Emojinating: Evolving Emoji Blends

João M. Cunha, Nuno Lourenço, João Correia,
Pedro Martins, and Penousal Machado

CISUC, Department of Informatics Engineering
University of Coimbra
{jmacunha,naml,jncor,pjmm,machado}@dei.uc.pt

Abstract. Graphic designers visually represent concepts in several of their daily tasks, such as in icon design. Computational systems can be of help in such tasks by stimulating creativity. However, current computational approaches to concept visual representation lack in effectiveness in promoting the exploration of the space of possible solutions. In this paper, we present an evolutionary approach that combines a standard Evolutionary Algorithm with a method inspired by Estimation of Distribution Algorithms to evolve emoji blends to represent user-introduced concepts. The quality of the developed approach is assessed using two separate user-studies. In comparison to previous approaches, our evolutionary system is able to better explore the search space, obtaining solutions of higher quality in terms of concept representativeness.

Keywords: Evolutionary Algorithm · Emoji · Interactive Evolutionary Computation · Visual Blending .

1 Introduction

In the domain of visual representations, computers have been made to draw on their own (e.g. [1]), used as creativity support tools for drawing (e.g. [2]) and even been given the role of colleague in co-creative systems (e.g. [3]). These examples, however, are more related to Art and shift away from the Design domain, in which a specific problem is addressed – e.g. how to design an icon to represent a given concept.

The difficulty behind developing computational approaches to solve Design problems is that, in most cases, they greatly depend on human perception. For this reason, they can be seen as open-ended as there is no optimal solution since they hinge on the user preferences. Thus, assessing quality is a complex problem on its own. One possible way to tackle this problem is to develop a system that allows the user to choose which solutions are adequate. One of such approaches in the Evolutionary Computation domain is usually referred to as Interactive Evolutionary Computation (IEC). IEC has been seen as suitable for such open-ended design problems [4], since it is capable of accumulating user preferences and, at the same time, stimulating creativity.

Regarding the visual representation of concepts, a multi-purpose system has great potential to be used as an ideation-aiding tool for brainstorming activities, presenting the user with representations for concepts. In [5], we have presented such a system. It uses a dataset of input visual representations (emoji), which are combined using a visual blending process to represent new concepts. Despite being able to achieve a higher conceptual coverage than the one from emoji system [6], the implemented system does not employ an effective strategy for exploring the search space – it only considers the best semantically matched emoji for blend generation. This approach ignores most of the search space and does not guarantee that the solutions are the most satisfactory for the user – one of the shortcomings identified in [5].

In this paper, we tackle the aforementioned issues by proposing a IEC framework that combines a standard Evolutionary Algorithm with a method inspired by Estimation of Distribution Algorithms to evolve visual representations of user-introduced concepts. We conduct two user-studies to compare our results with the ones reported in [5, 6].

2 Related Work

In this paper, we present a visual blending system for concept representation that uses an interactive evolutionary approach. As such, our work addresses two different topics: Visual Representation of concepts and Interactive Evolutionary Computation. In this section, we will present related work for both topics.

2.1 Visual Representation of Concepts

There are different approaches to the visual representation of concepts. One of the approaches consists in gathering a set of individual graphic elements (either pictures or icons), which work as a translation when put side by side – e.g. translating plot verbs into sequences of emoji [7] or the *Emojisaurus* platform[1].

Other approaches are based on input visual representations, which are used to produce new ones. The co-creative system Drawing Apprentice [3], for example, uses convolutional neural networks to perform real-time object recognition on the user sketch and responds with drawings of related objects. Ha and Eck [8] use sketches drawn on *Quick, Draw!*[2] to train a recurrent neural network capable of generalising concepts in order to draw them. The system is able to interpolate between several concepts (e.g. a pig, a rabbit, a crab and a face), which can be used for representing new concepts through visual blending – e.g. [9].

Visual blending consists in merging two or more visual representations to produce new ones. Approaches to visual blending can be divided into two groups based on the type of rendering used: photorealistic or non-photorealistic. One example from the photorealistic group is the system Vismantic [10], which uses three binary image operations (juxtaposition, replacement and fusion) to produce visual compositions for specific meanings (e.g. fusion of an image of an

[1] emojisaurus.com, retrieved 2019
[2] quickdraw.withgoogle.com, retrieved 2019

electric light bulb with an image of green leaves to represent *Electricity is green*). Another photorealistic example is the generation of new faces from existing ones, by combining face parts [11]. Non-photorealistic examples are the generation of visual representations for *boat-house* [12] or for the blends between the concepts *pig/angel/cactus* [13], which combine input visual representations to represent the new concepts. While these explorations only address a reduced number of concepts, the system presented in [5] – upon which this paper builds – works on a bigger scale by combining Semantic Network exploration with visual blending to automatically represent user-introduced concepts using emoji.

2.2 Interactive Evolutionary Computation

Evolutionary Algorithms (EAs) are computational models inspired by the Theory of Natural Selection. They are normally used in problems in which it is possible to assess the quality of solutions based on a specific goal. However, for problems in which quality is highly subjective and dependent on human perception, approaches that involve the user in the evolutionary process are seen as better suited [4]. Such approaches are often referred to as Interactive Evolutionary Computation (IEC) and are characterised by having a user-centred evaluation process. IEC has been used in different domains such as: Fashion, to produce shoe designs according to user taste [14]; Poster Design, to evolve typographic posters [15]; or even Information Visualisation, to explore the aesthetic domain [16]. In terms of symbol generation and visual representation of concepts, IEC has also been seen as a possible approach to solve problems.

