MuSyFI - Music Synthesis From Images

André Carvalho dos Santos
andre.carvalho.dos.santos@tecnico.ulisboa.pt

Instituto Superior Técnico, Lisboa, Portugal

September 2020

Abstract

Creativity has accompanied humanity since the beginning of time. The ability to be creative has also allowed us to innovate and continue to innovate for centuries. Creativity is also inevitably linked to inspiration - the spark that initiates the whole creative process - which can arise from extrinsic or intrinsic means.

With this work, we aim at creating a program that tries to model this creative process in the field of Computational Creativity. For this purpose, we use images as our source of inspiration and implement a possible translation between visual and musical features. The output of our program is comprised of three different musical artifacts: an automatic version, a co-created version, and a genetic version.

The automatic version is built by extracting features from the image and mapping them into musical features, the co-created version is built by adding harmony lines manually composed by us to the automatic version, and finally the genetic version is built by applying a genetic algorithm to a mixed population of automatic and co-created versions.

The three versions were evaluated for six different images by conducting surveys. It was evaluated whether people considered our musical artifacts music, if they thought the artifacts had quality, if they considered the artifacts 'novel', if they liked the artifacts, and lastly if they were able to relate the artifacts with the image in which they were inspired.

There are still many improvements that can be made, either by following the same approach and building on top of it, or by following another approach. Still, from 300 respondents we can see that people considered that our musical artifacts were novel, and that they had quality, while also generally thinking that the genetic version was the best, justifying the genetic algorithm's implementation.

Keywords: Computational Creativity, Inspiration, Feature Translation, Genetic Algorithm, Music Generation

1. Introduction

Nowadays, everywhere we turn, we are surrounded by Artificial Intelligence (AI). However, we are still far from General Artificial Intelligence (GAI), which refers to the capability of machines to learn, understand and perform any intellectual task a human can. Creativity, though not exactly a task, is one of the many virtues the human brain still holds the upper hand over AI programs. That leads us to the field of Computational Creativity (CC). Being a relatively young field of study, many questions are still left unanswered, the main one perhaps being "How exactly can we simulate creativity using a computer?". In this work we approach this question through a very specific example: to fully or partially automatically generate music from seemingly unrelated information, in this case, images.

The purpose of this work is easy to state: take an image, any image, and generate music that can be related to it. This relation is subjective and so there is virtually an infinite number of musical artifacts that can be generated given the same image.

One of the possible ways to answer these questions was addressed by Teixeira and Pinto [10]. Taking this into account, one of the reasons that led to this work was to continue the research done in [10], while taking a different approach to it.

Another topic that motivated this thesis is that of inspiration. Inspiration, as creativity, is hard to understand and as such has yet to be defined formally. Nevertheless, most of us associate it as being an intrinsic part of the creative process. We wanted to try to understand better how inspiration influences the creative process.

We should point out that the goals for this work were the following: to contribute to the field of CC and understand if the musical artifacts created can be considered creative; to understand if the musical artifacts can be considered both music and aesthetically pleasing; to assess if images are relatable to the musical artifacts inspired by them or not. Each one of these goals is subjective, which

makes evaluation harder.

The rest of the paper is organized as follows. Section 2 contains a short review of related work and tries to put some of the remaining work into context. Section 3 discusses feature extraction from images and describes the main features extracted. Section 4 describes the visual to music mapping and goes into detail on how feature information is converted into the melody and harmony of our musical artifacts. Section 5, on the other hand, describes in detail the genetic algorithm developed in this work, its structure, the different possible mutations and crossovers, and how the fitness function is computed and used. In Section 6 we present our results and a discussion. Finally, Section 7 concludes the paper with both a critical summary as well as some indications for future work. Our musical artifacts and respective images can be seen and heard on our website: http://web.tecnico.ulisboa.pt/ist178488/.

2. Related Work

In order to try and emulate creativity, we first need to try and understand it. Only then can we study how to model creativity so we can try to implement a creative process, more specifically, a creative musical generation process.

2.1. Creativity

Margaret Boden's [1] definition of creativity is as follows: "creativity can be defined as the ability to generate novel, and valuable, ideas.". The term *valuable* is ambiguous, as there are many meanings to it, depending on the context ("interesting, useful, beautiful, simple, richly complex, and so on").

As for *novelty*, the author defends that there are two distinct types: Historical Novelty (H-Creative) and Psychological Novelty (P-Creative). The first refers to when something entirely new to mankind is created and it is very rare. The second refers to "everyday creativity", meaning something new to a person, be it the creator or the "audience".

We decided to abide by Boden's theory of creativity since we believe it to be simple and yet pragmatic.

