Sound Synthesis and Creation: Development of an application for composition support

Instituto Superior Técnico, Lisboa, Portugal

João Pedro Almeida Teixeira Paixão
joao.p.paixao@tecnico.ulisboa.pt

***Abstract*—Music is experiencing rapid and profound change. This is not a mere change of musical movement or genre as before, but a drastic shift in perspective: in the way we listen to it, how we perceive it, and how we create it. Machine Learning and Artificial Intelligence are now an unavoidable part of modern popular Music, both in conception (Audio Effects, Rhythm Generators) and dissemination to the general public (Streaming Recommendation Algorithms, Playlist Generators).**

In this Thesis, we explore these technologies as cooperative "partners" in the musical creative process. We propose an application that helps the creative process of Electronic Music Production, inspired by common techniques familiar to modern DAWs' (Digital Audio Workstation) workflow. This program creates sounds and organizes them into an arrangement, alongside MIDI compositions that are modified using effects present in modern Electronic Music genres, drawing motivation from works of artists like *Lorenzo Senni* and *Flume*.

As important as the developed software and algorithms, is the exploration of Computer Creativity (CC), where we seek new ways of using Algorithmic Composition (AC) to generate creative processes, and lead a study on the reaction of musicians to our System's creation.

***Index Terms*—Computational Creativity, Electronic Music Production, Algorithmic Composition, MIDI Modification, Musical Arrangements.**

I. Introduction

A. Motivation

Gaining traction in the past decade, Artificial Intelligence (AI) has made its way into the common stream of Music software. In today's age, almost every DAW is equipped with a variety of *Plugins* and *VSTs* (Virtual Studio Technology) that use Machine Learning (ML) and AI capabilities. The current tendency is that every step in digital music composition, production, and mastering, will be supported by intelligent technologies that not only process the input audio but also "listen" to it and make decisions based on their characteristics.

However, computer music generation from scratch still is a hardly accepted topic and did not yet become a staple amongst musicians. It's safe to say that we are at least one step from a reality where computer-generated music shares common ground with human-made music.

Alternatively, there is, for now, a wide opportunity for exploration of AI technologies and Computational Creativity (CC), not as stand-alone artists and composers, but as autonomous collaborators that assist the creative process, from drums and melody variations to *mastering* and *producing*.

The plugin manufacturer iZotope created Neutron [1], which listens to a track, identifies the given instrument, and makes suggestions for the mixing of said track. The software company Algonaut created Atlas [2], a software that uses AI to map a user's sample library into a 2D grid, grouping them together according to their characteristics such as timbre and style, facilitating sample picking. It also builds possible drum kits with these sounds. It is in this field of action that this Thesis follows this study, exploring the artificially creative capabilities of producing and composing alongside the human musician.

B. Problem Definition

The goal of this Thesis was to create an application that helps the process of musical composition. This platform provides two key functions, that can be used separately or together: one is to **synthesize, edit and manipulate input sounds to create a library of new sounds**, with a synthesis model that incorporates CC; the other is to integrate these sounds in a musical composition and have the capability to **edit, modulate, and add effects to the arrangement of these sounds**, also in an (artificially) creative manner.

With these objectives in mind, this work aims to create a software product that helps the creative process of choosing samples for given MIDI Patterns, and the following process of editing and altering these samples to create novel sounds. Additionally, this System modifies the input MIDI Patterns (in the temporal domain), in creative ways that go hand-in-hand with the manipulation of the respective samples.

This platform was mainly designed for tonal content, so it focused on processing MIDI inputs with harmony and melody, rather than rhythmic patterns.

Our System's MIDI Pattern manipulations were influenced by modern electronic music composition styles and MIDI editing techniques, such as MIDI chopping, glitch effects, scratched vinyl effects, and other erroneous and unpredictable manipulations, which incorporate the vocabulary of styles like *Dubstep* and *Hyperpop*. The "pointillistic" nature of the sample attribution and manipulation was also inspired by the works of Lorenzo Senni, a musician and visual artist known for his genre-defying Experimental-Trance pieces, most noticeably in *Scacco Matto* [3].

To construct our artificially creative process, we mainly explored Genetic Algorithms, most noticeably in the "Sam-

ple Choosing" Process, where we dove into ML Evaluation (Fitness) Functions, trained by data collected with the help of 13 musicians and music producers from Lisbon. For other creative modules, such as MIDI Modification (Mod) and the Choice of Parameters for a Synthesizer, Rule-Based Systems were constructed, with artistic choices curated by the author of this Thesis (and musician), respecting the music genre's popular characteristics.