Dorris et al. [17] used IEC to evolve anthropomorphic symbols that represented different emotions (e.g. anger, joy, etc.). The genotype of each individual was a vector of nine real-valued numbers that corresponded to the angles of the nine limbs (e.g. torso, left shoulder, right elbow, etc.). Dozier et al. [18] focused on emoticon design using an interactive distributed evolutionary algorithm – multiple processors working in parallel. It allowed several participants to interact in simultaneously, evolving solutions that are the result of their judgements. The emoticons were represented as a vector of 11 integer variables, which corresponded to the y-coordinates of 11 points (e.g. the first three codified the left eyebrow). Piper [19] also used a distributed approach, proposing an interactive genetic algorithm technique for designing safety warning symbols (e.g. *Hot Exhaust*). It used previously drawn symbol components as input which were then combined to produce new symbols. According to Piper [19], this distributed approach allowed the replacement of the usual focus group in symbol design process with a group of participants interacting using computers in a network. Hiroyasu et al. [20] proposed an interactive genetic algorithm that uses a crossover method based on probabilistic model-building for symbol evolution according to user preference. Each individual (symbol) was a combination of a color (HSB system) and a shape (from a set of eight different shapes). Estimation of distribution algorithms (EDAs) are based on the idea that statistical information about the search space can be extracted and used to modify the probability model, reducing the search space and leading faster to good solutions [21, 22].

Our approach takes inspiration from EDA methods to provide a way to quickly and efficiently search for solutions that match the user preferences.

3 Background

Emoji are pictorial symbols that are well integrated in the written language, which is observed in the growing number of emoji-related tools and features – e.g. search-by-emoji supported by Google[3], and the Emoji Replacement and Prediction features implemented in iOS 10[4].

Due to their large conceptual coverage (currently there are 2823 emoji in the released Emoji 11.0) and associated semantic knowledge, they are suitable to be used in computational approaches to the visual representation of concepts. In [5], we presented a system that uses a combinatorial approach to represent concepts using emojis as visual representations. It integrates data from three online open resources: Twitter's Twemoji 2.3: a dataset of emoji fully scalable vector graphics, in which each emoji is composed of several layers; EmojiNet, a machine readable sense inventory for emoji built through the aggregation of emoji explanations from multiple sources [23]; and ConceptNet: a semantic network originated from the project Open Mind Common Sense [24].

The system has three components: (i) the *Concept Extender* (CE) searches related concepts to a given concept using ConceptNet; (ii) the *Emoji Searcher* (ES) searches for existing emoji semantically related to a given word; and (iii) the *Emoji Blender* (EB) produces new emoji using visual blending by adding / replacing layers. For a more detailed description, we refer the reader to [5].

In [6], we assessed the system's performance using 1509 nouns from the New General Service List (NGSL) [25] and reported that the system was able to produce emoji for 75% of the list. Despite considering these results as good, the tested system only uses the best semantically matched emoji for each concept, not exploring the full search space. With this in mind, we propose an evolutionary approach to explore the search space and find visual representations of concepts that match user preference.

4 Our approach

In the context of this project, we use the *Emoji Searcher* (ES) and the *Concept Extender* (CE) components presented in [5], and we introduce an novel approach to explore the search space using IEC methodologies. As such, an evolutionary system was implemented. The *Emoji Blender* (EB) component was modified in order to work together with the evolutionary engine, in which the generated blends are the phenotype of individuals.

[3] forbes.com/sites/jaysondemers/2017/06/01/could-emoji-searches-and-emoji-seo-become-a-trend/, retrieved 2018.

[4] macrumors.com/how-to/ios-10-messages-emoji/, retrieved 2018.

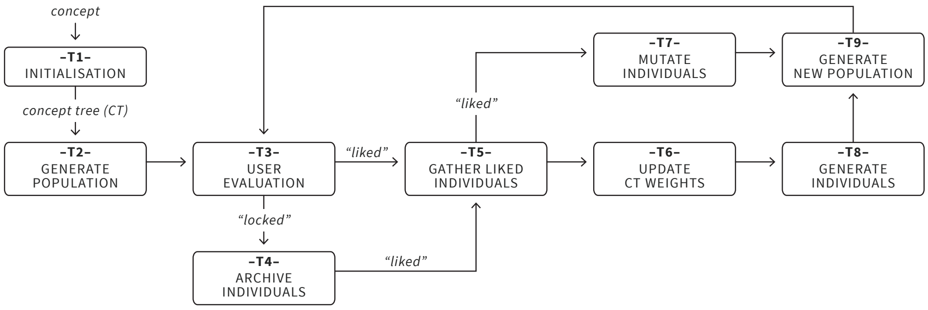

Fig. 1. Evolutionary framework diagram, showing tasks (T1-9) and objects, e.g. *concept tree* (CT). The user starts by introducing a concept, which will be used to generate a random set of solutions (T1 and T2). Then, the user evaluates the individuals by selecting the ones that fit his/her preferences (T3), referred to as "liked individuals". The user can also select the individuals to be stored in an archive (T4), referred to as "locked individuals". After the evaluation, the fittest individuals ("liked") are gathered from the population and from the archive (T5). The gathered individuals are then used to produce offspring through mutation (T7), and to update the weights of the Concept Tree (T6) – a graph-structured object with which new individuals are generated (T8). This process can be repeated indefinitely until the user is satisfied.

