MuSyFI - Music Synthesis From Images

André C. Santos[1], H. Sofia Pinto[1], Rui P. Jorge[2], Nuno Correia[3]
[1]INESC-ID, Instituto Superior Técnico
[2]CESEM, FCSH, Nova University of Lisbon
[3]NOVA LINCS, FCT, Nova University of Lisbon

andre.carvalho.dos.santos@tecnico.ulisboa.pt, sofia@inesc-id.pt
ruijorge@fcsh.unl.pt, nmc@fct.unl.pt

Abstract

MuSyFI is a system that tries to model an inspirational computational creative process. It uses images as source of inspiration and begins by implementing a possible translation between visual and musical features. Results of this mapping are fed to a Genetic Algorithm (GA) to try to better model the creative process and produce more interesting results. Three different musical artifacts are generated: an automatic version, a co-created version, and a genetic version. The automatic version maps features from the image into musical features non-deterministically; the co-created version adds harmony lines manually composed by us to the automatic version; finally, the genetic version applies a genetic algorithm to a mixed population of automatic and co-created artifacts.

The three versions were evaluated for six different images by conducting surveys. They 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. We gathered a total of 300 answers and overall people answered positively to all questions, which confirms our approach was successful and worth further exploring.

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

Introduction and Motivation

MuSyFI tries to model an inspirational creative process by automatically and semi-automatically generating music from images. Having chosen images as source of inspiration, any image, it generates music that can be perceived as being related to it. This relationship is subjective, since there is virtually an infinite number of musical artifacts that can be generated from an image. We do not target a sonification endeavour as we do not merely translate visual features into musical features to produce sound, but rather attempt to model a possible creative inspirational process whose starting point are images.

We aimed that our musical artifacts could be considered creative, aesthetically pleasing, and that the music could be relatable to the images that inspired them. Each one of these goals is subjective, which makes evaluation harder. This was positively evaluated through questionnaires answered from 300 respondents.

The rest of the paper is organized as follows: We start by reviewing related work. Next, we discuss feature extraction from images and describe the main features extracted. We then explain in detail the pursued visual to music mapping. Afterwards, we describe the genetic algorithm developed in this work, along with all its processes and the tests we did to validate our parameter choices. We present and discuss our results, and conclude 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[1].

Related Work

There are several systems that generate music computationally - EMI (Cope 1989), GenJam (Biles 1994), MuseNet (Payne 2019), to name a few - using different approaches - from Knowledge-Based Systems (KBSs) to Artificial Neural Networks (ANNs). However, few try to model inspiration. In one such work, Horn et al. (2015) extract the dominant colours from an image and shape a 3D vase according to those colours, effectively implementing an inter-domain mapping of features.

Later, Teixeira and Pinto (2017) generated music inspired in images, outputting three versions: raw, harmonized and genetic. For the raw version, the image is divided into quadrants which are then mapped to measures. The colours in each quadrant determined the notes and the chords played in each measure. The notes were chosen from a diatonic scale, and chords were major or minor, depending if the colour was warm or cold, respectively. The rhythm was picked from 44 drum patterns from a database, based on the emotion state of the quadrant given by its visual characteristics. The harmonized version adds a bass line to the raw version, and limits chords to three or four for the whole artifact, obtained from the whole image. Likewise, only two drum patterns are used in this version. Finally, the genetic version applies a GA to a population of 24 individuals (the raw version, the harmonized version, and 22 other variations thereof). The GA's fitness function is a combination of seven different criteria.

[1]http://web.tecnico.ulisboa.pt/ist178488/

Proceedings of the 12th International Conference on Computational Creativity (ICCC '21)
ISBN: 978-989-54160-3-5

Our work addresses the same problem as Teixeira and Pinto but takes a very different approach. For example, the authors divided the image into a grid of quadrants and analysed the image quadrant by quadrant, whereas we divide the image into its saliencies and non-salient background. Additionally, Teixeira and Pinto use pixel based features, whereas we also added shape and position related features.

Both approaches produced interesting and promising results. The difference lies in the path chosen to arrive at the same goal. Both paths are valid and both paths lead us through interesting landscapes, which, by its very nature, attests to the subjectivity involved in this approach and the weight personal aesthetics has in the final result.

Image Feature Extraction

To use images as an inspiration source, we first need to extract features from them to map these into musical features. In other words, we need to process the image.

Saliencies

Saliencies are features that draw attention to us when looking at an image or a series of images and saliency detection is an active research subfield of computer vision. OpenCV (Bradski 2000) has a saliency detection module with two static saliency[2] detection algorithms. One of those algorithms is the StaticSaliencyFineGrained (Montabone and Soto 2010). The authors based themselves on center-surround differences our eyes use to identify saliencies in images. An example of the saliency map obtained from this algorithm is shown in the second image of Figure 1, 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.