By means of questionnaires, a set of musicians rated our System's outputs in terms of "likableness" and creativity, and compared them to examples made by people. Finally, we used creativity metrics of "well-being" and "cognitive-effort" presented by Colton et al. [4], to compute quantitative measures of musicians' "judging impact", from "Provocation" to "Instant Appeal".

The application was implemented with *Python 3.10* in a *Windows 10* environment. It is available in this ***Git Hub Repository***, and can run on a *Git Bash* Terminal (Instructions on the Repository).

II. Related Work

A. Algorithmic Composition and CC Techniques

1) Genetic Algorithms: Evolutionary Algorithms tackle optimization problems by means of operations that are inspired by biological evolution processes. The general approach is to create a set (population) of individuals, each one containing a set of possible parameters to optimize, and then, with an iterative procedure based on biological processes, evolve this population with the best-fitted individuals. Genetic Algorithms (GA) derive from these algorithms, where each individual is represented as a genetic code, and mechanisms such as *Selection*, *Reproduction*, and *Mutation* are deployed. The "Fitness Function" (FF) is what determines each individual's survival, representing the optimization problem in itself.

GAs are a prominent tool in Music AC and CC, from *Gen Jam*'s jazz solos generation, to the adaptation of musical motifs based on musician interaction on the *Multi-Agent-Based Evolver* (MABE). These algorithms have shown to be useful for optimization problems with vast search spaces, given their stochastic properties and use of biological mechanisms that prevent local convergence or other gradient-based problems.

2) Neural Networks: Artificial Neural Networks (NN) are models that emulate the learning capabilities of biological neural networks, based on the linear combinations of non-linear activation functions. Their ability to learn patterns and create intricate musical structures has resulted in groundbreaking projects and products in this sector. Google's Project Magenta [5] is one of the biggest AI projects on music generation, from drum patterns to chords and melodies, with a core foundation on NNs. "Data-driven" algorithms such as Amper [6] and MuseNet [7] also create compositions, whereas the first example is directly used in films, videos, and commercials.

B. Creativity Evaluation

Colton et al. [4] described creative acts through his Computational Creativity Theory, which consists of two models: *Framing information, Aesthetic measure, Concept, and Expression of a concept* (FACE), which outlines the possible creative processes of a system, and *Iterative Development-Execution-Appreciation* (IDEA), which reports the "*impact*" of said creative act. A particularly relevant notion introduced by Colton is the quantitative metrics of *well-being* rating (wb_m), which describes how much each member m of an ideal audience liked a creative act A (from -1 to 1); and the *cognitive-effort* rating $ce_m(A)$, that measures the audience's degree of mental exertion (and time invested) required to process or understand creative product "A" (from 0 to 1). From these ratings, a set of measurements were designed to quantitatively characterize the impact of a creative act A (Eq.1). n is the total number of audience members and $\bar{wb}_m$ is the average of well-being ratings.

$$
\begin{aligned}
disgust(A) &= \frac{1}{2n}\sum_{i=1}^{n}(1 - wb_i(A)).\\
divisiveness(A) &= \frac{1}{n}\sum_{i=1}^{n}|wb_i(A) - \bar{wb}(A)|.\\
indifference(A) &= 1 - \frac{1}{n}\sum_{i=1}^{n}|wb_i(A)|.\\
popularity(A) &= 1 - \frac{1}{2n}\sum_{i=1}^{n}(1 + wb_i(A)).\\
provocation(A) &= \frac{1}{n}\sum_{i=1}^{n}(ce_i(A)).
\end{aligned}
\tag{1}
$$

Equation 2 presents the subsequent compound measures created through manipulation of the measures in Equation 1.

$$
\begin{aligned}
acquired\_taste(A) &= \frac{popularity(A) + provocation(A)}{2}.\\
instant\_appeal(A) &= \frac{1 + popularity(A) - provocation(A)}{2}.\\
opinion\_splitting(A) &= \frac{1 + divisive(A) - provocation(A)}{2}.\\
opinion\_forming(A) &= \frac{divisive(A) + provocation(A)}{2}.\\
shock(A) &= \frac{1 + disgust(A) - provocation(A)}{2}.\\
subversion(A) &= \frac{disgust(A) + provocation(A)}{2}.\\
triviality(A) &= \frac{1 + indifference(A) - provocation(A)}{2}.
\end{aligned}
\tag{2}
$$

III. Architecture and Experiments

Figure 1 shows a simplified representation of the overall architecture. Two main chains of processes can be noted: the processing of samples; and the processing of MIDI. The first chain has the objective of editing and manipulating input sounds to create a set of new sounds (samples). The second chain modifies an input MIDI composition, creating new MIDI

files. These two chains, plus the optional use of a Synthesizer that modifies even further the samples with effects, create a unique Arrangement, playing a set of chosen samples with respect to the resulting MIDI compositions. The "Build Audio" unit creates the output audio file with this Arrangement.