The approach presented in this paper has a two-level evolution: on a macro level, it uses a method that takes inspiration from EDAs to direct the search to areas that match the user preference; on a micro and more specific level, it uses a standard EA to focus the evolution on certain individuals. The approach is schematically represented in Fig. 1.

4.1 Concept Tree and General Evolution

The ES and CE components are used together to produce a graph-like structured object from the user introduced concept (T1 in Fig. 1). This object – which we will refer to as Concept Tree (CT) – stores the conceptual and emoji data produced from the analysis of the concept (see Fig. 2). It has two different levels: concept level, in which related concepts are connected (e.g. the concept *god* is connected with the related concepts *judge people*, *judge men*, *justify hate* and *quiet storm*); and the emoji level, which stores the sets of emoji retrieved for each concept (e.g. *judge* has a set of emoji and *men* has another, see Fig. 2).

The complexity of the CT object depends on the type of concept introduced by the user. If it is a single-word concept (e.g., *god*), related double-word concepts are searched and then corresponding emoji are retrieved for each of the words. If the user introduces a double-word concept, no related concept is required and the system directly retrieves emoji.

Taking inspiration from EDA methods, a weight value is assigned to every concept in the set of related concepts (in case they exist) and to each emoji. These weights are also stored in the CT object. When we generate new individuals

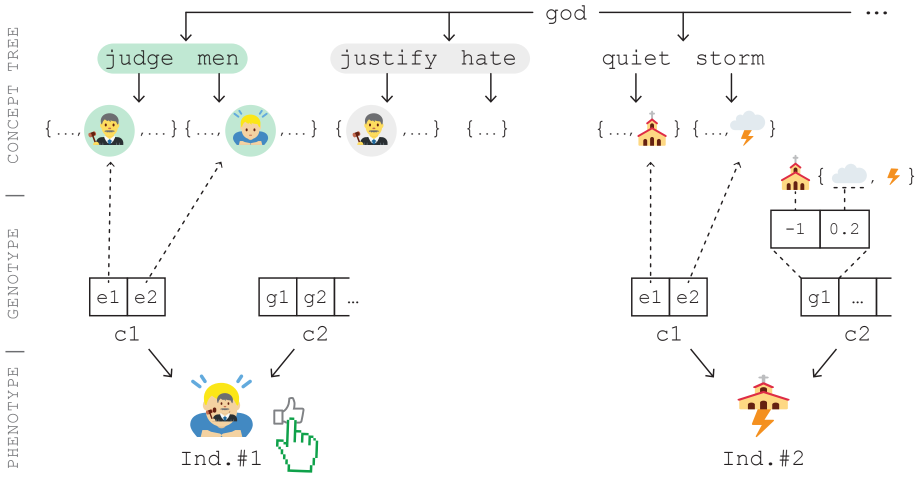

Fig. 2. Individual Representation and weight update system. The figure shows the two chromosomes (*c1* and *c2*) of two individuals' genotypes. It also shows gene 1 (*g1*) of *c2* from individual #2 in detail. Regarding the weight system, the individual #1 is being "liked", which directly increases the weights of the concepts / emoji marked in green and indirectly increases the ones of concepts / emoji marked in grey.

(T2 and T8), the weights are used to select both the concept and the two emoji for the new individual – the higher the weight, the greater chances it has of being selected. Initially the weights are all set to 1 and are updated in each generation according to user preferences (T6 in Fig. 1).

4.2 Representation

The emoji from the Twitter's Twemoji dataset are composed of layers – e.g. the *storm* emoji in Fig. 2 has two layers, a *cloud* and a *lightning*. As already mentioned, each visual blend is the phenotype of an individual (see Fig. 2). The individuals are encoded using a two chromosome genotype, which codify the combination between two emoji. The first chromosome (*c1* in Fig 2) stores the two emoji (*e1* and *e2*) used in the blend. The second chromosome (*c2* in Fig 2) is responsible for defining the exchanges of parts that occur in the blend. This is codified by having each exchange stored as a gene (e.g. *g1*). Each gene corresponds to a set of two values: the first defines the part from *e1* that will be used as replacement (-1 in Fig 2) and the second corresponds to the layer that will be replaced in *e2* (0.2 in Fig 2). As the number of layers is not the same among emoji, we use numbers in the [0,1] interval, which correspond to the position of the layer in the layer array. The value -1 can also be used, when the whole emoji is to be used instead of a layer (e.g. when a juxtaposition blend occurs). For example, for individual #2 in Fig 2 the whole *church* emoji is used as replacement (encoded by the "-1" of *g1*) and the *cloud* layer is defined as replaceable part (encoded by the "0.2" of *g1*).