To improve upon this result, we used another image processing algorithm, GrabCut (Rother, Kolmogorov, and Blake 2004), also implemented in the OpenCV library. It segments the image into foreground and background homogeneous regions. The GrabCut algorithm usually needs a human to indicate where the background and foreground are. However, 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 algorithm's four possible values[3], we can then feed the image to the algorithm and use its output to obtain the correct saliencies of the image. We can observe this process in Figure 1.

Contours A contour is a curve that joins all the continuous points along a boundary which encircles a region of pixels that have the same colour or intensity. Contours were used to study the shape of the saliencies we extracted. We used the `findContours` function from OpenCV which receives a binary image as input and finds its contours. It is based on

[2]Static saliencies are detected specifically in single images rather than in image sequences or video.

[3]Foreground, Probable Foreground, Probable Background, Background.

Figure 1: Clockwise, starting from the top left image: Original image, its saliency map, and respective GrabCut output and input images.

(Suzuki 1985) and it works by applying border following to the binary image, labeling the borders it finds. Here, border and contour are used interchangeably.

We then find the contour's centroid and plot the distance to the centroid along the border, starting from the minimum distance point and continuing along the border, counterclockwise. This contour distance plot can be seen in Figure 2 with respect to the dog image of Figure 1. The y-axis represents the distance to the centroid 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, since bigger saliencies have bigger contours, bigger saliencies generate larger contour plots as well. The triangles in the plot are explained in subsection Melody.

Colour

We use both Hue, Saturation, Value (HSV) and Hue, Saturation, Lightness (HSL) colour models to extract what we call the dominant colours of an image. We divide the 360 hues into 12 main hue bins. 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. We can then plot the dominant colours histogram. In Figure 3 we show the histogram for the original dog image. The y-axis represents the number of pixels and the x-axis represents the respective hue bin. All hues whose bars are higher than the black horizontal line (representing the 10% threshold) are dominant colours. We should note that, for each bin, we have two overlapping 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 extracted. We do this to have an idea of the *pure* hue and an approximation of the actual hue that is present in the image. In Figure 3, an example of this are the two bars representing the main hue marked

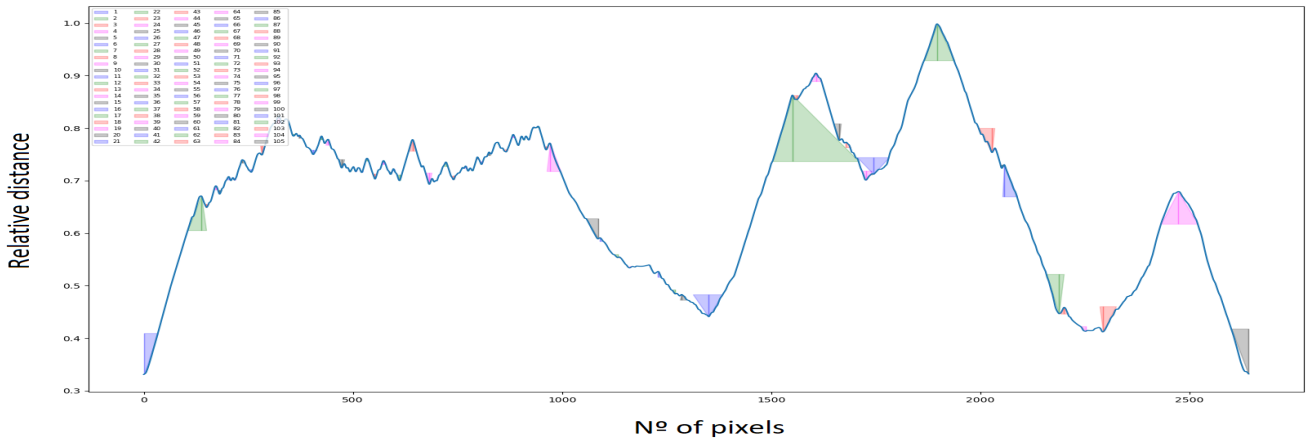


Figure 2: Contour distance plot for the dog image saliency with peak triangles fit onto them.

with number 2, one bar being lime green and the other being a darker shade of the same green. The bins are numbered from $[0, 11]$.

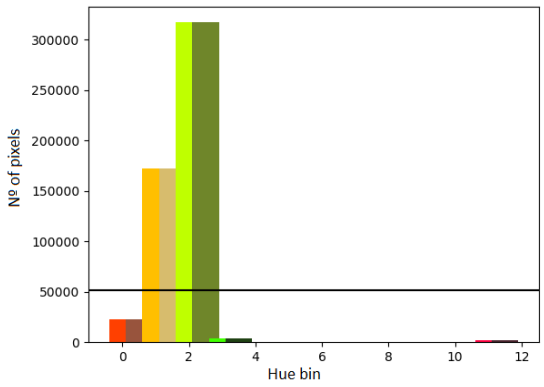


Figure 3: Colour histogram for the original dog image of Figure 1.