1) Instrument Classes: Our System can build an Arrangement with three monophonic instruments (one MIDI File per instrument) with the following categories: Bass, Harmony, and Melody. These categories are allusive to three elements that form the tonal structure of a composition:

1) ***Bass***, usually in a lower register, sets the foundation of the song, emphasizing the root notes.
2) ***Melody*** is perceived as the leading element of a music piece, usually of a higher register.
3) ***Harmony*** has the role of adding notes, primarily different from the other two elements, creating a consonance in relationship with those other elements' notes.

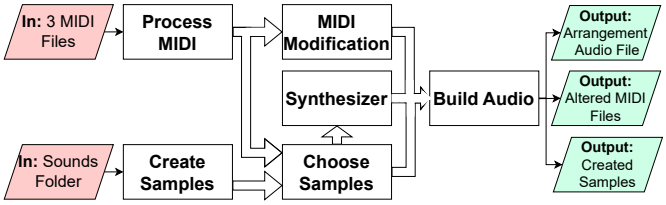

Fig. 1: Flowchart of the System's Architecture (Simplified).

Another instrument class was created in an effort to detect "unwanted" sounds for our System's purpose: "Miscellaneous" sounds, including atonal, "noise", and polyphonic sounds.

A. Sampler

The sampler's objective is to create samples by cutting sounds provided by the user. After this, an algorithm removes the "Miscellaneous" samples and classifies the remaining ones into the three categories of instruments, all based on features that can be associated with Timbre.

The Data set for the Instruments Classification is composed of 158 sounds from two sources: 151 sounds from ***freesound.org*** and 7 synthesized sounds using the *Behringer Deepmind 12* synthesizer. After cutting the sounds, 626 samples were obtained, 103 of which were used for testing. 7 spectral features were used, all extracted with the *librosa* Python package: Spectral Contrast (4 coefficients), Zero Crossing-Rate, Spectral Flatness, Bandwidth, Spectral Centroid, MFCCs (12 coef.), and Roll-Off Frequency (2 coef.).

We used Support Vector Machines (SVC) to perform both Miscellaneous and Instrument Classification. Miscellaneous Classification got an 85% F1-Score, with 85.3% Precision and 85.4% Accuracy, and the Instrument Classification got 89.9% weighted Precision and F1-Score, with 88.2% Accuracy. Considering the subjective nature of the Instrument Classes, these were very positive results, achieving a considerable distinction between Bass, Harmony, and Melody.

B. Modifying MIDI

This part of the System has the objective of reading and extracting information from the MIDI files provided by the user, and then applying effects to those MIDI patterns according to that information. Firstly the music piece's sequence of sections is determined, based on the similarity of the MIDI notes between sections. This is also useful in the Sample Picking Process in Section III-D.

1) MIDI effects: These are the developed MIDI effects (Mods), inspired by techniques used by musicians in modern electronic music:

1) **Multiplication:** Divide one note into smaller notes.
2) **Anticipation:** Anticipate note with a high-frequency repetition of said note. The space between added notes increases exponentially.
3) **Mute:** Mute specific note.
4) **Freeze:** Extend the note so that it only ends immediately before the next note.
5) **Velocity:** Change the velocity of a chosen note for a velocity based on the Average Velocity evolution function (explained in III-C).

Figure 2 shows a visual example of each MIDI Mod.

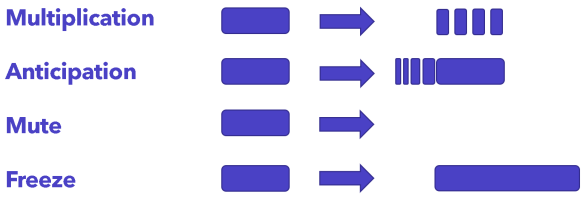

Fig. 2: Representation of each MIDI Mod's effect on one MIDI note.