In the current implementation, two types of blends are used: replacement (a part of *e1* is replaced by *e2*) and juxtaposition (*e1* and *e2* are put side by side or one over the other). In both cases, only one exchange is encoded per individual. In the future, we plan to implement a third blend type (fusion), which uses several exchanges of parts – already supported by the chosen representation.

4.3 User evaluation

In each generation, a population of 20 individuals is presented to the user, who is able to perform two different actions that affect the evolutionary process: mark individuals as "liked", which increases their fitness; and store them in the archive.

When an individual is "liked" (e.g. Ind.#1 in Fig.2), the weights of the CT are updated (T6). It directly increases the weight of the related concept behind the individual and of the used emoji in the sets belonging to the concept (marked in green in Fig.2). A process of indirect weight assignment is also used as the system searches for the used emoji in other concept's sets and also increases the weight of the emoji and corresponding concept (marked in grey in Fig.2). This fosters related concepts that also use the same emoji and allows the system to find solutions that might also be of interest to the user. The weight increment is calculated based on the sum of weights of a set and it varies according to the type – a direct increment is 5% of the weight sum and an indirect is 2%. In order to make the evolutionary system work, the user does not need to classify every single candidate solution but only select the ones considered interesting.

4.4 Weight equalisation

A method of weight equalisation was implemented, which means that, as the evolutionary process progresses, there is a tendency towards an equal distribution of weights. The weights between concepts and between emoji inside sets will eventually converge to the same value, if not stimulated. This allows the system to achieve diversity even in late generations, avoiding unwanted convergence.

First, the average of weights inside a set is calculated. The weights that are above average are updated according to the following function ($\text{EQ\_RATE} = \frac{1}{3}$):

$$new\_weight = (current\_weight - average\_weight) \times \text{EQ\_RATE} \tag{1}$$

The method of weight equalisation is particularly useful when used together with an archive, which allows the user to increase population diversity and explore other areas of the search space, without losing individuals deemed as good. Despite being different from classic EAs (in which convergence is a goal), this approach fits the problem, as the goal is to help the user find the highest number of interesting, yet considerably different, solutions.

4.5 Archive

An archive is often used to avoid the loss of population diversity by storing good individuals that can later be reintroduced in the population – e.g. [14, 26]. Another possible use is to store the fittest individuals in order to use them to automatically guide the search towards unseen solutions – e.g. [27].

In our case, diversification of the population is achieved with weight equalisation (as already described). Our archive works as a storage of individuals and has two main functionalities: (i) to save individuals and to avoid losing them in the evolutionary process; (ii) allowing the user to activate a permanent "liked" status that leads to a constant fostering of individuals. This option helps in situations that the user has found an individual with an interesting trait (e.g. the use of a specific emoji) and wants to constantly stimulate it without having to manually do it on each generation. As explained before, selecting an individual as "liked" not only fosters its specific evolution but also has an effect on general evolution – changing CT weights and consequently affecting the generation of new individuals.

Moreover, storing the individual in the archive is the only way of guaranteeing that it is not lost when moving to the next generation. It allows the user to store good solutions and focus on other possibilities, while being able, at any time, to further evolve the stored individual, by activating the "liked" option. Combined with the weight equalisation, this makes it possible for the system to increase its diversity and, at the same time, avoid the loss of good individuals. This strategy allows the user to continuously change its exploration goal and try to find new promising areas in the search space.

4.6 Mutation

In addition to being used to update the CT weights (T6), user-evaluated individuals (the "liked" ones) are also employed in the production of offspring in each generation (T7). These are gathered from both the current population and the archive, being the individuals marked with "like". From each "liked" individual, a set of 4 new individuals are produced (e.g. in Fig. 3, the 4 "bread-rhinos" in the population were generated by mutating the "liked" one in the archive). The parent individual goes through a mutation process, in which three types of mutation may occur: (i) emoji mutation (20% probability of occurring) – the emoji used as replacement is changed; (ii) layer mutation (80% probability of occurring per gene) – the replaced layer is changed (e.g. all "bread-rhinos" in the population except the first); and (iii) blend type mutation (5% probability) – this mutation changes the type of blend to juxtaposition, in which the emoji are used together and no replacement occurs (e.g. the first "bread-rhino" in the population). If a blend type mutation happens, no layer mutation occurs. The values presented were empirically obtained through experimentation and adjustments.

The use of the layer and emoji mutation types covers two situations: (i) adequate emoji are being used but the layer is not the correct; (ii) the exchange of layers is considered good but using different emoji may lead to a better solution.