Edges

Edge detection is a classical image processing problem that tries to identify points in an image where brightness changes abruptly. The set of these points forms a set of curved lines called edges. John F. Canny developed a staple algorithm for edge detection that was eventually named after him, the Canny edge detector (1986) which is implemented in OpenCV's `Canny` function. The high and low thresholds were defined empirically and subjectively as 30 and 200, respectively, and we used the same thresholds for every image in our dataset.

Feature Mapping

Having presented the visual features and how we extracted them, we now explain the visual to musical feature mapping we conceived by explaining how general features of our artifacts were defined, as well as how the melody and harmony for our musical artifacts were pieced together.

General Composition Features

Before defining the melody and harmony parts of our musical artifacts, we first define their time signature, tempo, and key/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 having a faster pace than an image that does not have as many edges. We defined a minimum tempo of 60*bpm* and a maximum of 150*bpm* since they are relatively slow and fast tempos, respectively. We apply the Canny edge detector algorithm, count the number of non-zero pixels (edge pixels), and divide them by the total number of pixels in the image, obtaining what we call the $edge\_ratio$. This ratio is then divided by 0.3 to normalize it, since that was the maximum observed edge ratio in our dataset with our parameter choice. We use this new ratio to define the tempo of the song, being that 0 corresponds to 60*bpm*, 1 corresponds to 150*bpm*, and other ratios are distributed linearly according to Equation (1).

$$tempo = \lfloor\!\lfloor \frac{edge\_ratio}{0.3} \times (150 - 60) + 60 \rfloor\!\rfloor \qquad (1)$$

Finally, regarding the key, we decided to generate tonal musical artifacts so our pieces have a tonal center with which a diatonic scale is associated with. To choose the tonal center of our scale, we used colour. We extract the most dom-

inant hue tone of the image, and use the association of Figure 4, where we overlap a 12 hue tone circle with the Circle of Fifths[4] since similar hue tones harmonize well with each other (ex.: red and orange) as well as adjacent notes in the Circle of Fifths (ex.: C and G). We use 12 hue bins so as to make a direct association between colours and semi-tones. We should note that the first association made was that red be associated with A, since 440 Hz is the standard tuning pitch and corresponds to the A tone, and in the visual spectrum, 440 Hz corresponds to the colour red, which is its first visible colour.

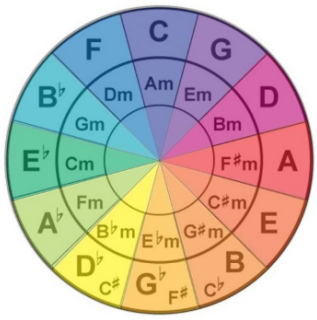


Figure 4: Colour and tone association according to the Circle of Fifths.

To define whether the scale chosen is major or its relative minor, we turned to the average Value (from the HSV colour model) that dominant colour has. 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".

Melody

Melody is usually what 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.

We wanted to use the shape of our saliencies and map it in some way into the melody of our musical artifacts. We associated more angular shapes to higher-pitched sounds, and flatter shapes to lower-pitched sounds. A similar association was studied by Ramachandran and Hubbard (2001). Furthermore, sharper shapes give, in our opinion, a bigger sense of urgency and speed when compared to rounder shapes. So we associated 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 5.

To measure the different types of shapes, we find the saliencies' contour plot peaks, fit triangles onto them, and then measure shape properties through the triangles. To be able to draw triangles onto the peaks, we need three points for each one. The first point is the peak point itself. The

[4]The Circle of Fifths is a musical tool that depicts the relationship between the 12 different tones on the chromatic scale.

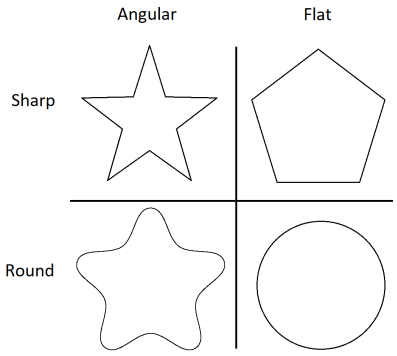


Figure 5: Different types of shapes.

other two are the baseline points, for which we first need to define the baseline value of the triangle. This is done by finding the halfway points between the peak and the previous/next peaks in the contour. Then, we define the baseline value as the median value between these two halfway points. Having the baseline value (fixed y), we just need to find the x coordinates for the triangle's baseline points. We do this by intersecting the baseline and the contour plot and finding, to the left and to the right of the peak, the points whose values are closest to the baseline to form the triangle[5]. We define the neighbourhood where we search for these points as being $2 \times (peak\_value - baseline) \times contour\_size$ to both sides of the peak, where $peak\_value$, $baseline$, and $contour\_size$ represent the ordinate of the peak, the ordinate of the baseline, and the number of pixels of the contour, respectively. Then we minimize the $error$ for each of the neighborhood's points (x, y), given by Equation (2):

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

$peak\_x$ is the abscissa of the peak. The weights, 1000 and 0.06, were attributed empirically. After drawing the triangles onto the contour distance plot, we obtain a plot as in Figure 2.

