

Maintaining Population Diversity in Evolutionary Art using Structured Populations

Eelco den Heijer

Objectionation and VU University Amsterdam
Faculty of Sciences
Amsterdam, Netherlands
eelcodenheijer@gmail.com

A.E. Eiben

VU University Amsterdam
Faculty of Sciences
Amsterdam, Netherlands
gusz@cs.vu.nl

Abstract—Maintaining population diversity is an important and difficult task in Evolutionary Computation in general and Evolutionary Art in particular. A lack of population diversity will result in inefficient search behaviour and premature convergence. In this paper we investigate the effect of using spatially structured populations on population diversity in Evolutionary Art. To this end, we perform several experiments with unsupervised evolution (no human in the loop) of aesthetically pleasing images using a panmictic model Evolutionary Algorithm, a distributed Island Model (with a Best-First selection scheme and with the Multikulti algorithm) and a Cellular Evolutionary Algorithm. In our Island Models experiments we use a number of different parameters settings for number of islands, island size, migration interval, migration size, and initialisation methods. In our Cellular EA experiments we use different settings for width, height and neighbourhood. We also compare the use of structured populations with the use of a panmictic EA with enhanced genetic operators. We find that the use of structured populations is beneficial for maintaining both phenotype and genotype diversity. All configurations of Island Models and Cellular EA outperform our standard panmictic EA on population diversity.

Keywords—Evolutionary Art, Population Diversity, Genetic Programming, Cellular Evolutionary Algorithms, Island Models

I. INTRODUCTION

Evolutionary Art (EvoArt) is a thriving research discipline that investigates the application of methods from Evolutionary Computation (EC) in the creation of aesthetically pleasing content. Population Diversity is the amount of variance among the individuals in a population, and is important in EC in general and in EvoArt in particular. A loss of population diversity will lead to inefficient search behaviour and may lead to premature convergence. Unsupervised evolutionary art is a subfield within EvoArt that investigates the evolution of aesthetically pleasing images (or ‘art’) without human aesthetic evaluation; all fitness evaluations are performed by aesthetic measures. Many aesthetic measures come from a young research field called ‘computational aesthetics’. The formalisation of aesthetic appreciation into models and algorithms is a daunting task, and there is reasonable consensus on the observation that most current aesthetic measures act primarily as *heuristics* in the search process [5], [15]. Another observation is that EvoArt is inherently more concerned with exploration than with exploitation [15], [16]. The ability of an EvoArt system to perform exploration is closely linked to its creative potential output; Margaret Boden defines creativity as the ability to create novel, surprising and valuable ideas

[6]. In [7] Margaret Boden describes three types of creativity; combinational, exploratory and transformational creativity. Combinational creativity is the process of creating novel ideas by combining existing ideas in unexpected ways. Exploratory creativity is the process of producing novel ideas by starting from an existing idea, and changing that idea in small steps to ‘explore’ the surrounding conceptual space for novel ideas. Transformational creativity is the most radical form of creativity, in which one changes the conceptual space (or one ‘changes the rules’). We believe that the first two forms of creativity (combinational and exploratory) are within reach of current EvoArt systems. Both forms of creativity do require a high degree of exploration, which suggests that EvoArt needs a high(er) level of exploration in comparison with other EC domains. The need for exploration mandates a continuous high population diversity in the EvoArt system, both for unsupervised EvoArt systems and for IEC EvoArt systems. For this reason, we think that population diversity is an important issue in both supervised and unsupervised evolutionary art. In several EC publications (e.g. [14]), phenotype diversity of the population equals the spread of fitness values, and the phenotype distance between two individuals equals the difference in their fitness values. We think that this assumption does not hold in the domain of EvoArt; two images can be quite similar to a human observer, but may be evaluated differently by an aesthetic measure. And two visually different images may receive the same fitness evaluation by an aesthetic measure. We therefore suggest that phenotype diversity in EvoArt should be calculated by the difference between images that are produced by the genotypes, and in this paper we calculate the phenotype diversity using a well-known image distance measure (Stricker & Orengo, see Section IV-B). This paper is the second in a series on maintaining population diversity in evolutionary art. In previous work we investigated the effect of using custom crossover, mutation and initialisation on population diversity. In future work we intend to investigate the effect of niching techniques and fitness sharing on population diversity in EvoArt systems, and in this paper we compare the standard, panmictic EA model with two models of structured populations; Island Models or IM and Cellular Evolutionary Algorithms or CEA.

Our research questions are

- 1) Can we maintain and/ or increase population diversity in an evolutionary art system by using an IM approach and/ or a CEA approach?
- 2) IM and CEA have a number of parameters in addi-

tion to the ‘standard’ evolutionary parameters; What parameters within IM and CEA are important for population diversity in our EvoArt system?

- 3) Will an increase in population diversity, using any of the aforementioned methods, result in less efficient search behaviour (i.e. a slower increase in fitness)?
- 4) In previous work we have investigated the effect of custom genetic operators (which perform a local search to increase diversity) in a panmictic EA. How well does this panmictic EA with extended genetic operators initialisation, mutation and crossover [15] compare to the structured population configurations (IM, IM with Multikulti and CEA) on maintaining population diversity?

The rest of the paper is structured as follows; first we discuss related work in Section II. Next, we discuss population diversity and spatially structured populations (IM and CEA) in Section III. In Section IV we describe our fitness function, our two distance functions that we use to measure population diversity and our experiments. The results of the experiments are presented in Section V and we end this paper with conclusions and directions for future work in Section VI.