C. Applying effects

To determine where the effects would be placed in the MIDI pattern, we devised a Rule-Based System that takes into account two characteristics of the original pattern: Note Density (ND) and Average Key Velocity (AV). Both measures were calculated for each section, creating two-time series.

These rules are intended to simulate the musical dynamic progression of electronic music. Giving an example with idioms common to electronic music producing: "Build-Ups" (Crescendos) are associated with an increase of MIDI effects probability, and "Breakdowns" are associated with a constant high probability of effects. In our case, we correlated "Breakdowns" with song sections that had the biggest ND value. AV was used to determine the note's velocity when the Velocity MIDI effect happened. Likewise, when a "Breakdown" occurred, the key velocity of a MIDI effect would be higher than in a "Build-Up" section.

With this in mind, three windows were used to describe the evolution of effects probability for each section, as shown in Figure 3: Crescendo **(A)**, where the probability of pattern modification rises; Mezzo **(B)**, in which the piece is at a moderate static dynamic (probability of pattern modification

is low); and Forte **(C)**, where the piece is at a higher level of static dynamic (high probability of modification). The x-axis of Figure 3 represents time in MIDI ticks, and y is the probability of pattern modification, on a scale from 0 to 1.

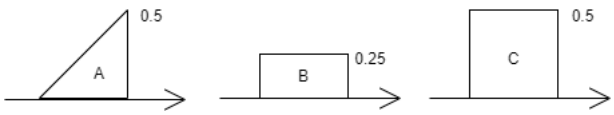

Fig. 3: Three possible windows of MIDI effect's probability/intensity evolution through a song's section

The sequence of these windows (one per section) is then determined by the "relative entropy" H_r of each section (ND or AV) and by the previous window state m_u^{i-1}. An example of these rules is represented in Equation 3.

$$\begin{aligned} H_r^{i-1} > H_r^i \quad \wedge \quad m_u^{i-1} = B \quad \longrightarrow \quad m_u^i = A \\ H_r^{i-1} > H_r^i \quad \wedge \quad m_u^{i-1} = A \quad \longrightarrow \quad m_u^i = B \\ H_r^{i-1} < H_r^i \quad \wedge \quad m_u^{i-1} = C \quad \longrightarrow \quad m_u^i = C \end{aligned} \tag{3}$$

D. Genetic Algorithm: Choosing Samples

After classifying all samples into each Instrument, comes the part of choosing the combination of samples for the final Arrangement with the following structure: one sample per instrument i (s_in), for each section of the song. It's important to remind that every instrument can have repeating sections.

To find the "best" sequence of samples, a GA was used, with a NN for Fitness Function. The GA was the best option considering the search space of considerable magnitude.

1) Fitness Function: The Objective of the Fitness Function's NN was to receive a representation of a sequence of samples and return a single score from -1 to 1. Two principles of "Arrangement Evaluation" were set:

1) Time-Dependency: The Likableness of an Arrangement can only be fully evaluated through the time-dependent sequence of samples: the order in which each sample occurs is determinant.
2) "Spatial" Dependency: The Likableness of an Arrangement is influenced by the combination of sounds in each time step: the interaction between different instruments matters.

With these principles established, we concluded that the best approach was to test two architectures: Recurrent (RNN) and Convolutional Neural Networks (CNN). The first is commonly used to classify and predict time series, and the latter is used to explore spatial dependencies in N-dimensional planes, as in image and video classification. As in Section III-A, we chose to experiment with different configurations of timbre-associated spectral features, using the *Librosa* Library.

2) Dataset: To build our dataset, we gathered 14 Electronic musicians from Lisbon, completing a total of 3 thousand Arrangement Scores. We asked each musician to classify two sets of Arrangements: one is a baseline test of 25 examples, the same for everyone, so we could analyze the "agreeableness" between musicians; and the other was a set of Arrangements different for everyone. For the sake of simplicity, we asked the raters to only use these 3 possible ratings: -1 (you did not like it), 1 (you liked it), or 0 (you feel indifferent/uncertain).

To measure the musicians' agreeableness on the baseline classifications, we first interpreted this evaluation as a binary classification: the negative class refers to -1 and the positive class is relative to 1. Given that very few 0 evaluations were given (9 in 275), we considered them as negative scores, which also helps interpret the results.