Each triangle maps to a single note and we add notes onto the musical sheet sequentially from left to right. The first note corresponds to the minimum contour distance point.

To turn the triangles into notes, we need to define the notes' pitch and duration. For the pitch, we calculated the peak's angle using trigonometry. We 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, but still choosing a similar note to the one picked, we fit a gaussian around the chosen note with a standard deviation $\sigma = 3 \times \frac{repeated\_note^2}{2}$, where $repeated\_note$ is the number of times that note is chosen consecutively.

[5]The contour plot is not continuous, so we need to find the closest point of the intersection and not the exact points.

For the duration, we calculate the triangle area to contour distance's integral ratio over the triangle's baseline points. We rounded the ratios to the decimal point and, since higher ratios means a good fit between the triangle and the peak, i.e., a sharp peak, and a lower ratio a rounder peak, we 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.

Rests were also defined from contour distance plots. We measure the relative distance between peaks - that is, we count the number of pixels between peaks, and divide them by the total number of pixels of the contour - and then round it 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 use the saliency's most dominant colour average Lightness (from the HSL colour model): the higher the lightness, the higher the octave and vice versa. We defined the range of possible octaves from C0 to C6. Then, we divided the range of possible Lightness values into seven different bins, classified the average Lightness into one of these bins linearly, and directly associated lightness bins with octaves.

Regarding the timbre of the saliencies' melody lines, we decided to use the most dominant colour of each saliency - since timbre is also known as tone colour - and we mapped different hue tones to different families or groups of instruments. This association is completely subjective and could have been done in many different ways, but we tried to arrange families so that neighbouring families were associated with similar colours. Since Lightness determines the octave of our tracks and for the same Hue we can have very different Lightness values as the two colour channels are independent, Hue only determines the instrument family and not the instrument itself. Accounting for Lightness, our association is presented in Table 1. Minimum and maximum Lightness values, which roughly correspond to black and white, were associated to Timpani and a Piccolo Flute since they are low and high register instruments, respectively.

Finally, each saliency inspires a melody line, so when an image has more than one saliency, multiple melody lines are generated. They are played radially, that is, they start sooner if their respective saliency is closer to the center of the image and vice versa.

Harmony

We associated harmony to the non-salient background, i.e, the image that remains when we remove the saliencies. We analyse the non-salient background as a whole, defining one harmony track per image. First we extract the non-salient image's dominant colours, and we use the same association of Figure 4 to define the tonality of the harmony track's chords. However, we define for each tonality five different types of chords: major and minor chords, augmented and diminished chords, and power chords[6].

If the chord's dominant colour has a Lightness of 0.9 or higher, the chord is associated with an augmented chord; if it is lower than 0.1, it is associated with a diminished chord. If its Saturation is lower than 0.25 (with its Lightness between 0.1 and 0.9), the chord becomes a power chord. If none of the cases above happen, the colour is 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, a major chord is used.

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 dominant colour in the image: more dominant colours are more likely of being selected and vice versa. The range of possible octaves for the harmony track is between C2 and C5, inclusively. To assign it, we calculate the median octave between the melody tracks of the artifact 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. We calculate the relative distance between each dominant colour and the most dominant colour of the image around the colour circle, and calculate the average colour distance for the whole image. Then, we assign the harmony track's instrument by picking the longest track's dominant hue as the hue center, and then traversing that distance in the colour circle in a clockwise or counterclockwise 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 Table 1.

If an image has only one melody track, its harmony track is comprised of a chord line, that is, at each measure all notes from its previously defined chord are played. If the image has several melody tracks, the harmony line consists only of a bass line, so only the tonic of the chord is played at each measure.

Two versions are presented at this stage: an automatic version and a co-created version. The automatic version is obtained using the mapping we explained in this section, with harmony lines solely comprised of whole notes. The co-created version stems from the automatic version, but its harmony track is selected from a set of manually composed harmony tracks. In total, we composed 24 harmony lines: one bass line and one chord line for each family of instruments from Table 1. The chord or bass line is chosen according to the instrument family.