2.2. Methods

In investigating which methods and techniques are usually used in music generation and recognizing that these methods are contained within the methods of CC, we found a comprehensive survey on algorithmic composition [4] where the authors then present us with a list of its main types of methods:

1. Grammars
2. Symbolic, Knowledge-Based Systems
3. Markov Chains
4. Artificial Neural-Networks
5. Evolutionary and Other Population-Based Methods
6. Self-Similarity and Cellular Automata

The first two are what the authors consider to be "classical 'good old-fashioned AI'". Numbers 3 and 4 are machine learning methods, and number 5 falls into the optimization methods category. The last one cannot be considered AI, but the authors chose to include them as well since they are an important algorithmic composition approach, nonetheless.

The authors sum up the different methods and categorize them according to their own taxonomy, condensing it all into one diagram of which we show a simplified version in Figure 1.

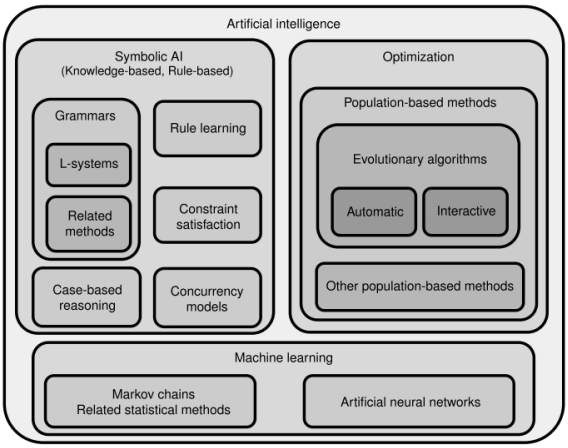


Figure 1: Taxonomy of the methods reviewed in [4]

Having reviewed all the methods here presented, we believe that deep learning techniques and Genetic Algorithms (GAs) are the most intriguing and worth further exploring, and so we decided to implement our own GA in our program. This will be described in more detail in Section 5.

3. Image Feature Extraction

To use images as an inspiration source, first we need to be able to extract features from them so we can then map these into musical features. In other words, we need to process the image. In the following subsections, we explain what image features we extracted, how we extracted them, and what further processing we did.

3.1. Saliencies

Saliencies are not formally defined in the field of computer vision, they are simply features that draw attention to us when looking at an image or a series of images. However, saliency detection is an active research subfield of computer vision, even if it lacks

a formal definition for saliencies. OpenCV [2] has a saliency module with two static saliency[1] detection algorithms. One of those algorithms is the StaticSaliencyFineGrained which was taken from [5].

Montabone and Soto in [5] based themselves on how humans focus their attention on particular areas of an image. The human eye retina has ganglion cells that respond to bright areas surrounded by a dark background (on-center ganglion cells), and to dark areas surrounded by a bright background (off-center ganglion cells). This center vs surround model can be computationally implemented by calculating what is called the center-surround differences. The input image is converted into greyscale and different surround values are used to generate different intensity maps. An integral image[2] is also used to preserve the image's original resolution, without making the calculations too computationally heavy. After calculating the center-surround differences for different surround values, the authors generate the saliency map by summing all of the different intensity maps.

The saliency map obtained from the StaticSaliencyFineGrained algorithm is shown in Figure 2b, next to the original image. As can be seen, either the dog or parts of it are identified as being salient, as well as some parts of the grass. However, the dog is not identified perfectly as a whole.

(a) Original image **(b)** Saliency map

Figure 2: Saliency map of an image, next to its original image

To improve upon this result, we resort to another image processing algorithm, the GrabCut algorithm, which is based on [8] and also implemented in the OpenCV library. It segments the image into foreground and background homogeneous regions using some pre-specified knowledge of where these regions should be. This segmentation is done by representing the image as a graph - nodes representing pixels and edges representing neighbourhood relationships between pixels - and using an optimization technique to cut the edges that are less expensive to cut. Edge weights are defined by an energy function, which is derived from the pre-specified information, and which defines if neighbouring pixels probably belong to the same region (more expensive edge) or if there is a boundary between them (less expensive edge).

Usually, the GrabCut algorithm receives an image and an associated bounding box with it to indicate where our region of interest is, or it needs a human to directly mark the background/foreground approximate regions in the image. It converts this information into a greyscale image with only four different shades of grey - categorized as Foreground, Probable Foreground, Probable Background and Background - which is then used as the algorithm's input. The output has the same format. By using the saliency maps, we can bypass the human input and still obtain accurate and autonomous results. If we classify each pixel in the saliency map as one of the four values previously mentioned, we can then feed that image to the algorithm and use its output to obtain the correct saliencies of the image.

We can observe this process in Figure 3. The first column has the saliency map previously shown, the second column has the GrabCut algorithm's input and the third column has its final output.

Figure 3: Saliency map and respective GrabCut input and output images

3.1.1 Contours

In image processing, a contour is a curve that joins all the continuous points along a boundary that encircle a region of pixels that have the same colour or intensity [6]. We used contours to study the shape of the saliencies we extracted.