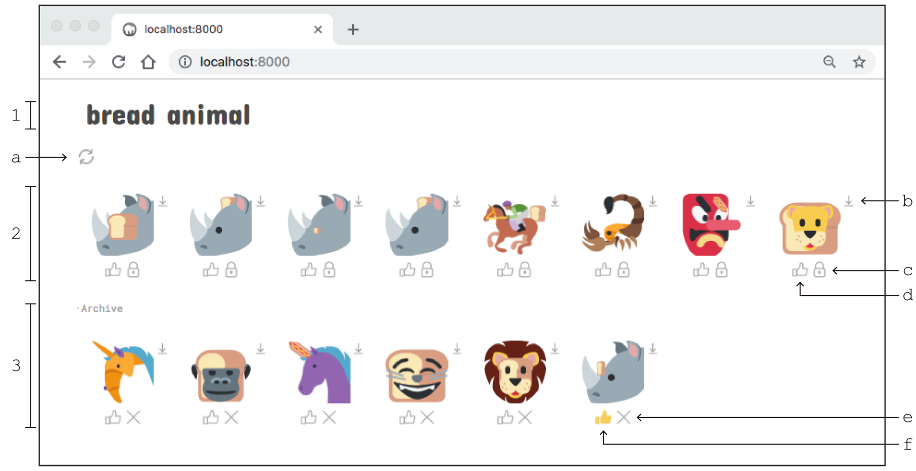

Fig. 3. The interface is divided into 3 areas: *search area* (1), *population area* (2) and *archive area* (3). There are 5 different button types that allow the user to interact with the system: *next generation* (a), *download* (b), *lock* (c), *like* (d) and *remove from archive* (e). A "liked" individual has an activated *like* button (f). The number of individuals in the population was intentionally reduced to increase the legibility of the figure.

4.7 Offspring

The offspring produced from parent individuals (T7) are added to a pool, from which they are afterwards randomly selected for the next generation. The number of individuals in the population is constant (20). As such, there is a maximum percentage of the new population (30%) that is used for individuals generated from parents through mutation. The remaining percentage corresponds to new individuals generated from the CT (T8). When generating individuals from scratch using CT, the probability of juxtaposition is set to 20% and of replacement to the remaining 80% – replacement can lead to many more different solutions than juxtaposition and, as such, it should occur more frequently.

4.8 Interface

The IEC system was implemented as a web-based application, which allows user interaction (see Fig. 3). The interface has three areas: the *search area*, the *population area* and the *archive area* (1–3 in Fig. 3). The *search area* is where the user introduces the concept (e.g. *bread animal* in Fig. 3).

The *population area* presents the current population, showing the visual representation of the blends. Each individual has buttons: the "like", which is used to evaluate the individual (d); the "lock", which stores the individual in the archive (c); and one to download the visual representation of the individual (b).

Individuals in the *archive area* also have a "like" button, which is used to active/deactivate the evaluation of the individual (the choice is maintained between generations), and a button to remove it from the archive (e in Fig. 3).

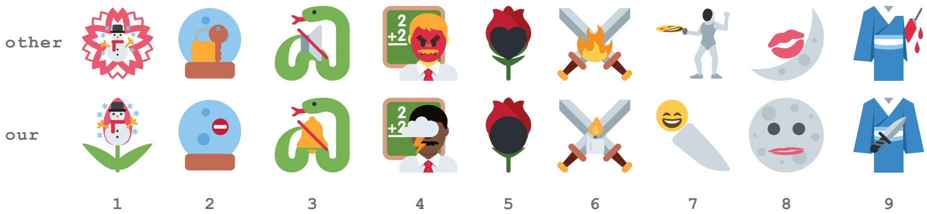

Fig. 4. Blends used in user-survey #1 for the concepts (1-9) *frozen flower*, *secrets in the future*, *silent snake*, *storm of the teacher*, *the darkest rose*, *the flame of swords*, *the laughing blade*, *the sexy moon* and *the sharp silk*. Blends in the top row are from [5] and the ones in the bottom row are obtained with our system.

Table 1. User-study #1 results expressed in percentage for each concept.

		answers (%)			
#	**concept**	our image	"equally good"	image by [5]	"none"
1	*frozen flower*	54.8	12.9	16.1	16.1
2	*secrets in the future*	9.7	0	58.1	32.3
3	*silent snake*	12.9	22.6	61.3	3.2
4	*storm of the teacher*	22.6	9.7	58.1	9.7
5	*the darkest rose*	9.7	16.1	16.1	58.1
6	*the flame of swords*	0	6.5	90.3	3.2
7	*the laughing blade*	45.2	12.9	16.1	25.8
8	*the sexy moon*	19.4	0	64.5	16.1
9	*the sharp silk*	32.3	3.2	3.2	61.3

5 Results and discussion

In this section, we present and discuss the experimental results obtained from two user-studies. User-study #1 compares the results from our approach with the ones presented in [5], for a set of concepts. User-study #2 assesses the efficiency of the system in the production of visual representations for single-word concepts and compares the results with the ones described in [6]. In the last part of this section, we make a more general analysis of the approach.

5.1 User-study #1: comparing images from two approaches

The first study was used to assess if our approach could lead to better solutions than a non-evolutionary deterministic version of the system, as presented in [5].

In [5], the system was used by 22 participants to generate visual representations for a set of ten concepts and the best solutions were collected. We used our system to produce solutions for the same concepts (see Table 1) and conducted a survey with 31 participants to assess if our system produced better solutions.

We used a multiple choice survey, with one question for each concept, following the model *Which of the following images represents better: [insert concept]?* Each question had four randomly ordered answer options: image produced by our system, image from [5], "equally good" and "none". In order to produce the survey, two people used the system together to select a good representation for each concept (see Fig. 4). Despite the risk of introducing bias towards our preferences, this was a necessary step to reduce the number of options presented to the user. One of the concepts (*serpent of the year*) was not used because the users could not find any good solution different from the one presented in [5].