Genetic Algorithm

Up until this point, our work can be interpreted as feature translation between domains and, although we believe our feature mapping is novel and rich, feature mapping alone does not model any creative process. While we could argue it possibly models the underlying inspirational process of

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

		Octaves						
		C0	C1	C2	C3	C4	C5	C6
Instrument Families/Groups	Marimbas/Xylophones(0)	Timpani(48)	Marimba(13)	Marimba(13)	Marimba(13)	Xylophone(14)	Xylophone(14)	Piccolo(73)
	Clarinets/Flutes(1)	Timpani(48)	Clarinet(72)	Clarinet(72)	Clarinet(72)	Flute(74)	Flute(74)	Piccolo(73)
	Oboes/Bassoons(2)	Timpani(48)	Bassoon(71)	Bassoon(71)	English Horm(70)	Oboe(69)	Oboe(69)	Piccolo(73)
	French Horn/Saxophones(3)	Timpani(48)	French Horn(61)	Baritone Sax(68)	Tenor Sax(67)	Alto Sax(66)	Soprano Sax(65)	Piccolo(73)
	Brass(4)	Timpani(48)	Tuba(59)	Trombone(58)	Trombone(58)	Trumpet(57)	Trumpet(57)	Piccolo(73)
	Organ(5)	Timpani(48)	Church Organ(20)	Church Organ(20)	Church Organ(20)	Church Organ(20)	Church Organ(20)	Piccolo(73)
	Harpsichord(6)	Timpani(48)	Harpsichord(7)	Harpsichord(7)	Harpsichord(7)	Harpsichord(7)	Harpsichord(7)	Piccolo(73)
	Harp(7)	Timpani(48)	Orchestral Harp(47)	Orchestral Harp(47)	Orchestral Harp(47)	Orchestral Harp(47)	Orchestral Harp(47)	Piccolo(73)
	Strings(8)	Timpani(48)	Contrabass(44)	Cello(43)	Viola(42)	Violin(41)	Violin(41)	Piccolo(73)
	Piano(9)	Timpani(48)	Piano(1)	Piano(1)	Piano(1)	Piano(1)	Piano(1)	Piccolo(73)
	Celesta(10)	Timpani(48)	Celesta(9)	Celesta(9)	Celesta(9)	Celesta(9)	Celesta(9)	Piccolo(73)
	Vibraphone/Glockenspiel(11)	Timpani(48)	Tubular Bells(15)	Tubular Bells(15)	Vibraphone(12)	Vibraphone(12)	Glockenspiel(10)	Piccolo(73)

Table 1: Hue to instrument family association.

a creative process, we need something more to model the exploratory component of creativity. With this in mind, and also to try to improve upon our results, we use a genetic algorithm.

Structure

We first generate the initial population by using our feature mapping n times to produce n musical artifacts, where $n = 100$ denotes our population size. We use a mixed initial population of 50% automatic musical artifacts and 50% 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 after which we output the fittest individual, which represents our genetic version.

Validation

A study was conducted where we observed how the fitness function evolved to select the values for population size, number of iterations, and percentages of crossover and mutation. We first tested different population sizes - 25, 50 and 100 - and we observed that the fitness of individuals increased the more we increased this number, but we settled for 100 since a larger number made the program slower. Next, we tested the number of iterations, starting with 50, then 100 and finally 300. We observed significant improvements over the fitness values of individuals, but a stagnation at around 300 iterations, so we fixed that parameter at that value. Finally, we initially set our crossover and mutation percentages as 90% and 80% respectively since both mechanisms are extremely important in the evolution of our musical artifacts. We then dropped each one individually to 10% and observed that the fitness values obtained were worse, so we kept the initial parameters of 90% for crossover and 80% for mutation.

Crossover

Crossover happens between a pair of individuals and it always involves half of each musical artifact's measures, although not necessarily the same half each time.

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 selected musical artifacts. Harmony track crossover is identical, 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.

The crossover step is crucial since it is what allows for a mixed harmony line in our genetic version. We use the fitness function's last criterion to try to group the genetic version's harmony line into uniform groups of the other versions' harmony lines. Hence, the genetic version's harmony line tends to switch between the two periodically.

Mutation

Mutation can happen to any selected individual. 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 contiguous 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 tonic to another pitch from the pitches associated with the dominant colours of the non-salient image. Finally, melody track instrument mutation simply mutates a randomly selected melody track's instrument to one of its neighbouring colour's instruments according to our association. 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.

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*. It is usually defined as a set of criteria that optimize a function, but since there are no optimal musical compositions, the criteria we present here are subjective, even if based in music theory

concepts. With f being the final fitness value, we defined our criteria as follows:

- If the musical artifact starts or ends with the tonic chord: $f \leftarrow f + 100$.
- If the musical artifact starts or ends with a note that belongs to the chord of that measure: $f \leftarrow f + 100$.
- Every time the underlying chord appears in three consecutive measures: $f \leftarrow f - 60$. If the harmony line is played by a bass line instead of a chord line, we check the tonic itself.
- For each melody track and for each of its measures, when there is a note on the strong beat: $f \leftarrow f + 10 \times note\_duration$, where $note\_duration$ denotes the duration of the note.
- For each melody track and for each of its measures, when its strong note belongs to the measure's chord: $f \leftarrow f + 40 \times note\_duration$.
- For every melody track note, if that note belongs to its respective chord: $f \leftarrow f + 30 \times note\_duration$.
- For every melody track note that does not belong to either its respective chord or its respective scale: $f \leftarrow f - 60$.
- For each melody track, if each of its measures has its respective chord notes: $f \leftarrow f + 100$ per chord note.
- If a melody track's measure has no notes that belong to its respective chord: $f \leftarrow f - 200$.
- If an interval between melody track notes is bigger than 12 semi-tones: $f \leftarrow f - 2 \times pitch\_difference$, where $pitch\_difference$ is the difference between the notes' pitches.
- If there are multiple melody tracks that are played by the same instrument: $f \leftarrow f - 50$ per repeated instrument.
- Finally, $f \leftarrow f - 200\times$ the standard deviation from the groups of co-created *vs* automatic harmony lines. At the start of the GA, since all musical artifacts only have groups of one type: $f \leftarrow f - 200 \times number\_measures$, where $number\_measures$ denotes the number of measures of a musical artifact.

The overall fitness value of an individual is the linear combination of the different criteria values. Some of these criteria were given stronger predominance and hence higher values. These values were defined subjectively and empirically, and are relative.

Evaluation

The hypotheses we wanted to verify were if people 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. Since all of our goals are subjective, to evaluate our results we decided to survey people. The hypotheses questions were evaluated with a Likert scale from 0-5.

Evaluation methodology

We evaluated our system for six different images. The images were selected in order to have a balance between more abstract and more concrete images, images with different colour spectra and edge predominance, as well as images that had only one predominant saliency, images with several saliencies, and images with no apparent saliencies at all. The images selected can be seen at http://web.tecnico.ulisboa.pt/ist178488/ and they include an Elf image, a Dog image, Rothko's Green and Maroon, Pollock's Blue (Moby Dick), Picasso's Girl Before a Mirror, and Mondrian's Composition No.10.

For each image, we generated three different musical artifacts, or versions - an automatic one, a co-created one, and the genetic one. That means that in total we evaluated 18 different musical artifacts. We split the six different images between four different surveys, two images per survey, one of the surveys repeating two of them, but paired differently. The surveys are identical in terms of the questions asked and their structure, the only differences were the images and the musical artifacts presented in each one.

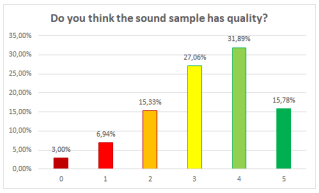
Regarding the structure of each survey, we first profile the respondents by age, gender, degree of musical knowledge, and music genre(s) they are most familiar with or prefer best. Both the degree of musical knowledge and music genre preference are asked to see if we could find any patterns among people of the same groups, regarding those categories, but only the former was relevant as discussed in the Result Analysis. The other demographics did not amount to any relevant findings either.

Then, we present people one of the three different versions for one image. We first ask people if they think the sound sample has quality, if they think it is novel - with a description the term -, and if they enjoyed listening to the sound sample. Only after having answered these questions we present the respondent with the image from which the musical artifact was generated. We then ask if they relate the sound sample to the image shown. These questions are evaluated using a Likert scale of 0-5. We do not tell the respondent the sound sample was inspired by the image. Then we repeat this for each of that image's other versions. The order in which we present the different versions is different for each image. Since we have three different versions, the possible ordered arrangements we can make with them are exactly six (matching the number of different images we chose to evaluate our system with), so we choose a different permutation for each image. This was done to avoid order bias, i.e., preventing people from getting too familiar and not answer each musical artifact's questions independently from the other ones.

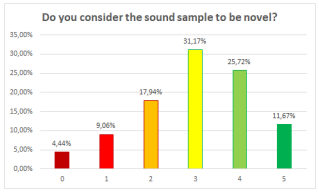
We also decided not to limit our respondents to people who did not know *a priori* about our project to try to obtain a larger number of answers and to be able to study for any bias regarding their previous knowledge of our musical artifacts being made by a computer. We asked people this at the end of each survey. The surveys were mainly distributed via social networks.

Result Analysis

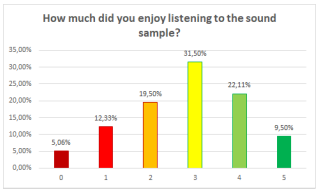
Analysing the answers as a whole, we obtained exactly 300 answers in total, of which 87% of the respondents said they considered our musical artifacts as music, with only 13% saying they did not. Regarding the four main questions asked (Figure 6), generally speaking results are fairly positive across all four questions, particularly regarding quality where the most answered value is 4.