OpenCV has a function called `findContours` that receives a binary image as input and finds its contours. It is based on [9] and it works by applying border following to the binary image, labeling the borders it finds. Here, border and contour are used interchangeably.

An example of this algorithm's results can be seen in Figure 4, next to the final saliency image. In green we have the contour or border itself, the blue dot is the centroid of the contour, the blue circle is the circle that encloses the saliency centered at the saliency's centroid, and the red dot in the border represents the point with minimum distance to the centroid along the border.

We then plot the distance to the centroid along the border, starting from the minimum distance

[1] Static saliencies are detected specifically in single images rather than in a sequence of images or video.

[2] In an integral image, each pixel's value is the sum of its intensity value plus every pixel above it and to its left.

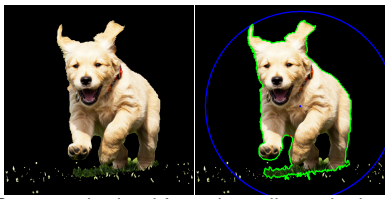


Figure 4: Contour obtained from the saliency in the dog image of our dataset

point - represented by the red dot - and continuing along the border, counterclockwise. The result of the dog image can be seen in Figure 5.

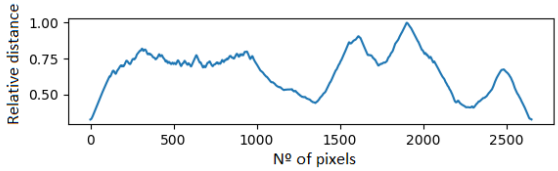


Figure 5: Plot of the dog image's contour distance

In Figure 5, the y-axis represents the distance to the centroid, which is relative to its maximum value, and the x-axis represents the number of pixels along the contour. This means each plot always has a peak of value 1 and that, as expected, bigger saliencies have bigger contours and thus bigger contour plots.

3.2. Colour

We use both the Hue, Saturation, Value (HSV) and Hue, Saturation, Lightness (HSL) models to extract what we call the dominant colours of an image, i.e., the colours that stand out. In practice, to obtain these dominant colours, we convert each Red, Green, Blue (RGB) image pixel into HSL coordinates and divide the 360 different hue values into 12 different hue bins, each having $360/12 = 30$ possible values. We then count the number of pixels that belong to each bin. If a hue bin has at least 10% of all the image's pixels, then it is considered a dominant hue tone. The 10% value was obtained empirically, and while it might seem low, it allows us to retain important colour information about the image that would otherwise be lost. For each hue bin, we calculate as well the average Saturation (from the HSV model), the average Lightness, and the average Value.

Having counted the number of pixels for every hue bin, we can plot their histogram. Furthermore, we can calculate these histograms for the whole image, only for each saliency, or for the non-salient image, i.e., the image that remains when we remove every saliency. In Figure 6 we show the histogram for the original dog image (Figure 2a). The y-axis represents the number of pixels and the x-axis represents the respective hue bin. Also, the dominant colours are all the hues whose bars are above the black horizontal line.

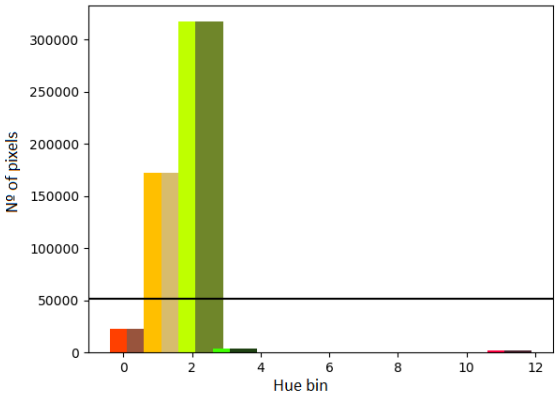


Figure 6: Colour histogram

We should note that, for each bin, we have two different bars: the first one represents the respective hue with max saturation and max lightness, and the second one represents the same hue but, with the average saturation and average lightness calculated. Also, we should note that the bins are numbered from $[0, 11]$.

3.3. Edges

Edge detection is a classical image processing problem. The goal of edge detection is to be able to identify points in an image where the brightness changes abruptly. The set of these points forms a set of curved lines that we call edges. John F. Canny developed in 1986 a staple algorithm for edge detection that was eventually named after him, the Canny edge detector [3].

The Canny edge detector is implemented as well in the OpenCV library in function `Canny`, so we just need to apply it to our image dataset. The high and low thresholds that were chosen were 30 and 200, respectively, and we used the same thresholds in every image in our dataset.

4. Feature Mapping

Having seen the visual features extracted and how exactly we were able to extract them, we now proceed to explain the visual to musical feature mapping we did: first by pointing out how we specified some overall musical features, and then by explaining how we did the melody and harmony parts for our musical artifacts.

4.1. General Song Features

The musical features we need to define before going into the melody and harmony parts of our musical artifacts are the time signature, the tempo, and the key or scale.