II. RELATED WORK

Maintaining population diversity is an important topic within EC, and the literature on the topic is extensive. In this section we will mention work that has been done on maintaining population diversity with EvoArt and GP, and we will mention literature on the use of spatially structured populations in order to preserve population diversity. In previous work [15] we used a genotype distance function and a phenotype distance function to perform a local search step to our crossover and mutation operators. Adding custom genetic operators did increase phenotypic diversity, but the added computational costs were high.

There have been a number of publications on the maintenance of population diversity in creative ecosystems. McCormack and Bown describe an approach using a creative ecosystem where organisms change their environment in which they operate using a technique called ‘niche construction’ [19], [8]. Using this approach, their system is able to maintain a high level of diversity in the artistic output.

Maintaining population diversity in GP has been studied extensively; a good overview is [10].

Using island models to maintain population diversity has also been researched extensively, good overviews are [23] and [1]. Araujo and Merelo extended the standard Island Model EA with a migration policy that favours the exchange of ‘different’ individuals (as opposed to more conventional policies where ‘best’ individuals are exchanged); they named their algorithm the Multikulti algorithm [3]. In section IV we will describe a number of experiments in which we use the Multikulti algorithm in our EvoArt system.

The literature on IM is extensive, most notably on the role the IM specific parameters (such as migration size, interval, etc.). Cantú-Paz has investigated the influence of different migration policies [11], [12]. Skolicki et al investigated the role of migration size and migration interval within IM and found that these two parameters play an important role in the success of IM [21].

Cellular EAs originate from work done in the parallelisation of EAs in the early 1990s, a good overview of the field is [23]. Alba et al have investigated various layouts of the grid and toroidal populations (using different widths and heights) [1], and we use a number of their findings in our CEA experiments (see Section IV-G).

III. STRUCTURED POPULATIONS

In a canonical EA the selection of individuals for crossover and mutation is usually done across the entire population, and this is called the *panmictic* model. In the early nineties, several EC researchers suggested ideas to restrict the selection to parts of the population (in order to try to solve multimodal problems), which lead to the idea of spatially structured populations.

Island Models; a well-known example of a spatially structured population is the Island Model or IM. Island Models (IM) are distributed models of EA, where individuals are distributed among isolated islands. At certain intervals (migration interval) the islands exchange one or more individuals (migration size). Islands are connected to each other according to a certain island topology. Popular islands topologies are ring, fully connected, star, small world, random and dynamic. The isolated

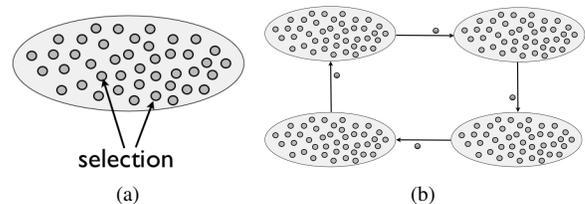


Figure 1. (a) A panmictic population performs selection throughout the entire population, (b) Island models perform selection only within islands, and perform periodic migration to exchange individuals.

nature of the islands prevents the domination of the entire population by one or a few individuals, and the exchange of individuals prevents premature convergence on the islands. Several researchers have reported promising results with using IM [1], [3], [23] but the use of IM comes with additional complexity; next to the standard evolutionary parameters (population size, tournament size, etc.) IM also require parameters for migration (size, interval and selection), islands (number, size and topology). Good overviews of IM are [23] and [1]. Araujo et al tested a number of migrant selection policies (select most fittest, select random, select most different) and found that exchanging the most different migrant between islands often results in efficient search behaviour and in high levels of population diversity [3]. The authors called their selection policy ‘Multikulti’. Since we intend to increase and maintain population diversity in our EvoArt system, we have added this Multikulti method in our IM implementation, and have done a number of experiments with them (see Section IV-F).

Cellular Evolutionary Algorithms; another well-known implementation of the idea of spatially structure populations is the Cellular Evolutionary Algorithm (CEA). Cellular EAs, or Lattice Cellular EAs have been around since the dawn of Evolutionary Computation [18]. A CEA has a structured population of a particular form, such as the one dimensional

line, the one dimensional ring, or the 2D grid (either flat or toroidal). Each individual has a fixed location (or coordinate) in this population structure. A CEA defines a neighbourhood of each individual and selection is only performed on this neighbourhood [23]. The size, shape and neighbourhood have an important influence on the search behaviour [2]. Since

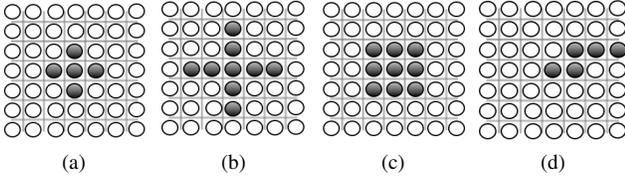


Figure 2. Three popular neighbourhood functions used in CEA; (a) linear 5 (also known as von Neumann neighbourhood), (b) linear 9, (c) compact 9 (also known as Moore neighbourhood) (d) one possible outcome of random walk 4 (take 4 random steps -up, down, left or right - from the original cell, ignore previous steps, and ignore the original cell). Random walk is taken from [13]

the selection is only performed on the local neighbourhood, diffusion of fit individuals is slow, and CEAs are usually more exploratory than panmictic EAs (see [23], chapter 4). Well-known neighbourhood functions are linear5, linear9, compact9 and random walk (see Figure 2). In addition, one can vary the width and height of a CEA; a thinner topology (where width > height) gives lower selection pressure [23], and tend to be more efficient in multi-modal problems [2].