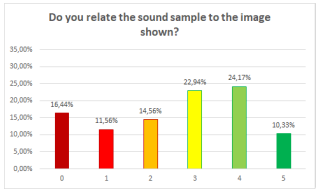
(a) Quality assessment question



(b) Novelty assessment question



(c) Aesthetics assessment question



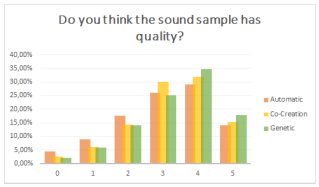
(d) Relationship assessment question

Figure 6: Assessment of our main hypotheses.

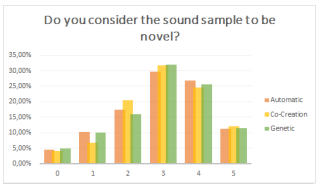
Next, we separate people who knew about the project (106 answers) from people who did not (194 answers). In general, people who knew about our project tend to lower their expectations and evaluate our artifacts as being better, and the opposite can be said about people who did not know anything about it. A note should be made about musicians or people with a degree in music however: they are the complete opposite, i.e., those who know about our project give our artifacts the worst scores, but those who do not evaluate them extremely positively. However, we should also point out that in total we had 12 answers for this category, 6 of people who knew and 6 of people who did not, so, while it is an interesting hint worth further exploring, we acknowledge that personal bias might be a factor here with such a small sample size. We also divided the overall results by version (Figure 7) and we can see the genetic versions were generally preferred.

Finally, we examined together the quality and novelty values for the images included in our surveys. The results are shown in Tables 2 and 3, where we present, for each image's versions, the median value, the Interquartile range (IQR), the average, and the weighted average of the answer distribution for each question. The weighted average takes into more consideration stronger opinions and is given by Equation (3) where w_av represents the weighted average, and $f(i)$ the absolute frequency for answer value i. The cells shaded in red are the worst evaluated versions for each question and each different average, whilst the green ones are the best evaluated versions.

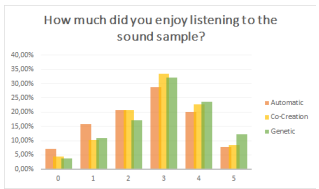
$$w\_av = \frac{\sum_{i=0}^{5} i \times f(i) \times (|2,5-i| \times 2)}{\sum_{i=0}^{5} f(i) \times (|2,5-i| \times 2)} \quad (3)$$



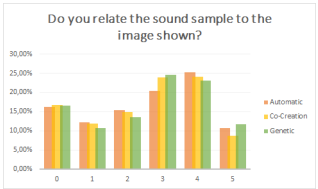
(a) Quality assessment question



(b) Novelty assessment question



(c) Aesthetics assessment question



(d) Relationship assessment question

Figure 7: Assessment of our main hypotheses for the three different versions of our system.

There is only one version whose average quality values are below 2.5 (the neutral value, considering the Likert scale used), all the other averages - including all novelty ones - are above this value. Taking these results into account, as well as Boden's creativity theory (2009) stating that creativity is some sort of combination between *value* and *novelty*, we can affirm that all our artifacts can be considered creative from a qualitative point of view. We should point out that a person who does not like a particular genre (subjective point of view) might still acknowledge that a particular song has quality, from an objective point of view. In that sense, we chose to ask if the artifacts had *quality* instead of what *value* they had.

Discussion

Recalling our goals once again, we aimed at generating musical artifacts that could be considered music and creative, aesthetically pleasing, and that could be related to the images in which they were inspired. After analysing results, we can safely say that our goals were met: the surveyed people generally considered our artifacts valuable and novel - and hence creative -, they generally enjoyed listening to them, and considered them to be music. Lastly, although results were more polarized, there were still more overall positive answers than negative ones regarding the relation between images and their respective musical artifacts. This is only natural since such relation is highly subjective. We can also conclude from the surveyed data, combining the answers for all the images, that our genetic version appears to be preferred over the other two versions, which also legitimizes its implementation.

Do you think the sound sample has quality?	face			pollock2			mondrian		
Version	A	CC	G	A	CC	G	A	CC	G
Median	4	4	4	3	3	3	4	4	4
IQR	1	1	1	2	2	2	1	1	1
Average	3,4929	3,5786	3,5714	2,7407	3,0123	2,9877	3,4688	3,5625	3,6563
Weighted Average	3,9176	3,966	4,0372	2,8556	3,2568	3,2563	3,9018	4,0964	4,095
Do you think the sound sample has quality?	rothko			picasso			dog		
Version	A	CC	G	A	CC	G	A	CC	G
Median	4	4	4	3	3	3	3	4	4
IQR	1	1	1	2	2	2	2	1	1
Average	2,2609	2,8406	3,029	2,8352	3	3,1868	3,1209	3,3187	3,5165
Weighted Average	2,1317	3,0519	3,2803	2,9507	3,1706	3,4728	3,4389	3,5983	3,9079

Table 2: Quality metrics across each of the different image's versions.