The most common time signature in music today is $\frac{4}{4}$, while other time signatures (like $\frac{6}{8}$ for example) are usually used to compose more complex

musical pieces, so we decided to define the time signature for all our musical artifacts as being $\frac{4}{4}$ as well.

Regarding the tempo of our musical artifacts, we chose to associate it with the number of edges in an image. Images with more edges seem in our opinion more frenetic and with a faster pace than an image that does not have as many edges.

We defined a minimum tempo of 60bpm and a maximum of 150bpm since they are relatively slow and fast tempos, respectively. We apply the Canny edge detector algorithm to the image to obtain its respective edge image, and we simply count the number of non-zero pixels in it and divide them by the total number of pixels in the image, obtaining what we call the edge ratio. That ratio is then divided by 0.3, since that was approximately the maximum edge ratio found in our dataset. Equation (1) defines how the tempo is assigned.

$$tempo = int\left(\frac{edge\_ratio}{0.3} \times (150 - 60) + 60\right) \quad (1)$$

Finally, regarding the key of the artifacts, we decided to generate tonal musical artifacts, so our pieces have a tonic center with which a diatonic scale[3] is associated with. To choose the tonic center of our scale, we resorted to colour. Since we divided the 360 different hues into 12 main different hue bins and there are in total 12 possible halftones in music, we extract the most dominant hue tone of the image, and use the association of Figure 7, where we overlap the 12 hue tone circle with the Circle of Fifths.

The idea behind using the Circle of Fifths is that a fifth interval is very harmonious and sounds ”good” when we hear it, just like when we put red next to orange in a painting, for example. We should note that the first association made was that red be associated with A, since 440 Hz is the standard tuning pitch, which corresponds to the A tone, and in the visual spectrum, 440 Hz corresponds to the colour red which is its first visible colour.

To define whether the scale chosen was major or its relative minor, we turned to the average Value that dominant colour had. If it is lower than or equal to 0.5, the scale chosen is the minor one. If it is higher than 0.5, the scale chosen is the major one. This was done because major scales sound ”brighter”, while minor ones sound ”darker”.

4.2. Melody

Melody usually stands out in a song. With this in mind, we decided to associate the melody part of our musical artifacts to the saliencies we extracted.

[3] A diatonic scale is a scale with seven notes of which five are whole steps and two are half-steps. The standard C major scale is diatonic for example.

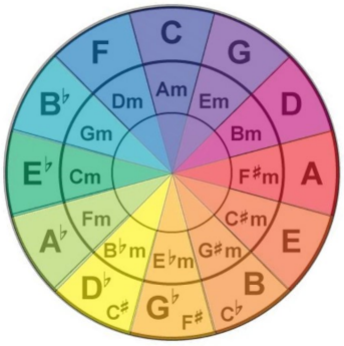


Figure 7: Colour and tone association according to the Circle of Fifths

We wanted to be able to use the shape of our saliencies and map them in some way into the melody of our musical artifacts. We believe different shapes can be associated with different types of sounds. Namely, more angular shapes suggest in our opinion higher-pitched sounds, and flatter shapes suggest lower-pitched sounds. A similar association was studied by Ramachandran and Hubbard [7], which became known as the Kiki/Bouba effect. Furthermore, sharper shapes give in our opinion a bigger sense of urgency and speed when compared to rounder shapes. So we associate the first type of shapes with quicker notes strung together, and the second with slower, longer notes. We can see different shape examples in Figure 8.

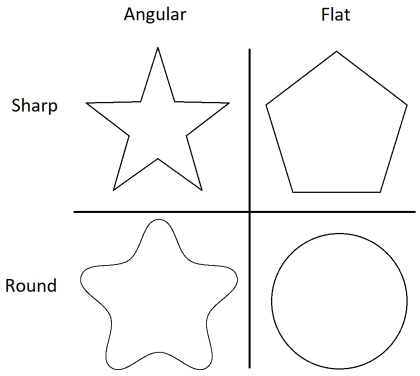


Figure 8: Different types of shapes

To measure how angular or flat and sharp or round an arbitrary saliency contour shape is, we decided to find its peaks and fit triangles onto them. After finding the peaks, to be able to draw triangles onto them, we need three points for each peak. The first point is the peak point, which we already have. For the other two points, we first need to define a baseline for the triangle, which is done by first finding the halfway points between peaks,

and then calculating the median of each peak, i.e., the median distance value between halfway points where the peak lies. Having the baseline value, and consequently the y coordinate for the other two points of the triangle, we just need to find the contour points to the left and to the right whose value is closest to the baseline to form the triangle. We define a neighbourhood, where we search for these points, as being $2 \times (peak\_value - baseline) \times contour\_size$ to both sides of the peak. Finally, we also have to take into account that a point further away from the peak is probably worse than one closer to it, even if its y value is closer to the baseline value. Given the peak point and the neighborhoods, we calculate the error for each of the neighborhood's points given by Equation (2)

$$error = |baseline - y| \times 1000 + |peak\_x - x| \times 0.06 \quad (2)$$

and we choose the points that minimize this error. The weights of 1000 and 0.06 were attributed empirically. After drawing the triangles onto the contour distance plot, we obtain a plot like the one in Figure 9.