IV. EXPERIMENTAL SETUP

In this section we briefly describe our experimental setup. First, we describe our two distance functions. We calculate genotype diversity using the Ekárt & Németh distance function for expression trees (Section IV-A). Next, we calculate phenotype diversity using an image distance function by Stricker & Orengo (Section IV-B), and we describe the Ralph & Ross aesthetic measure that we use as our fitness function (Section IV-C). In Section IV-D we describe the methodology used in the experiments described in this paper

A. Genotype Distance

The structural distance metric by Ekárt and Németh is an efficient and fast metric for expression trees. The metric calculates the distance between two expression trees by performing a node by node comparison of the nodes of the expressions. If no node is present in one of the two expressions, a ‘null’ node is used in the comparison. The metric uses several rules for the different types of nodes (literals, functions, null etc.), and we refer to [17] for details.

B. Phenotype Distance

In several papers on population diversity, the phenotype distance between two individuals equals their difference in fitness. In EvoArt systems this observation would be very difficult to maintain. Let us imagine two very different images I_a and I_b , and suppose these images would score equal using our aesthetic measure; this would mean that our images are equivalent, but certainly does *not* mean that they are equal. Therefore, it is necessary to use an image distance function to calculate the phenotype distance between individuals. To

Table I. IMAGE FEATURES AND THEIR WEIGHTS USED IN OUR STRICKER & ORENGO IMAGE DISTANCE FUNCTION

Image feature	Weight
Hue (avg)	4
Hue (sd)	4
Hue (skewness)	4
Saturation (avg)	1
Saturation (sd)	1
Saturation (skewness)	1
Intensity (avg)	2
Intensity (sd)	2
Intensity (skewness)	2
Colourfulness (avg)	2
Colourfulness (sd)	2
Colourfulness (skewness)	2

this end, we have implemented the Stricker & Orengo image distance function [22]. This distance function d_{so} computes the distance between two images I_a and I_b by calculating the distance between the two image feature vectors v_a and v_b , where

$$d_{so}(I_a, I_b) = \frac{\sum_{i=0}^{i<N} w_i \cdot |v_{a_i} - v_{b_i}|}{\sum_{i=0}^{i<N} w_i} \quad (1)$$

where N is the number of image features (in our implementation $N = 12$, see Table I for the 12 image features). For the image features we used the average, standard deviation and skewness of the hue, saturation, intensity and colourfulness of the colour pixels of the image (in HSV colour space). Each image feature is assigned a weight w and the weights are shown in Table I.

C. Ralph & Ross Bell Curve

In all experiments in this paper we used one aesthetic measure as a fitness function; the Ralph & Ross [20] bell curve. We have performed many experiments with a number of aesthetic measures, and we chose the Ralph & Ross bell curve for the experiments in this paper because it can be regarded as a ‘difficult’ aesthetic measure; increase in fitness using this aesthetic measure is typically slow when compared to other aesthetic measures. The aesthetic measure by Ross & Ralph is based on the observation that many paintings exhibit functions over colour gradients that conform to a normal or bell curve distribution [20]. The authors suggest that works of art should have a reasonable amount of changes in colour, but that these changes in colour should reflect a normal distribution (hence the name ‘Bell Curve’). The calculation takes a number of steps, and we refer to the original paper for details [20].

D. Methodology

In our experiments we chose a number of evolutionary parameters, and we will present them here. First of all, the comparison between the panmictic EA, IM and CEA should be as ‘fair’ as possible, so we decided to make all populations the same size, 256. The island models use N islands of M individuals such that $N \times M = 256$. In similar fashion, for the CEA we use a toroidal grid of $w \times h$ such that $w \times h = 256$. Next, we performed 20 generations in all experiments, and performed 30 runs for each configuration. Each generation we calculated average fitness, average genotype diversity and average phenotype diversity. In the case of IM, each island sent its entire island population to a central broker (since we wanted to calculate the overall population diversity, not just

Table II. FUNCTION AND TERMINAL SET OF OUR EVOLUTIONARY ART SYSTEM

Terminals	x,y, ephem_double, golden_ratio, pi
Basic Math	plus/2, minus/2, multiply/2, div/2, mod/2
Other Math	log/1, sinh/1, cosh/1, tanh/1, atan2/2, hypot/2, log10/1, squareroot/1, cone2/2, cone3/2, cone4/2
Relational	minimum/2, maximum/2, ifthenelse/3
Bitwise	and/2, or/2, xor/2
Noise	perlinnoise/2, fbm/2, snoise/2, vnoise/2, marble/2, turbulence/2, plasma/2
Boolean	lessthan/4, greaterthan/4
Other	smooth_noise/2, moire/2, chaoticdust/2, parabol/2

Table III. GENERIC EVOLUTIONARY PARAMETERS OF OUR EVOLUTIONARY ART SYSTEM USED IN OUR EXPERIMENTS; THE PARAMETERS ARE USED IN OUR PANMICTIC MODEL AND IN OUR IM AND CEA EXPERIMENTS. SPECIFIC IM AND CEA PARAMETERS ARE GIVEN IN TABLE IV AND TABLE V.

Symbolic parameters	
Representation	Expression trees
Initialisation	Ramped half-and-half (depth between 2 and 5)
Survivor selection	Tournament, Elitist (best 1)
Parent Selection	Tournament (size 2) (for MI) For CEA: tournament 2 on neighbourhood
Mutation	Point mutation
Recombination	Subtree crossover
Fitness function	Ralph & Ross Bell Curve
Numeric parameters	
Population size	256 (all configurations)
Generations	20
Runs	30
Crossover probability	0.90
Mutation probability	0.10
Maximum tree depth	8

the diversity on the island itself). We calculate population diversity by calculating the average distance (either genotypic or phenotypic) between each individual in the population.