As a binary classification, we can assess agreeableness by the Fleiss' kappa value, a statistical metric that measures inter-rater reliability: how different evaluators provide consistent and similar ratings for the same set of items. This value ranges from -1 to 1, where 1 means raters are in perfect agreement, 0 means they are in agreement as if we classified by chance (randomly), and -1 indicates they are in less agreement than would be expected by chance. Equation 4 shows the formula to calculate the kappa (k) value, where P_o is the observed proportion of agreement among raters, and P_e is the expected proportion of agreement due to chance. Kappa values above 0.6 suggest considerable agreement among raters. Values below 0.4 usually indicate poor agreement.

$$k = \frac{\overline{P}_o - \overline{P}_e}{1 - \overline{P}_e} \tag{4}$$

Our baseline test resulted in a Fleiss kappa value of 0.28, so there was a very low agreement level amongst raters. In addition, the dataset was unbalanced, where 66.5% of ratings were negative.

Extrapolating these results to the 3K data set, it was expected that training a model to accurately score Arrangements would be very difficult: the data would be discordant some of the times, making it difficult to learn coherent patterns; and also would be difficult to evaluate the model's performance. Another restriction to this learning task is the size of the dataset: 3 thousand data points are undoubtedly a small number of samples, even for small linear regression tasks.

In response to these restrictions, a bigger dataset was created, by generating scores based on the classification of individual sounds. The generated Arrangement scores were determined under the simple assumption: if the sequence of sounds had at least one sound scored negatively, the Arrangement would also have a negative score. With this, a data set with 50 thousand data points was generated, with a 60% balance of negative scores. The classification of sounds was made by one musician.

E. Neural Network Architectures

Two objectives were in mind when developing these architectures:

1) The NN should have the smallest number of trainable parameters possible (around a thousand or less), so it could be potentially trained with a considerably small data set, with around 10 thousand samples.

2) Efforts should be made so that the model is interpretable, and clearly explores the "space" and time dependencies in the musical Arrangements.

Besides that, experiments showed training the model as a binary classification obtained better results. The output activation function was a Sigmoid function, and the loss function was binary Cross-Entropy.

1) RNN Architecture: Figure 4 shows the first of the two tested RNN Architectures, with one RNN Cell per Instrument, whilst the latter had only one Cell for all stacked Instruments. Each sample is represented as an array of Spectral Features. Firstly, the RNN Cells explore the time dependencies of the sample sequence, and then fully connected layers (FC) explore dependencies between instruments. The Three Cell model had 1975 parameters, while the single Cell one had 4935. Even though the second model attempted to reduce complexity in terms of RNN Cells, it increased the FC size considerably.

2) CNN Architecture: In the CNN models, the input shape was totally different: In the first model (3D CNN), the Arrangement was a time sequence of "frames", each being an MFCC 2D plane with dimensions $(n_{inst} \times 4, n_{mfccs}) = (12, 12)$, were each instrument had 4 intervals of 12 MFCC coefficients. These intervals were determined by the Spectral Envelope markers of each sample, as shown in Figure 5 (ADSR: Attack, Decay, Sustain, Release).

The second design tried to mitigate a simplification made on the first model: the convolutional window processed each time frame uniformly, so the filter would result in the same relationship between different Envelope Markers (e.g. the filter is the same for: 1. the "A-D-S-R" sequence in one instrument; 2. the "S-R-A-D" transition from one instrument to another). With this in mind, the second model was a 2D CNN, where each channel is dedicated to a single Envelope Marker, resulting in the input dimensions $(timesteps, n_{mfccs}) = (5, 12)$ with 12 channels. Note that for the CNN models, the number of sections (time-steps) is fixed, while in the RNN was not.

3) NN Performance: Table I shows the performance of all architectures on the generated 50K dataset (20% test split). In both RNN models, Long Short-Term Memory (LSTM) Cells were used, as they showed better results when compared to Gated Recurrent Units (GRU).

TABLE I: Performance of all architectures for the "group" 50K data.

Architecture	3 RNN	1 RNN	3D CNN	2D CNN
Accuracy (%)	73.8.1	68.8	84.0	89.6
Precision (%)	37.0	32.4	48.9	60.4
Recall (%)	95.5	88.9	94.2	96.7
F1-Score (%)	53.3	47.5	64.4	74.3
MSE	0.34	0.30	0.27	0.28

The 2D CNN had the best performance of all, getting an F1-Score of 74.3%. It also had the best Accuracy and Precision. Comparing the two architecture types, CNNs not only performed better but also met the number of parameters limit (711 and 909, respectively). Within RNNs, the model with 3 Cells performed better than a single cell.