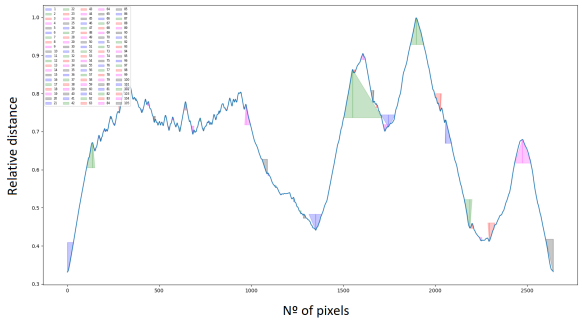


Figure 9: Contour distance plot for the dog image saliency with peak triangles fit onto them

Each triangle maps to a single note and we build the notes from left to right. The first note corresponds to the minimum contour distance point. To turn these triangles into notes, we need to define the pitch and the duration of those notes.

For the pitch, we measured the peak's angle using trigonometry. We decided to fit a whole scale between the minimum and maximum angles of the contour plot, and every note in between is uniformly distributed, with more acute angles representing higher notes, and vice versa. Also, our angle to pitch distribution is not deterministic. To generate more diversity between notes, we fit a gaussian around the chosen note with $\sigma = 3 \times \frac{repeated\_note^2}{2}$, where $repeated\_note$ is the number of times that note is chosen consecutively.

For the duration, we decided to use the triangle area to contour distance's integral ratio. We rounded the ratios to the decimal point, and defined that whole notes correspond to ratios rounded to 0.0, half notes correspond to ratios between 0.1 to 0.4, quarter notes correspond to ratios between 0.5 to 0.7, eighth notes correspond to ratios rounded to 0.8, and sixteenth notes correspond to ratios rounded to 0.9 and 1.

The last thing we defined with the contour distance plots were rests. We measure the relative distance between peaks, and then round them to the decimal point. These relative distances are usually very small, so we associated them with rest durations as follows: if the relative distance is rounded to 0.0, no rest is added between notes; if it is rounded to 0.1, an eight-note rest is added; if it is rounded to 0.2, a quarter note rest is added; every value higher than that corresponds to a half note rest being added between notes.

To define the octave of our melody tracks, we decided to use the saliency's most dominant colour average lightness: the higher the lightness, the higher the octave and vice versa. The range of possible octaves is then from C0 to C6. We divided the Lightness range of possible values into seven different bins, and classify the average lightness into one of these bins.

Regarding the timbre of our saliencies, we decided to use the most dominant colour of each saliency - since timbre is also known as tone colour - and we mapped different families or groups of instruments to different hue tones. Our association can be seen in Figure 10.

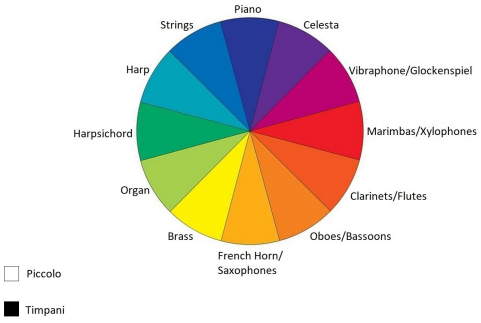


Figure 10: Hue tone to group of instruments association.

We should also point out that each image can have more than one saliency. We decided that, in that case, the saliencies' melody lines are played radially, that is, they start sooner if they are closer to the center of the image.

4.3. Harmony

We associate the harmony with the non-salient image, i.e, the image that remains when we remove the saliencies. We analyse the non-salient image as a whole, defining one harmony track per image, for each of our musical artifacts. We extracted the

non-salient image's dominant colours, and we use the same association used in Figure 7 to define the tonality of the chords we use in our harmony track. However, we define for each tonality 5 different types of chords: major and minor, augmented and diminished chords, and power chords [4].

If the chord's dominant colour has a Lightness value of 0.9 or higher, it is associated with an augmented chord; if it is lower than 0.1, it is associated with a diminished chord. If its Saturation value is lower than 0.25 (with its Lightness between 0.1 and 0.9), the chord becomes a power chord. Only if none of the cases above happens is the colour then associated with a major or minor chord. In that case, the type of chord is defined as follows: if the Value of the dominant colour is 0.5 or lower, the colour is associated with a minor chord; if it is higher than 0.5, the chord becomes major.

We assigned one chord to each of the musical artifact's measures. The chord for each measure is chosen according to the dominance of its dominance colour in the image. The range of possible octaves for the harmony track is between C2 and C5, inclusively. We do the median between the melody tracks and subtract one to this value.