The genotype diversity and phenotype diversity were calculated by calculating the average genotype distance (using our genotype distance function described in Section IV-A) between each individual. The phenotype distance was calculated in the same way, using our image distance function described in Section IV-B. For the panmictic setup and the CEA experiments this was straightforward, since there is only one population in these setups.

E. Panmictic

We performed one experiment with a standard panmictic model that served as a baseline for the other experiments. All parameters are given in Table III. This configuration has label ‘Pan’. In previous work we investigated the effect of using custom genetic operators initialisation, crossover and mutation in a panmictic EA. The operators perform a small local search in each step, and choose the most distant individual (for more detail, we refer to [15]). In order to compare the use of custom genetic operators with the use of structured populations, we did an additional experiment with a panmictic EA with these custom genetic operators (we call this configuration ‘Pan2’).

F. Island Models

We performed 10 different experiments (using 10 different configurations) with Island Models; 5 experiments with a

Table IV. PARAMETERS SETTINGS OF OUR ISLAND MODEL EXPERIMENTS (PLAIN ISLAND MODELS AND ISLAND MODELS WITH THE MULTIKULTI ALGORITHM)

Name	Islands	Island size	Number Ind.	Migr. Int.	Migr. Size
im1/ mk1	2	128	256	5	2
im2/ mk2	4	64	256	5	2
im3/ mk3	8	32	256	5	2
im4/ mk4	16	16	256	3	1
im5/ mk5	32	8	256	3	1

Table V. SPECIFIC PARAMETERS SETTINGS OF OUR CELLULAR EA EXPERIMENTS. THE NEIGHBOURHOODS ARE EXPLAINED IN FIGURE 2.

Name	Width	Height	Number Individuals	Neighbourhood
cea1	16	16	256	Linear 5
cea2	16	16	256	Linear 9
cea3	16	16	256	Compact 9
cea4	32	8	256	Linear 5
cea5	64	4	256	Linear 5
cea6	16	16	256	Random Walk 4

generic island model setup, and 5 experiments with island models using the Multikulti algorithm. In our experiments we varied migration interval, migration size, number of islands, migration selection policy, island size and initialisation method. We did *not* vary island topology (all experiments use a ring topology), replacement selection (select unfittest) or total number of individuals (256). In our generic IM configurations, we use a selection scheme whereby an island sends the fittest individuals as migrants to other islands. The standard IM configurations are labelled ‘IM1’ to ‘IM5’. In the 5 experiments with the Multikulti algorithm, each island sends its most distant individuals to other islands. The distance is calculated using the Stricker & Orengo image distance function (see Section IV-B). Apart from the difference in migrant selection policy, all settings are the same as the settings for the generic IM experiments. The Multikulti configurations are labelled ‘MK1’ to ‘MK5’.

G. Cellular EA

We implemented a cellular EA with population size of 256. In the default setting, the CEA has a dimension of 16×16 , a default Linear5 neighbourhood (see Figure 2a). In our experiments with CEA we varied the neighbourhood of the CEA and the width and height. Alba et al found that ‘thin’ CEAs (where width > height) have lower selection pressure and perform better in multi-modal problems [2]. Experiments cea2 and cea3 use a different neighbourhood function. In our CEA experiments 4 and 5 (cea4, cea5, see Table V) we use a layout of 32×8 and 64×4 respectively. In CEA experiment 6 we use a custom random walk neighbourhood that does 4 random steps in the neighbourhood (taken from [13]).

V. RESULTS

In this paper we tested 1 panmictic configuration, 5 configurations with island models using a ‘select best’ selection scheme, 5 island model configurations using the Multikulti algorithm, and 6 configurations using a Cellular EA. We ran each configuration 30 times, and calculated the average fitness, average genotype diversity and average phenotype diversity in each generation. The results of our experiments are presented in Figure 3.