Nonetheless, these results are only satisfactory considering the limitations of this classification problem: the data set was "generated", so it did not accurately describe the musicians' scores, which, in reality, were considerably disagreeable and inconsistent. Notwithstanding, it is possible to say that these original architectures show positive signs of potential for future exploration of Arrangement Evaluation.

4) GA Results: The left side of Figure 6 shows the distribution of Individuals' (Arrangements) scores through the first five generations, where we can see a clear convergence of the distribution to the positive class (above 0.5). After the 5th generation, the distribution remained concentrated on that class, with some marginal exceptions due to the Mutation and creation of new Individuals. The right side of Figure 6 shows that maximization of the GA's best score was obtained, increasing from below 0.93 to 0.98.

F. Synthesizer

The synthesizer block takes the chosen samples for each instrument and synthesizes new samples. This block randomly assigns a synth unit for each instrument:

1) **Auto-Filter:** Low-pass or high-pass filter with an LFO modulating the cutoff frequency.
2) **Granular:** Divides sound into smaller pieces (grains) and reorders them. Also adds spaces between the grains.
3) **Interpolator:** Interpolates (Amplitude-wise) from a given sound to another sound that is stored in a wavetable.

There is a set of parameters for each synth, that change throughout the Arrangement's sections. This creates sounds that are constantly evolving, emulating what modern electronic music producers do with a technique called *automation*: drawing lines in a 2D plain that map the change of a specific parameter throughout time (in a DAW).

1) Genetic Algorithm: To assemble the final Arrangement, a GA was used to determine each Synth's Parameters.

Each section type has a set of parameters for the given Synth unit, and the GA mechanism changes their order and creates new sets of parameters. The type of Synth is attributed to each instrument randomly.

2) Fitness Function: Our FF was inspired by Iannis Xenákis's work, more specifically "Metastasis". This music piece generates an "intricate web of sound", by means of the technique of "micropolyphony", where a large number of instruments are layered creating a complex and dense sound structure. This pattern can be represented in a 2D plane, with time on the x-axis and musical elements such as pitch on the y-axis, called a graph score. In this score, these mathematical techniques applied on various instruments form geometric patterns that evolve over time [8].

The visual effect present in these graphical scores sparked the inspiration for our FF: throughout the music piece, we can see that patterns are formed as instruments converge in the latent space, and other times diverge, having a "swarm-like" behavior. In the case of Xenákis, this latent space mostly mapped pitch through time, but in our case, we mapped the

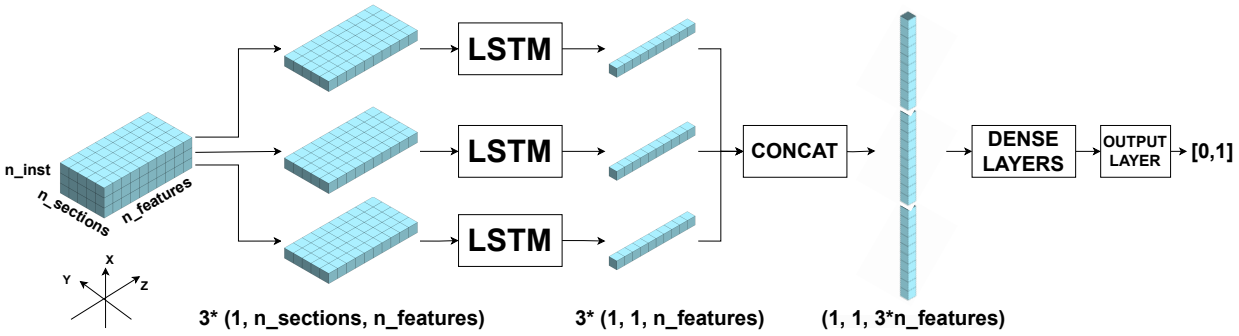

Fig. 4: Representation of the Three RNN Cells Model.

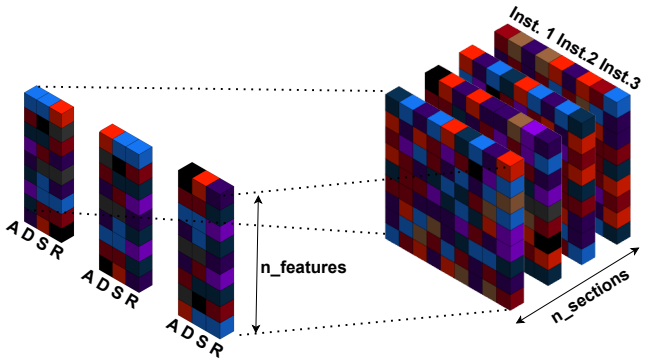

Fig. 5: Representation of the 3D CNN Model Input.