The survey was conducted with 31 participants, with age between 18-32. The results are shown in Table 1. We can see that for two of the concepts (*frozen flower* and *the laughing blade*) our image was selected as better by the majority of the participants and for *the darkest rose*, 25.8% selected it as better or "equally good". Moreover, for *the sharp silk*, despite the majority of the participants selecting the option "none" (consistent with previous results [5]), our image still had better results than the image presented in [5], which was only selected by 1 participant. All in all, our approach was competitive in 4 out of the 10 concepts.

5.2 User-study #2: testing with concepts from NGSL

In user-study #2, we compare the results from our approach with the ones described in [6], in which a set of 1509 nouns from the New General Service List (NGSL) [25] was used to assess the quality of the system presented in [5]. In the study described in [6], the system allowed the user to introduce a concept and presented an *existing emoji*, *related emoji* (gathered from semantic search to related concepts) and *blends*. It was assessed if each noun was represented by its (i) *existing emoji* and by its (ii) *related emoji* or *blends*. Based on the results presented in [6], we divided the noun list into four groups:

- *group 0:* the system was not able to produce blends. This group was excluded as it could not be used due to the lack of blends to evolve;
- *group 1:* system produces blends but neither the related emoji/blends nor *existing emoji* were reported to represent the concept;
- *group 2:* system produces blends and only the related emoji/blends were reported to represent the concept;
- *group 3:* system produces blends and the existing emoji were reported to represent the concept (the related emoji/blends may also represent).

Moreover, we crossed the list with a dataset of concreteness ratings [28], obtaining a value of concreteness for each noun – from 1 (abstract, language based) to 5 (concrete, experience based). Despite not being a goal in this paper, we divided each noun group in three subgroups to assess if there is any relation between concreteness and representation easiness: (A) low concreteness, (B) medium concreteness and (C) high concreteness.

We conducted a survey with eight participants in which each participant used the system to generate visual representations for 9 randomly selected concepts

Table 2. User-study #2 results for *quality*, *number of solutions*, *number of generations* and three combinations of quality (Q) / exported (E) /generations (G) that correspond to "early quit without results", "early quit with poor results" and "early satisfaction" (expressed in number of nouns and divided by noun group).

	quality			**# exported**			**# generations**				Q≤3	Q≥4
	1	>1 & ≤3	≥4	0	1	>1	<15	≥15 & <30	30	E=0 & G<20	& E>0 & G<20	& E>0 & G<20
group 1	8	6	10	6	8	10	10	7	7	3	4	6
group 2	9	5	10	6	11	7	8	10	6	4	2	7
group 3	6	3	15	4	12	8	13	10	1	4	3	10

(one from each subgroup). As the goal for this survey was to achieve maximum coverage of each subgroup, we decided to avoid noun repetition. Despite this, in low concreteness subgroups only few nouns existed – subgroup 1A had 4 nouns, 2A had 3 and 3A had 5 – which led to the repetition of nouns among participants for those subgroups. The participants used the system to evolve visual representations for the nouns, conducting only one run per noun and having a limit of 30 generations. They were asked to find individuals that represented the introduced noun and were allowed to stop the run before reaching 30 generations if they were already satisfied or if the system was not being able to further improve. For each noun, they were also requested to evaluate how well it was represented by the system, from 1 (very bad) to 5 (very good), and exported the solutions that they considered the best among the ones that represented the noun (see Fig. 5).

The results in Table 2 show that, in terms of quality, our system is able to produce solutions with quality equal or above "good" for almost half of the concepts in group 1 and 2 (10 out of 24), and for the majority of concepts in group 3 (15 out of 24). This is particularly important in group 1, for which the previous study [6] was not able to find any satisfactory solution. Moreover, the participants were able to find more than one concept-representative solution in 34% of the runs (25 out of 72), e.g. *invitation* in Fig. 5.

We were able to compare the individuals selected as the best by each participant for each concept with the solutions obtained by the system from [6] for the same concepts. In 38 out of 72 runs, the solution considered as the best was not produced by the approach from [6]. In addition, in 30 cases out of the 38 our solution was considered better than any of the solutions obtained with the system from [6] and in 5 was considered equally good. This shows that the evolutionary approach has clear advantages in comparison to the approach presented in [5, 6].

Concerning the number of generations, in 80% of the runs (58 out of 72) the participants stopped before reaching the generation limit, which can be indicative of two things: the system could not create blends that represented the concept or the user was already satisfied. To further analyse this matter, we used three combinations of quality/exported/generations that correspond to "early quit without results", "early quit with poor results" and "early satisfaction" (see Table 2). From the results we can see that in 11 runs, the participant

Fig. 5. Examples of blends selected by the participants as good solutions

stopped without any exported solution before reaching 20 generations, which indicates that the system was not being successful. In addition, the column corresponding to "early quit with poor results" shows that in 9 runs the participant considered that the system would not get any better. On the other hand, in 30% of the runs (23 out of 72) the participant was satisfied before reaching the 20th generation, which means that the system was able to quickly evolve solutions that pleased the user.