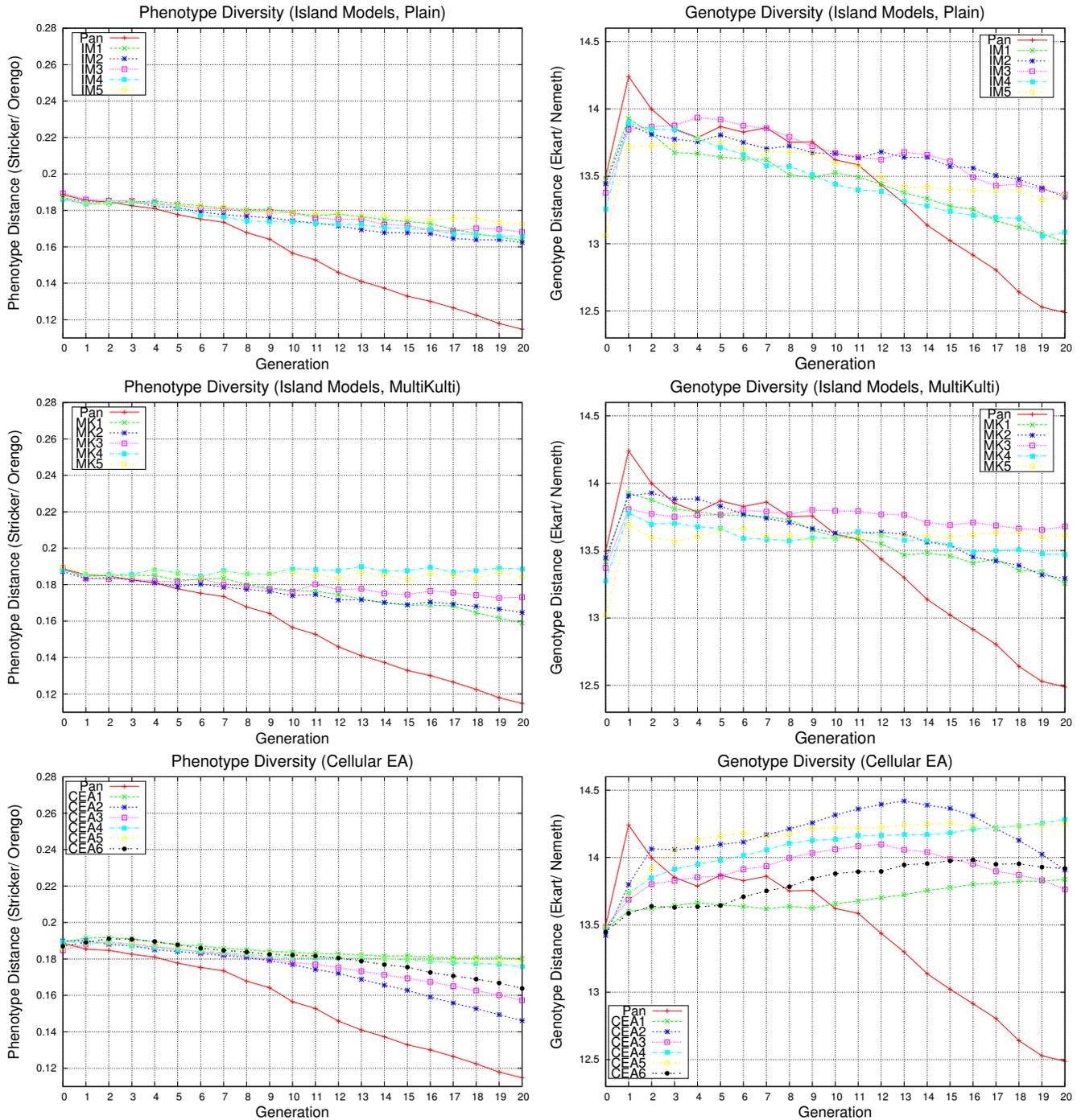


Figure 3. The resulting phenotype distance (left) and genotype distance (right) for 'Plain' Island Models (top), Island Models with the MultiKulti algorithm (middle), and Cellular EA (bottom). All numbers are averaged over 30 runs. 'Pan'- Panmictic, 'IM' - Island Models, 'MK' - Island Models with the Multikulti algorithm, 'CEA' - Cellular EA.

At first, we see that in our panmictic model EA (Pan) the phenotype diversity decreases over time, which is typical for the exploitation phase of an EA; average diversity decreases and average fitness increases. Next, we see that all uses of structured populations have a positive effect on the progress of both genotype diversity and phenotype diversity. When using IM, IM with Multikulti or CEA, the phenotype diversity either remains the same (as in the Multikulti configurations MK4 and MK5), or decreases rather slowly (MK1, MK2, MK3, IM3 and IM5). In general, IM with Multikulti scores a bit better on maintaining phenotype diversity than IM with the standard ‘select best migrant’ scheme (MK3, MK4 and MK5 all score higher than all ‘plain’ IM configurations). MK4 and MK5 also score better than all CEA configurations; the CEA configurations that score highest on phenotype diversity perform similar to the average MK configurations. CEA2 and CEA3 perform worst on phenotype diversity, they score worse than all MK and all IM configurations, but still perform better on phenotype diversity than our panmictic model (Pan).

When we look at the progress of phenotype and genotype diversity of the IM and MK configurations, we can detect a vague ‘step’ pattern in a number of configurations. These steps are caused by the migration intervals, and the length of each step corresponds to the length of the migration interval (either 3 or 5). The spike after migration is followed by a temporary decline in phenotype diversity, which suggest that the new migrant has produced visually similar offspring (thereby reducing phenotype diversity) or that the new migrant has not been selected for crossover/ mutation, and has disappeared in subsequent generations (thereby also reducing phenotype diversity). It is interesting to note that configurations with more islands perform better on the progress of phenotype diversity over 20 generations. The results of our Island Model experiments (both IM and MK) suggest an improvement of phenotype diversity when using more islands; in our IM experiments IM3, IM4, and IM5 (8, 16, and 32 islands) score slightly better than IM1 and IM2 (2 and 4 islands), and in our Multikulti experiments MK4 and MK5 perform best with 16 and 32 islands respectively.

The results from our CEA experiments show a similar picture (Figure 3). From the 6 configurations with CEA, 3 perform much better than the panmictic model (CEA1, CEA4 and CEA5), and 3 perform only a little better on progress of phenotype diversity (CEA2, CEA3, CEA6). When we compare the CEA results with the IM results, we see that 2 CEA configurations (CEA3, CE2) score less on phenotype diversity than the worst MK performer on phenotype diversity (MK1). When we look at the progression of genotype diversity over 20 generations, we see the following; all IM configurations perform better on genotype diversity than the panmictic model, and all CEA configurations also perform better than the panmictic EA. The CEA configurations CEA2 and CEA3 appear to have a peak of genetic diversity around the 11th to 13th generation. Figure 4 shows the progress of fitness for the IM, MK and CEA configurations. It is apparent that the IM and MK configurations perform worse on fitness progress than the CEA configurations. IM and MK even perform worse on fitness progress than the panmictic EA. Selection pressure is clearly high in the CEA configurations, especially in the configurations CEA2 and CEA3, and selection pressure is clearly low in the IM configurations.