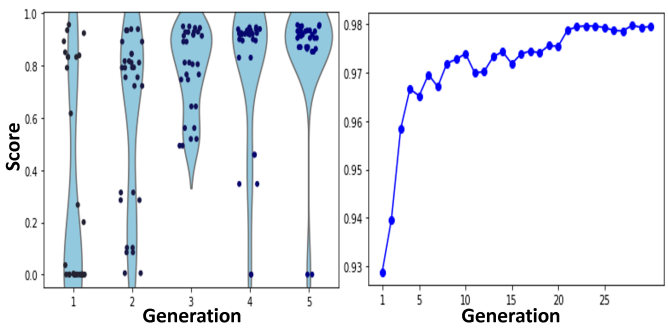

Fig. 6: Left: Distribution of Individuals for each generation in the Arrangement's GA (5 Generations). Right: Evolution of the GA's Best Score.

timbre/spectral similarity between instruments through time. For this, two mechanisms were used:

1) **Shaping the timbre:** The original samples for each instrument were altered by the Synth units, shaping the timbre and providing more or less similarity.
2) **Evaluating similarity:** To compare these sounds in each section, we computed the similarity between their spectral features, by means of a distance metric such as mean-squared-error (MSE).

Each Individual in the GA is a set of synth parameters for each instrument (and for each section). The FF ranks these Individuals based on the mean pairwise MSE between each instrument's spectral features. In some sections, this dissimilarity is rewarded, and in others penalized. The final score is the sum of each section's similarity score: in "converging" sections, the MSE is multiplied by -1 (penalization), and in the "diverging" sections MSE is just summed up (reward). The sections' "Objectives" were randomly chosen.

The extracted features for each synthesized sample were the same 22 coefficients (7 features in total) used in the Instrument Classification SVC model in Section III-A.

3) Fitness Scores Formula and Normalization: The Fitness Score for every individual, depicted in Equation 5, is the sum of the mean pairwise MSE between instruments ($\overline{MSE_s}$ in Equation 6), for every section S. Each $\overline{MSE_s}$ is multiplied by the "Objective" value for that section obj_s (-1 for convergence sections and 1 for divergence sections). For interpretation purposes, this score was normalized so that it would be in the range $[0, 1]$.

$$F_{IND} = \sum_{s=1}^{S} obj_s \cdot \overline{MSE_s}. \tag{5}$$

$$\overline{MSE} = \frac{\sum_{i=1,j=1}^{I} mse(features_i, features_j)}{I}. \tag{6}$$

As one can predict, scores will never reach the limits of the range: a 0 score would mean two sounds had identical features, and a 1 score would mean one sound would have all features at zero, and the other would have all at 1. With this in mind, it was expected that scores would take values that are marginally different from one another.

4) Synth GA Results: Each population had 10 Individuals, and the maximum number of Generations was 30. A 10% chance of Mutation was applied.

The violin plot in Figure 7 shows a noticeable increase in the mean Score throughout the generations, where the distribution shifted upwards from around 0.417 to more than 0.430. Coincidentally, the Best Score started below 0.422 and ended up above 0.432.

Even though these results show the GA worked, providing a maximization of the best Synth Parameters solution, it is clear some improvements can be made, mainly in the FF and the choice and processing of Features: 1. better Feature Normalization (such as Min-Max), which would need a new GA structure to compute the scaler with all the data at once; 2. use other spectral features that have more differences

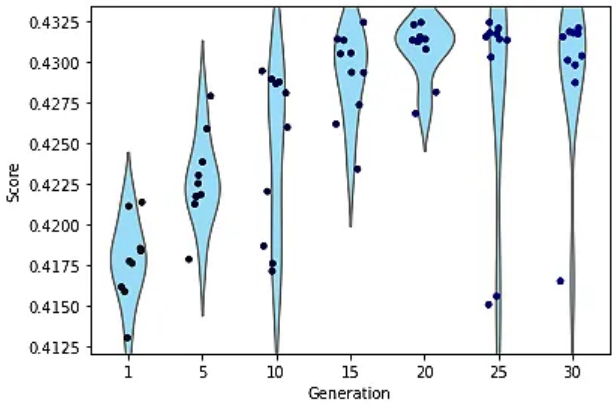

Fig. 7: Distribution of Individuals for each generation in the Synth's Genetic Algorithm

when comparing two signals; 3. use other Fitness parameters, considering other sounds' characteristics such as wb and ce.