Do you consider the sound sample to be novel?	face			pollock2			mondrian		
Version	A	CC	G	A	CC	G	A	CC	G
Median	3	3	3	3	3	3	3	3	3
IQR	2	2	2	2	2	2	2	2	2
Average	3,0786	3,1643	3,3214	3,0247	3,0123	3,0288	3,0247	3,0123	3,0288
Weighted Average	3,3521	3,5031	3,5901	3,2687	3,3797	3,3029	3,2687	3,3797	3,3029
Do you consider the sound sample to be novel?	rothko			picasso			dog		
Version	A	CC	G	A	CC	G	A	CC	G
Median	3	3	3	3	3	3	3	3	3
IQR	2	2	2	2	2	2	2	2	2
Average	2,8406	2,8261	2,7536	2,989	3,1758	2,9341	2,8791	2,8791	2,6044
Weighted Average	2,9461	3,0403	2,7712	3,1805	3,5561	3,2275	3,0876	3,1077	2,6054

Table 3: Novelty metrics across each of the different image's versions.

Conclusions and Future Work

MuSyFI is a computer program that takes inspiration from images and is capable of generating three different versions of musical artifacts: an automatic version, a co-created version, and a genetic version. We wanted to model an inspirational exploratory creative process, therefore we used images as a source of inspiration and we implemented a genetic algorithm.

For the automatic version, a feature translation was established according to subjective and empirical criteria. We did not aim at making direct mappings or any sort of sonification, choosing instead indirect and innovative methods of feature translation across domains. We also added simple harmony lines composed by us to make the co-created versions, effectively making our program a co-creative endeavour as well.

Our GA combines both automatic and co-created versions into one genetic version. The musical artifacts generated are fairly interesting and we can notice the influence both have on the final genetic version. We should note that, while all genetic versions artifacts follow a certain aesthetic guideline which is noticeable (defined by the GA's fitness function), they still differ considerably amongst themselves inside that conceptual space. Furthermore, our genetic algorithm adds both cohesion and diversity to our musical artifacts by guiding them through the evolutionary process according to a fitness function, which in turn is based on music theory concepts.

All three versions were evaluated across six different images for a total of eighteen musical artifacts evaluated. Our goals were to be able to generate musical artifacts that could be considered creative, that could be considered music and aesthetically pleasing, and that could be related to their respective images. To evaluate our musical artifacts on such subjective goals, we surveyed people and asked their opinions. We gathered a total of 300 answers, which allowed a thorough evaluation of our musical artifacts. Results were fairly positive, with most people answering favourably to all the hypotheses we set out to validate, inviting further exploration of different methods of generating music from images.

That said, much work can still be done, both from an image processing perspective - 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 be worth exploring further. 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, structure, 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 and *novelty*, exploring different possible outputs on top of those musical artifacts. There is also space for improvement regarding our GA, namely coping with a multiple objectives fitness function, or using a different evaluation framework like SPECS (Jordanous 2012), for example.

Acknowledgments

This work was supported by FCT project UIDB/50021/2020.

References

Biles, J. 1994. GenJam: A genetic algorithm for generating jazz solos. In *ICMC*, volume 94, 131–137.

Boden, M. A. 2009. Computer Models of Creativity. *AI Magazine* 30(3 SE - Articles).

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

Canny, J. 1986. A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence* PAMI-8(6):679–698.

Cope, D. 1989. Experiments in musical intelligence (EMI): Non-linear linguistic-based composition. *Journal of New Music Research* 18(1-2):117–139.

Horn, B.; Smith, G.; Masri, R.; and Stone, J. 2015. Visual information vases: Towards a framework for transmedia creative inspiration. *Proceedings of the 6th International Conference on Computational Creativity, ICCC 2015* (June):182–188.

Jordanous, A. 2012. A Standardised Procedure for Evaluating Creative Systems: Computational Creativity Evaluation Based on What it is to be Creative. *Cognitive Computation* 4(3):246–279.

Montabone, S., and Soto, A. 2010. Human detection using a mobile platform and novel features derived from a

visual saliency mechanism. *Image and Vision Computing* 28(3):391–402.

Payne, C. 2019. Musenet. `https://openai.com/blog/musenet/`. *OpenAi*, Accessed: 2019-11-06.

Ramachandran, V. S., and Hubbard, E. M. 2001. Synaesthesia–a window into perception, thought and language. *Journal of consciousness studies* 8(12):3–34.

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

Suzuki, S. 1985. Topological structural analysis of digitized binary images by border following. *Computer vision, graphics, and image processing* 30(1):32–46.

Teixeira, J., and Pinto, H. S. 2017. Cross-Domain Analogy: From Image to Music. *5th International Workshop on Musical Metacreation (MUME 2017)* 1–8.