The ideal configuration for an unsupervised EvoArt system

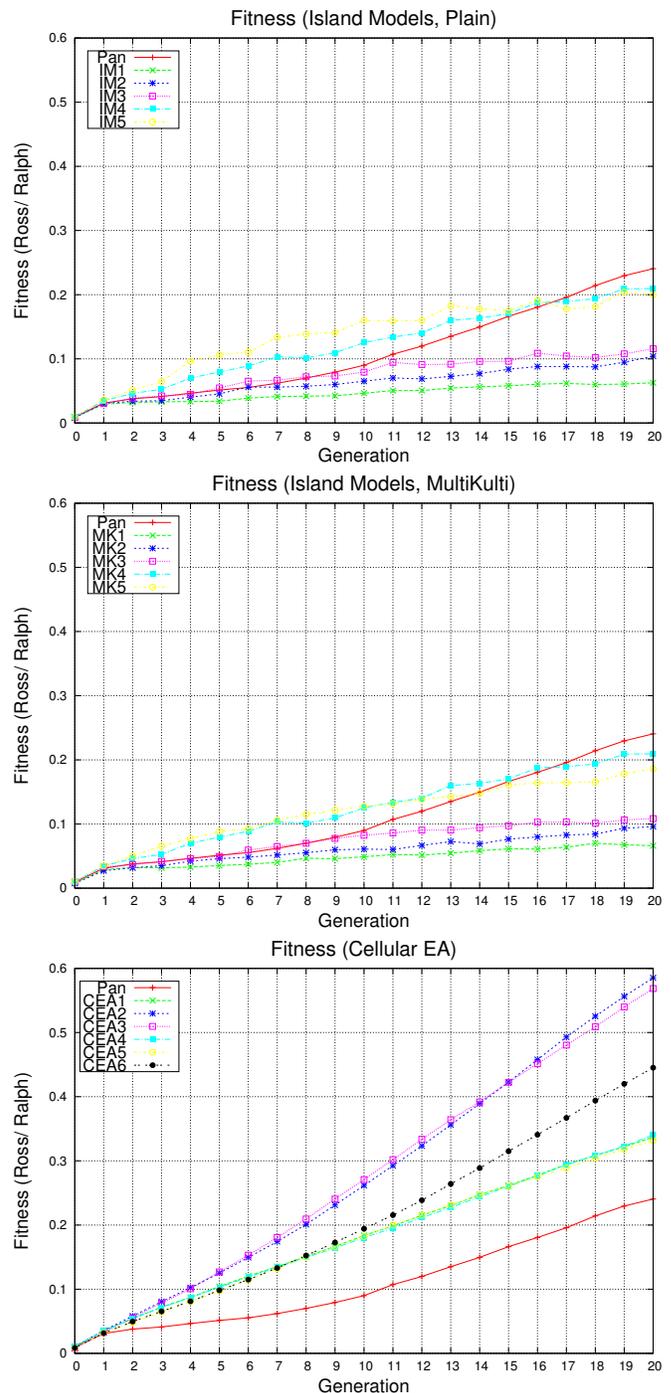


Figure 4. Fitness progression (all numbers are averaged over 30 runs). ‘Pan’ - Panmictic, ‘IM’ - Island Models, ‘MK’ - Island Models with the Multikulti algorithm, ‘CEA’ - Cellular EA.

would score high on both fitness and phenotype diversity, but the results from Figure 3 and 4 suggest that some configurations score high on progress in fitness (exploitation) and some configurations score high on phenotype diversity (exploration). In Figure 5 we present the normalised scores on fitness and phenotype diversity in the last (20th) generation for each configuration. All values are averages over 30 runs.

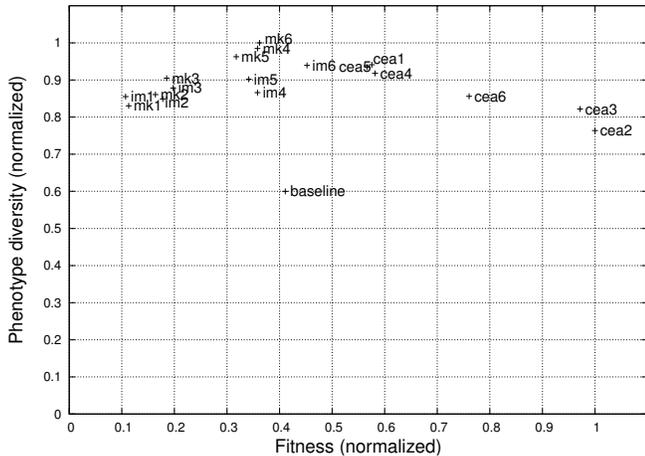


Figure 5. End scores of all configurations; each point represents the fitness and phenotype in the last generation (average over 30 runs), normalized between 0 and 1

A. Comparison with previous work

In our introduction we mentioned that this paper is a second in a series on population diversity in EvoArt. In previous work we developed custom genetic operators that perform a local search to find diverse new individuals through initialisation, crossover and mutation. We performed an experiment with a panmictic model using the augmented genetic operators from [15] and used all settings from Table III. There is one significant difference between this ‘Pan2’ configuration and the ‘Pan’ configuration; since the ‘Pan2’ configuration performs a local search step upon initialisation, crossover and mutation, the number of evaluations in the ‘Pan2’ configuration is much higher than in the ‘Pan’ configuration (even if the population size and number of generations are the same). The run times for ‘Pan2’ are therefore much higher than for ‘Pan’. We have included the comparison between our standard panmictic EA (Pan) with the augmented configuration (Pan2) for completeness, but since the Pan2 configuration uses more evaluations, we found that the comparison with the structured populations was not ‘fair’, and therefore we chose to present this comparison separately.

