  Palace  Vegetables

Uroš Mlakar, Janez Brest, Aleš Zamuda (UM)  DE triangular brushstrokes evolution
Best fitness values for all parameter sets for all images, $\text{MAXFES} = 1 \times 10^5$

Table 1. Obtained fitness over $T_{\text{max}}$ and $N_P$: test instances Liberty and Palace.

| $N_P$ | $T_{\text{max}}$ | Best | Liberty | Worst | Average | STD | Best | Liberty | Worst | Average | STD | Palace | Best | Liberty | Worst | Average | STD | Palace | Best | Liberty | Worst | Average | STD |
|-------|-----------------|------|---------|-------|---------|-----|------|---------|-------|---------|-----|---------|------|---------|-------|---------|-----|---------|-------|---------|-----|
| 10    | 16.33           | 14.11| 15.85   | 15.85 | 15.85   |     | 14.11| 15.85   | 15.85 | 15.85   |     | 14.11   |      |         |       |         |     |         |      |         |     |
| 20    | 15.85           | 14.11| 15.85   | 15.85 | 15.85   |     | 14.11| 15.85   | 15.85 | 15.85   |     | 14.11   |      |         |       |         |     |         |      |         |     |
| 30    | 15.85           | 14.11| 15.85   | 15.85 | 15.85   |     | 14.11| 15.85   | 15.85 | 15.85   |     | 14.11   |      |         |       |         |     |         |      |         |     |
| 40    | 15.85           | 14.11| 15.85   | 15.85 | 15.85   |     | 14.11| 15.85   | 15.85 | 15.85   |     | 14.11   |      |         |       |         |     |         |      |         |     |

Table 2. Obtained fitness over $T_{\text{max}}$ and $N_P$: test instances Vegetables and Baboon.

<table>
<thead>
<tr>
<th>$N_P$</th>
<th>$T_{\text{max}}$</th>
<th>Best</th>
<th>Vegetables</th>
<th>Worst</th>
<th>Average</th>
<th>STD</th>
<th>Best</th>
<th>Vegetables</th>
<th>Worst</th>
<th>Average</th>
<th>STD</th>
<th>Baboon</th>
<th>Best</th>
<th>Vegetables</th>
<th>Worst</th>
<th>Average</th>
<th>STD</th>
<th>Baboon</th>
<th>Best</th>
<th>Vegetables</th>
<th>Worst</th>
<th>Average</th>
<th>STD</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>16.33</td>
<td>14.11</td>
<td>15.85</td>
<td>15.85</td>
<td>15.85</td>
<td></td>
<td>14.11</td>
<td>15.85</td>
<td>15.85</td>
<td>15.85</td>
<td></td>
<td>14.11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>15.85</td>
<td>14.11</td>
<td>15.85</td>
<td>15.85</td>
<td>15.85</td>
<td></td>
<td>14.11</td>
<td>15.85</td>
<td>15.85</td>
<td>15.85</td>
<td></td>
<td>14.11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>15.85</td>
<td>14.11</td>
<td>15.85</td>
<td>15.85</td>
<td>15.85</td>
<td></td>
<td>14.11</td>
<td>15.85</td>
<td>15.85</td>
<td>15.85</td>
<td></td>
<td>14.11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>15.85</td>
<td>14.11</td>
<td>15.85</td>
<td>15.85</td>
<td>15.85</td>
<td></td>
<td>14.11</td>
<td>15.85</td>
<td>15.85</td>
<td>15.85</td>
<td></td>
<td>14.11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Uroš Mlakar, Janez Brest, Aleš Zamuda (UM) DE triangular brustrokes evolution
The fitness convergence graph of the best runs for all images.
Results ($T_i$ Dynamics Graph)

The dynamics of the number of triangular brushstrokes in the best vectors.
Baboon

Results (Image: Baboon)
Results (Image: Liberty)

Liberty

[Series of images showing a progression of triangle patterns resembling Liberty]

[Image of the Statue of Liberty]
Results (Image: Palace)

Palace

![ Palace images ]
Results (Image: Vegetables)

Vegetables
An evolvable lossy image representation using a jDE algorithm.

The performance of this encoding: competitive with the GA tree representation.

Experiments show promising results on sample images.

In the future we would like to address:
  - different evolutionary operators,
  - change control-parameters updating, and
  - testing on more images with different properties.
Thank you.

Questions?