For the timbre, we tried to measure if an image used colour tones close to each other, or colour tones that contrasted each other. To do this, we calculate the relative distance between each dominant colour and the most dominant colour of the image, and calculate the average colour distance for the whole image. Having the average colour distance for the image, we attribute the harmony track's instrument by picking the longest track's dominant hue as the hue center, and then traversing the circle in a clockwise or counter-clockwise fashion (randomly picked between the two) to decide the harmony track's hue, and consequently the harmony track's instrument, using our hue to instrument association from Figure 10.

If an image has only one melody track, its harmony track is composed of a chord line, that is, a harmony line in which all notes from its chord are played. If not, the harmony line consists only of a bass line, which only plays the tonic of its chord.

Finally, two versions are presented at this stage, an automatic one and a co-created one. The only difference between them is that, in the co-created version, both chord lines and bass lines were manually composed by us to each different family of instruments, and in the automatic version, the harmony lines are solely comprised of whole notes.

[4] Technically not a type of chord, but comprised by the tonic note and its perfect fifth.

5. Genetic Algorithm

In order to better emulate the creative process, we decided to implement our own GA, which we explain in this section.

5.1. Structure

Our GA is structured as follows: 1. Generation, 2. Selection, 3. Crossover, 4. Mutation, 5. Iteration. We first generate the initial population by using our feature mapping n number of times, where n denotes our population size. We use a mixed initial population of half automatic musical artifacts and half co-created artifacts. Our Selection step is standard, with an elitism factor of 25%. Then, each pair of individuals selected has a 90% chance of being crossed over and each one has an 80% chance of being mutated. We continue selecting individuals for 300 iterations when finally we output the fittest individual, which represents our genetic version.

5.2. Crossover

Crossover happens between a pair of individuals and it always involves half of each musical artifact's measures.

There are three different types of crossover that can happen: melody track crossover, harmony track crossover, and mixed crossover. The first type happens between the melody tracks of the two musical artifacts. Harmony track crossover is the same, but between the artifacts harmony tracks' measures. Finally, mixed crossover combines both previous types of crossover, switching both the chosen melody track's measures, and the same harmony track's measures across two musical artifacts.

5.3. Mutation

Mutation can happen to any individual selected. In a mutation, one feature of the selected individual is changed. In our GA, we defined six different types of mutations: note duration, note pitch, note switch, chord type, chord pitch, and melody track instrument.

The note duration mutation changes the duration of a randomly selected note, pitch mutation affects its pitch, and note switch mutation simply switches two notes. Chord type mutation changes the type of a randomly selected chord to another type (M, m, Aug, Dim, or PC), and chord pitch mutation changes its pitch to another pitch from the pitches associated with the dominant colours of the non-salient image. Finally, melody track instrument mutation simply mutates the randomly selected melody track's instrument to one of its neighbouring instruments according to Figure 10.

All of these different types of mutations can be applied to the selected individual, but only one of

them is chosen (in the case that individual is selected for mutation). The only type of mutation with a different probability of occurring is the melody instrument mutation with only a 1% chance of happening. Otherwise, the different mutations are distributed uniformly, each having a $99\%/5 = 19.8\%$ chance of happening.

5.4. Fitness Function

The fitness function evaluates how fit individuals are. In other words, it defines how "good" or "bad" individuals are, according to some criteria established *a priori*. We defined our criteria as follows:

- If the musical artifact starts or ends with the tonic chord, we add 100 to the fitness value. If the musical artifact both starts and ends with the tonic chord, $100 + 100 = 200$ is added to the fitness value.
- If the musical artifact starts or ends with a note that belongs to the chord of that measure, we add 100 to the fitness value. If it both starts and ends with a note of the measure's chord, $100 + 100 = 200$ is added to the fitness value.
- If the underlying chord appears three times in a row, that is, in three consecutive measures, we subtract 60 to the fitness value each time that happens. In the case that the harmony line is played by a bass line, instead of a chord line, we check the tonic itself, not only the underlying chord.
- For each melody track and for each of its measures, we check if there is a note on the strong beat, and if so, we add $10 \times note\_duration$, where $note\_duration$ is that note's duration.
- For each melody track and for each of its measures, we check if its strong note belongs to the measure's chord. If that is the case, we add $40 \times note\_duration$, where $note\_duration$ is that note's duration.
- For every melody track note, we add $30 \times note\_duration$ to the fitness value if that note belongs to its respective chord, where $note\_duration$ is that note's duration.
- For every melody track note that does not belong to either its respective chord or its respective scale, we subtract 60 to the fitness value.
- For each melody track, we check if each of its measures has its respective chord notes. We add 100 to the fitness value per chord note, that is, if no chord note is present in the measure we add 0, if one is present we add 100, if two are present we add 200, and if all three are present we add 300.
- If a melody track's measure has no notes that belong to its respective chord, we subtract 200 to the fitness value.
- If an interval between melody track notes is bigger than 12 semi-tones, we subtract the difference between their pitches times two.
- If there are multiple melody tracks that are played by the same instrument, we subtract 50 per repeated instrument to the fitness value.
- Finally, we subtract to the fitness function 200 times the standard deviation from the groupings of co-created vs automatic harmony lines. If a musical artifact only has groupings of one type we subtract $200 \times number\_measures$.