The use of the custom genetic operators with a local search results in a high phenotype diversity (Figure 6); the phenotype diversity actually increases with the generations, but the progress in fitness is very poor.

VI. CONCLUSIONS AND DISCUSSION

Our primary goal of this paper was to investigate whether we could maintain (or even increase) population diversity by using either IM (with or without Multikulti) or CEA. From our experiments we can conclude that the use of structured populations (either IM, IM with Multikulti, or CEA) all maintain a higher phenotype diversity than our standard Panmictic model. All structured population models used in our experiments scored higher on both genotype diversity and phenotype diversity than our standard Panmictic model. Next, we wanted to investigate which additional EA parameters (like migration size, island size in IM, and neighbourhood, width/height ratio in CEA) have a high influence on the phenotype.

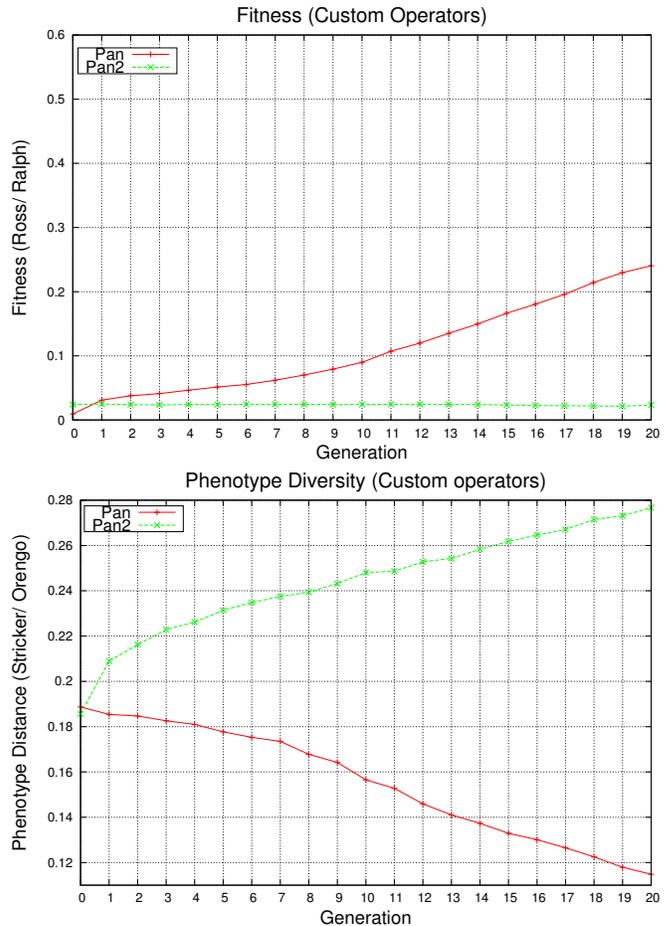


Figure 6. Fitness progression and genotype and phenotype diversity (all numbers are averaged over 30 runs). ‘Pan’- Panmictic, ‘Pan2’ - Panmictic with custom diversity operators

The result of the IM experiments suggest that having many (small) islands lead to increased phenotype diversity. We think that a small migration interval (we used a migration interval of 3 in our configuration MK4 and MK5) is needed when having many small islands; having a larger migration interval would lead to a decrease of diversity on the individual islands. There is clearly a dependency between island size, number of islands, migration interval, and migration size which requires further investigation.

With regard to the CEA parameters; the use of a thin grid population layout (where the width of the population is higher than the height) both led to higher phenotype diversity and genotype diversity in CEA, and slower fitness progression. The use of these population layouts clearly lead to lower selection pressure in CEA, and this result confirms work by Alba et al [2].

Our third research question concerns the tradeoff between exploitation and exploration; does an increase in population diversity (always) lead to a slower search (i.e. a slower increase in fitness). The tradeoff between exploitation and exploration is clearly visible in our CEA experiments; the three configurations that show the steepest increase in fitness also show the steepest decrease in phenotype diversity. The tradeoff is even more visible in our Panmictic model with

custom operators ('Pan2'); this configuration scored highest on progress of phenotype diversity, but scored worst on progress in fitness.

In our last research question we asked how our panmictic EA with custom genetic operators would compare to our configurations with structured populations. From our results we can conclude that the use of our custom genetic operators perform better than all structured populations on maintaining both phenotype and genotype diversity. Our 'Pan2' configuration was the only configuration that showed an increase in phenotype diversity. However; we have to emphasise that the panmictic EA with custom genetic operators performs a local search upon every mutation, crossover and initialisation step. This means that our 'Pan2' performs more evaluations in our typical run of 20 generations with a population of 256 individuals. The run times for our 'Pan2' configurations were longest of all configurations.

Another interesting finding is that in a number of experiments (most notably in the CEA configurations) an increase in genotype diversity coincided with a decrease in phenotype diversity, and vice versa. Similar findings have been reported by other authors within the GP field, most notably by Burke et al [9], [10] and by Tomassini et al [24]. We think that this is caused by the fact that a given image (phenotype) may have multiple, different genotypes; this is possible due to the introduction of bloat and introns during the evolutionary process.