One of the problems in IEC approaches is the weariness of the user [14]. In the end of the survey, the participants evaluated the weariness degree of the task from 1 (very low) to 5 (very high) and 50% of participants rated it as very low in weariness and the other 50% as low. We also asked the participants to evaluate the surprise degree from 1 (very low) to 5 (very high) – 25% rated it as 3 and 75% as 4. This shows that the system is able to generate solutions that are unexpected. However, as results are highly dependent on the concept, in a proper evaluation the surprise degree should be assessed for each concept.

When analysing the results, we could not observe any obvious relation between concept concreteness and easiness of representation. Our initial expectation was that concrete concepts would be easier to represent. The fact that we could not observe such correlation may indicate that using emoji blending to represent concrete concepts (e.g. *brain*) might not be the best approach. Moreover, some of the participants commented that they were trying to isolate an emoji, which is observed in some of the selected solutions – they tend to mostly show only one of the emoji (see blends for the concepts *anything* and *aircraft* in Fig. 5). However, further research is required on this subject as our remarks are only speculative and not statistically proven.

Another subject concerns the methods used in blend production. For single word concepts, the system gathers related double-word concepts to use in the blending process. The emoji belonging to each of the related concepts are not transferable to other concepts unless they are also in the emoji list of the concept, i.e. two individuals produced from different related concepts, one produced using *emoji* A and B and the other produced using *emoji* C and D, may never lead to the generation of an individual from *emoji* A and D. This is the reason behind some of users complaining that they were not being successful in "combining" emoji from two individuals. This is not very intuitive when using an IEC approach and should be addressed in the future.

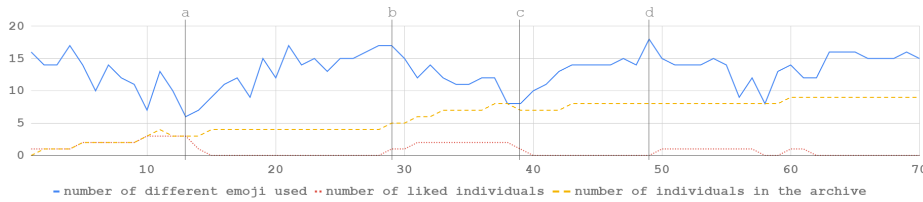

Fig. 6. Metrics progression along the generations of a run for the concept *cell fish* (best viewed in colour). A video of the run can be seen at https://rebrand.ly/evomusart19.

5.3 General analysis

The main goal behind a system for the visual representation of concepts is being able to produce at least one good solution. Our evolutionary system allows the user to explore different and unrelated areas of interest in the search space, often leading to several distinct solutions of good quality for the same concept.

To give an example of how the system reacts to user interaction, we show the progression of several metrics during one run (Fig. 6). It can be observed that the number of different emoji tends to decrease when solutions are marked as "liked", which shows that the population evolves towards similar solutions (e.g. from *b* to *c* in Fig. 6). The opposite is also verified: when no individual is "liked", the variation of the population tends to increase (e.g. from *a* to *b* and from *c* to *d* in Fig. 6). The increase in number of individuals in the archive highlights its usefulness for search space exploration and reflects the capability of the system to evolve solutions that match user preferences.

In the cases in which the system was reported to not being able to generate anything that represented the concept, the reason was related to the gathering of semantic knowledge and had nothing to do with the evolutionary engine (the main focus of this paper). In general, the efficiency of the system is highly dependent on the existing semantic knowledge, emoji found and user perception.

6 Conclusion and future work

The visual representation of concepts plays a central role in the work of graphic designers, being crucial in tasks such as icon design. Computational systems can be used to aid the designer in ideation processes by stimulating creativity. In this paper, we proposed an evolutionary approach for the generation of visual representations of concepts that combines a standard Evolutionary Algorithm with a method inspired by EDAs. This approach allows the system to perform both a general evolution to direct the search to areas that match user preference and a focused evolution based on user-selected individuals. In order to do this, we used an archive to store individuals and selectively enable/disable their evolution.

We compared our approach with existing ones, by conducting two user-studies. The results show that our approach allows the exploration of more of the search space and is able to present the user with better solutions. Future enhancements to the proposed approach include: (i) taking into account the semantic value attributed to related concepts and to emoji by the ES and CE components in the initialisation of weights, which may increase the fitness of the population in the first generation; (ii) considering fusion blend type and (iii) a mutation operator for layer transformations (scale, translation and rotation); (iv) using other aspects as ground for blending (e.g. conceptual information related to image schemas [29]); and (v) implementing automatic fitness and (vi) long-term learning (i.e. considering knowledge from past runs).

Demo video: https://rebrand.ly/evomusart19

7 Acknowledgments

João M. Cunha is partially funded by Fundação para a Ciência e Tecnologia (FCT), Portugal, under the grant SFRH/BD/120905/2016; and is based upon work from COST Action CA15140: ImAppNIO, supported by COST (European Cooperation in Science and Technology): www.cost.eu. This work includes data from ConceptNet 5, which was compiled by the Commonsense Computing Initiative and is freely available under the Creative Commons Attribution-ShareAlike license (CC BY SA 4.0) from conceptnet.io.