The overall fitness value of an individual is simply the linear combination of the different criteria values. These values were defined subjectively and empirically, and their values are relative, i.e., if we multiplied or divided every criterion's value by a constant, the fitness function would evaluate the same individuals in the same way.

6. Evaluation

To evaluate our results, we surveyed people and asked them if they thought our musical artifacts had quality, if they thought they were novel, if they enjoyed listening to them, and if they could relate them to their respective images. These questions were evaluated with a Likert scale from 0-5. We also present a criterion called Creativity Index that weproposed to help better evaluate our work.

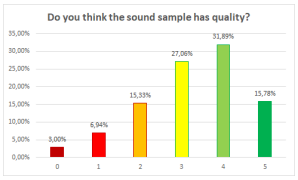
6.1. Creativity Index

Averaging the measured quality and novelty values for each musical artifact, we calculate what we call the Creativity Index (CI). This index is a rudimentary value we created to try to assess the creativity of our musical artifacts, and it is based on Boden's theory of creativity, which can be roughly summarized as "creativity = value + novelty". In that sense, our CI is simply given by Equation (3). We have to note that summing two different qualities such as quality and novelty is incorrect, but we took this liberty to more easily compare and evaluate our musical artifacts. Nonetheless, this index is an attempt at combining the two features proposed in Boden's theory, even if a naive and over simplistic one. The $avg(x)$ operator represents the average of the respective quantities. We also chose quality as being the best indicator of our musical artifact's value.

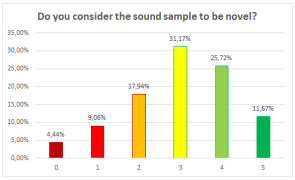
$$CreativityIndex = \frac{avg(Quality) + avg(Novelty)}{2} \tag{3}$$

6.2. Result Analysis

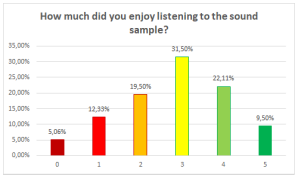
First analysing the answers as a whole, and disregarding images or versions, we obtained exactly 300 answers in total, of which 87% of the respondents said they considered our musical artifacts as music, with only 13% saying they did not. Regarding the four main questions asked, the results we obtained can be seen in Figure 11. Generally speaking, the results are fairly positive across all four questions, the worst answers perhaps being relative to people relating the sound to the images that were presented to them. Still, there are more positive answers (3 or higher) than negative ones in all questions, particularly regarding quality where the most answered value is 4.



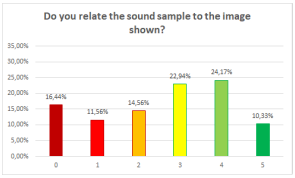
(a) Quality assessment question



(b) Novelty assessment question



(c) Aesthetics assessment question



(d) Relation assessment question

Figure 11: Assessment of our main hypotheses

Next, we analyse the musical artifacts by their different versions. As it can be seen in Figures 12 and 13, the genetic version usually has more favourable responses, albeit the difference is not that significant in most cases. In the case of music consideration, the genetic version topples the other versions, having a 6% increase in positive answers in relation to the automatic version. Also, in the quality and aesthetics question it has slightly better answers, which might be due to the harmony structure of our musical artifacts being more coherent and more familiar to most listeners. We should also point out that there is no significant difference between the relation of images to musical artifacts across versions, even if the automatic version is a much more direct translation than the other two versions, particularly the genetic version.

Finally, we calculated the CIs for the images included in our surveys. The results are shown in Table 1. First, we observe that no value is below the neutral point of 2.5, not even the lowest value, which means that we can say that our musical artifacts can be generally considered creative, according to our criterion. Next, the top 1% is given by two values in each row, which only means that those values are very close together (they are practically

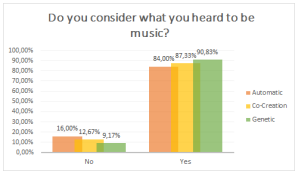
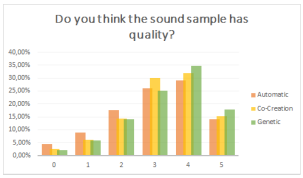
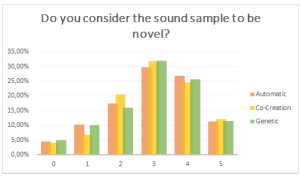