There are several possible paths for future work. First, we would like to investigate the role of the mutation rate on population diversity; we intend to perform a series of experiments in which we increase the mutation rate in steps (in a panmictic EA) and measure its effect on phenotype and genotype diversity (and of course, fitness). Next, we would like to explore the use of heterogenous islands, in which different islands use different fitness functions. One can think of exchanging migrants only between islands that use different fitness functions. In this setup, migrants will move to an island where it is likely that they will perform poor on the 'new' fitness function, so additional mechanisms must be implemented to prevent an ineffective migration policy. One can think of a credit system, whereby new migrants receive credits that remain valid for a number of generations, or a niching mechanism, in which migrants (and their offspring) stay in a separate niche for a number of generations.

We would also like to use improved distance functions for measuring phenotype diversity. There exist several image distance functions in literature, most notably from the field of content-based image retrieval. We think that our image distance function based on the Stricker & Orengo function provides a good trade-off between computational complexity (it is a very fast and efficient measure) and accuracy, but intend to explore more elaborate image distance functions in future work.

ACKNOWLEDGEMENTS

We thank SURFsara (www.surfsara.nl) for the support in using the Lisa Compute Cluster.

REFERENCES

[1] Enrique Alba and Marco Tomassini. Parallelism and evolutionary algorithms. *Trans. Evol. Comp.*, 6(5):443–462, oct 2002.

[2] Enrique Alba and José M. Troya. Cellular evolutionary algorithms: Evaluating the influence of ratio. *PPSN VI*, pages 29–38, London, UK, 2000. Springer-Verlag.

[3] Lourdes Araujo and Juan Julián Merelo. Diversity through multiculturalism : Assessing migrant choice policies in an island model. *IEEE Transactions on Evolutionary Computation*, 15(4):456–469, 2011.

[4] Richard K. Belew and Lashon B. Booker, editors. *Proceedings of the Fourth International Conference on Genetic Algorithms*. Morgan Kaufmann, 1991.

[5] Steven Bergen and Brian J. Ross. Evolutionary art using summed multi-objective ranks. In Rick et al Riolo, editor, *Genetic Programming Theory and Practice VIII*, pages 227–244. Springer New York, 2011.

[6] Margaret Boden. *The Creative Mind*. Abacus, 1990.

[7] Margaret Boden. *Creativity and Art: Three Roads to Surprise*. Oxford University Press, 2010.

[8] Oliver Bown, Jon McCormack, and Taras Kowaliw. Ecosystemic methods for creative domains: Niche construction and boundary formation. In *IEEE Symposium on Artificial Life (ALIFE)*, pages 132–139. IEEE, 2011.

[9] Edmund Burke, Steven Gustafson, Graham Kendall, and Natalio Krasnogor. Advanced population diversity measures in genetic programming. In *PPSN VII, LNCS 2439*, pages 341–350. Springer Berlin / Heidelberg, 2002.

[10] Edmund K. Burke, Steven Gustafson, and Graham Kendall. Diversity in genetic programming: An analysis of measures and correlation with fitness. *IEEE Transactions on Evolutionary Computation*, 8(1):47–62, 2004.

[11] Erick Cantú-Paz. Topologies, migration rates, and multi-population parallel genetic algorithms. In *GECCO'99*, pages 91–98, 1999.

[12] Erick Cantú-Paz. Migration policies, selection pressure, and parallel evolutionary algorithms. *J. Heuristics*, pages 311–334, 2001.

[13] Robert J. Collins and David R. Jefferson. Selection in massively parallel genetic algorithms. In Belew and Booker [4], pages 249–256.

[14] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and T. Meyarivan. A fast elitist multi-objective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6:182–197, 2002.

[15] E. den Heijer and A. E. Eiben. Maintaining population diversity in evolutionary art. In *EvoMusart 2012, Evolutionary and Biologically Inspired Music, Sound, Art and Design, LNCS 7247*, pages 60–71, Malaga, Spain, 2012.

[16] Steve Dipaola and Liane Gabora. Incorporating characteristics of human creativity into an evolutionary art algorithm. *Genetic Programming and Evolvable Machines*, 10(2):97–110, 2009.

[17] Anikó Ekárt and S. Németh. A metric for genetic programs and fitness sharing. In *Genetic Programming*, volume 1802 of *Lecture Notes in Computer Science*, pages 259–270. Springer Berlin / Heidelberg, 2000.

[18] Bernard Manderick and Piet Spiessens. Fine-grained parallel genetic algorithms. In J. David Schaffer, editor, *ICGA*, pages 428–433. Morgan Kaufmann, 1989.

[19] Jon McCormack and Oliver Bown. Life's what you make: Niche construction and evolutionary art. In Mario Giacobini et al, editor, *Applications of Evolutionary Computing, EvoWorkshops 2009*, volume 5484 of *Lecture Notes in Computer Science*, pages 528–537. Springer, 2009.

[20] Brian Ross, William Ralph, and Hai Zong. Evolutionary image synthesis using a model of aesthetics. In *IEEE Congress on Evolutionary Computation (CEC) 2006*, pages 1087–1094, 2006.

[21] Zbigniew Skolicki and Kenneth De Jong. The influence of migration sizes and intervals on island models. *GECCO '05*, pages 1295–1302, New York, NY, USA, 2005. ACM.

[22] Markus Stricker and Markus Orengo. Similarity of color images. In *Storage and Retrieval of Image and Video Databases III, Vol. 2*, pages 381–392, 1995.

[23] Marco Tomassini. *Spatially Structured Evolutionary Algorithms: Artificial Evolution in Space and Time*. Springer, Berlin, Heidelberg, 2005.

[24] Marco Tomassini, Leonardo Vanneschi, Francisco Fernández, and Germán Galeano. A study of diversity in multipopulation genetic programming. In Pierre et al Liardet, editor, *Artificial Evolution, LNCS 2936*, pages 243–255. Springer, 2004.