References

1. McCorduck, P.: Aaron's code: meta-art, artificial intelligence, and the work of Harold Cohen. Macmillan (1991)
2. Lee, Y.J., Zitnick, C.L., Cohen, M.F.: Shadowdraw: real-time user guidance for freehand drawing. In: ACM Trans. on Graphics (TOG). vol. 30, p. 27. ACM (2011)
3. Davis, N., Hsiao, C.P., Singh, K.Y., Magerko, B.: Co-creative drawing agent with object recognition. In: AIIDE-16 (2016)
4. Parmee, I.C., Abraham, J.A., Machwe, A.: User-centric evolutionary computing: Melding human and machine capability to satisfy multiple criteria. In: Multiobjective Problem Solving from Nature, pp. 263–283. Springer (2008)
5. Cunha, J.M., Martins, P., Machado, P.: How shell and horn make a unicorn: Experimenting with visual blending in emoji. In: Proc. of the Ninth Int. Conf. on Computational Creativity (ICCC 2018) (2018)
6. Cunha, J.M., Martins, P., Machado, P.: Emojinating: Representing concepts using emoji. In: Workshop Proc. from ICCBR18 (2018)
7. Wicke, P.: Ideograms as semantic primes: Emoji in computational linguistic creativity (2017)
8. Ha, D., Eck, D.: A neural representation of sketch drawings. arXiv preprint arXiv:1704.03477 (2017)
9. Karimi, P., Maher, M.L., Grace, K., Davis, N.: A computational model for visual conceptual blends. IBM Journal of Research and Development (2018)
10. Xiao, P., Linkola, S.: Vismantic: Meaning-making with images. In: Proc. of the 6th Int. Conf. on Computational Creativity, ICCC-15 (2015)

11. Correia, J., Martins, T., Martins, P., Machado, P.: X-faces: The exploit is out there. In: Proceedings of the Seventh International Conference on Computational Creativity (2016)
12. Pereira, F.C., Cardoso, A.: The boat-house visual blending experience. In: Proc. of the Symposium for Creativity in Arts and Science of AISB 2002 (2002)
13. Cunha, J.M., Gonçalves, J., Martins, P., Machado, P., Cardoso, A.: A pig, an angel and a cactus walk into a blender: A descriptive approach to visual blending. In: Proc. of the Eighth Int. Conf. on Computational Creativity (2017)
14. Lourenço, N., Assunção, F., Maçãs, C., Machado, P.: Evofashion: Customising fashion through evolution. In: Int. Conf. on Evolutionary and Biologically Inspired Music and Art. Springer International Publishing (2017)
15. Rebelo, S., Fonseca, C.M.: Experiments in the development of typographical posters. In: 6th Conf. on Computation, Communication, Aesthetics and X) (2018)
16. Maçãs, C., Lourenço, N., Machado, P.: Interactive evolution of swarms for the visualisation of consumptions. In: ArtsIT 2018 (2018)
17. Dorris, N., Carnahan, B., Orsini, L., Kuntz, L.A.: Interactive evolutionary design of anthropomorphic symbols. In: Evolutionary Computation, 2004. CEC2004. Congress on. vol. 1, pp. 433–440. IEEE (2004)
18. Dozier, G., Carnahan, B., Seals, C., Kuntz, L.A., Fu, S.G.: An interactive distributed evolutionary algorithm (idea) for design. In: Systems, Man and Cybernetics, 2005 IEEE Int. Conf. on. vol. 1, pp. 418–422. IEEE (2005)
19. Piper, A.K.: Participatory design of warning symbols using distributed interactive evolutionary computation. Ph.D. thesis, Auburn University (2010)
20. Hiroyasu, T., Tanaka, M., Ito, F., Miki, M.: Discussion of a crossover method using a probabilistic model for interactive genetic algorithm. In: SCIS & ISIS SCIS & ISIS 2008. Japan Society for Fuzzy Theory and Intelligent Informatics (2008)
21. Gong, D., Yan, J., Zuo, G.: A review of gait optimization based on evolutionary computation. Applied Computational Intelligence and Soft Computing (2010)
22. Pelikan, M., Goldberg, D.E., Lobo, F.G.: A survey of optimization by building and using probabilistic models. Computational optimization and applications (2002)
23. Wijeratne, S., Balasuriya, L., Sheth, A., Doran, D.: Emojinet: An open service and api for emoji sense discovery. In: Proc. of ICWSM-17 (2017)
24. Speer, R., Havasi, C.: Representing general relational knowledge in conceptnet 5. In: LREC. pp. 3679–3686 (2012)
25. Browne, C.: A new general service list: The better mousetrap weve been looking for. Vocabulary Learning and Instruction 3(1), 1–10 (2014)
26. Liapis, A., Yannakakis, G.N., Togelius, J.: Sentient Sketchbook: Computer-aided game level authoring. In: FDG. pp. 213–220 (2013)
27. Vinhas, A., Assunção, F., Correia, J., Ekárt, A., Machado, P.: Fitness and novelty in evolutionary art. In: EvoMUSART. Lecture Notes in Computer Science, vol. 9596, pp. 225–240. Springer (2016)
28. Brysbaert, M., Warriner, A.B., Kuperman, V.: Concreteness ratings for 40 thousand generally known english word lemmas. Behavior research methods 46(3), 904–911 (2014)
29. Cunha, J.M., Martins, P., Machado, P.: Using image schemas in the visual representation of concepts. In: Proceedings of TriCoLore 2018. CEUR (2018)