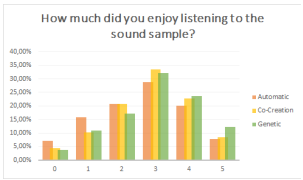
Figure 12: Music assessment question by musical artifact version



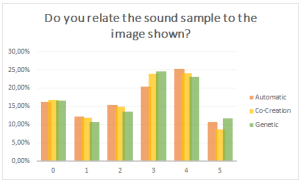
(a) Quality assessment question



(b) Novelty assessment question



(c) Aesthetics assessment question



(d) Relation assessment question

Figure 13: Assessment of our main hypotheses by musical artifact version

identical). We can also observe that every value has a higher weighted average[5] than its respective regular average (except for the lowest value), indicating more strong positive answers than negative answers. This difference becomes apparent when the best evaluated musical artifacts shift from automatic versions to genetic versions when considering the weighted averages vs the regular averages. This means we can probably consider the genetic versions as being more creative than the other versions, since we believe the weighted average represents better the answer distribution and gives relevant importance to the more extreme answers.

Creativity Index (CI)	face			pollock2		
	A	CC	G	A	CC	G
CI - Average	3,535714	3,37142857	3,4464	3,535714	3,371429	3,446429
CI - Weighted Average	3,634812	3,73454679	3,8137	3,634812	3,734547	3,813675
Creativity Index (CI)	**mondrian**			**rothko**		
	A	CC	G	A	CC	G
CI - Average	3,226563	3,28125	3,3242	2,550725	2,833333	2,891304
CI - Weighted Average	3,554947	3,65435715	3,6856	2,538922	3,04606	3,025748
Creativity Index (CI)	**picasso**			**dog**		
	A	CC	G	A	CC	G
CI - Average	2,912088	3,08791209	3,0604	3	3,098901	3,06044
CI - Weighted Average	3,06558	3,36333496	3,3501	3,263236	3,353009	3,256665

Table 1: Creativity Indexes across each of the different image's versions

6.3. Discussion

Recalling our goals once again, we set out to try to generate musical artifacts that could be considered creative, that were considered music and aesthetically pleasing, and that could be related to the images in which they were inspired. After

[5] In the weighted average, stronger opinions are given more prevalence.

analysing the results, we can safely say that our goals were mainly met: the surveyed people generally considered our artifacts valuable and novel - and hence creative -, they generally enjoyed listening to them (and considered them music), and, although results were more polarized in this case, there were still more overall positive answers than negative ones.

We can also conclude from the surveyed data that the reaction to questions regarding our genetic-generated version appears to be more favorable than that of the other two versions, which legitimizes its implementation.

7. Conclusions

With this work, we built a computer program that took inspiration from images and is capable of generating three different versions or types of musical artifacts: an automatic version, a co-created version, and a genetic version. Results were fairly interesting and promising, inviting further exploration of different methods and ways of trying to generate music from other sources of inspiration, or even from images as well.

Much work can still be done, both from the image processing part - trying to extract semantics from images or dealing with saliencies in another way for example - as from the music generation point of view - adding motifs and structure to the musical artifacts, and using other sounds than just MIDI sounds, to name a few.

Trying to combine machine learning techniques with a genetic algorithm could also prove worth further exploring. Since machine learning techniques are usually good at finding patterns, we could try to find what ”patterns” a song usually follows - what type of melodies, chord progressions, rhythmic sections are used, etc - and in that way try to measure its value, and then feed those outputs to a genetic algorithm with which we would try to add more diversity, more novelty on top of those musical artifacts.

References

[1] M. Boden. Computer Models of Creativity. *AI Magazine, 30(3), 23*, 2009.

[2] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[3] J. Canny. A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6):679–698, 1986.

[4] J. Fernández and F. Vico. AI Methods in Algorithmic Composition: A Comprehensive Survey. Technical report, Universidad de Málaga, 2013.

[5] S. Montabone and A. Soto. Human detection using a mobile platform and novel features derived from a visual saliency mechanism. *Image and Vision Computing*, 28(3):391–402, 2010.

[6] OpenCV. Contours : Getting started. `https://docs.opencv.org/3.4/d4/d73/tutorial_py_contours_begin.html`. Accessed: 2020-04-21.

[7] V. S. Ramachandran and E. M. Hubbard. Synaesthesia - A window into perception, thought and language. *Journal of Consciousness Studies*, 8(12):3–34, 2001.

[8] C. Rother, V. Kolmogorov, and A. Blake. “GrabCut” — Interactive Foreground Extraction using Iterated Graph Cuts. *ACM Transactions on Graphics*, 23(3):309, 2004.

[9] S. Suzuki and K. A. be. Topological structural analysis of digitized binary images by border following. *Computer Vision, Graphics and Image Processing*, 30(1):32–46, 1985.

[10] J. Teixeira and H. Sofia Pinto. Cross-Domain Analogy from Image to Music. Master's thesis, Instituto Superior Técnico, 2017.