G. Rendering Audio and Effects

After choosing the "best" Arrangement, the audio file must be created. For each MIDI note, the respective sample must have its pitch shifted to the desired frequency. After that, an Envelope cuts the sample into the desired duration and modulates its attack based on the key velocity. After this, if the user decides, the Synthesizer unit creates a new sound, given the respective Synth module. Finally, an *"Arctangent"* Compressor is applied.

IV. Evaluation and Results Analysis

The three main functionalities we want to evaluate in the questionnaire are the following: **1. the choice of samples, 2. MIDI Mods, 3. the Synth's Parameterization.**. This questionnaire was targeted at people with at least essential music education, not only because this System is designed for musicians, but also because the featured questions require a basic understanding of some Musical concepts.

This System's main objective is to complement a musician's creative process, not to recreate or substitute it. Nonetheless, it is valuable to assess if the System's creative choices are pleasant or/and if they come across as "alienating" and stand out of the composition/arrangement as an element with poor musical value. Considering this, the questionnaire centered its focus on comparing Human (Musician) and Machine creative decisions, aiming to answer the following questions:

1) Are these Arrangements considered to have some musical value? Are they "likable"?
2) Could these Arrangements be perceived as a Human creation?
3) Are these Arrangements (Sample picking and MIDI Mods) considered to be creative?

A. Questionnaires

The questionnaire has 2 exercises, the first with 4 music examples, and the second with two. In every exercise, the surveyee listened to 1-minute music pieces, and then answered questions about that example. Within each exercise, the music piece is an Arrangement that comes from the same original composition: in the first exercise *"Dance Tonight Revolution Tomorrow"*, and in the second exercise *"Canone Infinito"*, both by *Lorenzo Senni*.

1) Displayed Music Piece Examples: The first exercise evaluates Sample picking and MIDI Mod. These are the items for this exercise, each having one music piece:

1) **Exercise 1.1.** Arrangement with Samples picked by a Human, and no MIDI Mod (Original MIDI);
2) **Exercise 1.2.** Arrangement with Samples picked by the System, and no MIDI Mod (Original MIDI);
3) **Exercise 1.3.** Arrangement with Samples picked randomly, and no MIDI Mod (Original MIDI);
4) **Exercise 1.4.** Arrangement with Samples picked by a Human, with MIDI Mod.

The first three exercises aim to compare Sample choices since they have the same MIDI composition. The last exercise (1.4), with MIDI Mod, aims to compare a Human-made MIDI Composition and the resulting System's modified MIDI patterns. It is compared to Example 1.1, which has the same remaining conditions (Samples picked by a Human).

The second exercise compares two music pieces:

1) **Example 2.1.** Arrangement with Samples, MIDI Mod, and Synth Parameters chosen by a Human;
2) **Example 2.2.** Arrangement with Samples picked by a Human, with MIDI Mod and Synth Parameterization made by the System.

These two pieces aim to compare both MIDI Mod and Synth Parameterization, Human versus Machine. The Samples are the same for both examples, which were picked by a Musician. To allow a fair comparison, the Musician was given approximately the same amount of time as the System to create the MIDI Mods, which averaged half an hour. Also, the Synth units were the same for both examples, so the Musician could not use external effects.

2) Displayed Questions and Metrics: To compare music pieces, the two measures "Well-Being" (w_b) and "Cognitive-Effort" (c_e) mentioned in Section II-A were used. w_b is associated with the likableness of the music piece, in conformity with the conventional musical paradigm the listener resides. c_e, on the other hand, was used to assess the degree of mental exertion (and time invested) required to process or recognize Arrangements' characteristics such as timbre, loudness, and rhythm, but also how hard it is to capture the theme, message, or idea that the Arrangement conveys. As mentioned in Colton et al. [4], c_e is linked to the "novelty" aspect of the music piece, measuring the disruption of a current artistic landscape, hence being a compatible measure of "creativity".

Nonetheless, we also assess creativity separately, because there is an important distinction to be made between c_e and creativity. As cited in Colton et al. [4], *"positive appreciation tends to peak as novelty increases, but as the results of a creative act become too novel, it becomes difficult to put them into context, and overall appreciation of them drops*

(Wundt 1874)". This precisely describes the point: is the music piece considered creative and has positive appreciation, or does it merely sound considerably novel and disruptive? This is an important question to factor in when evaluating the performance of the System's output compared to a Human Arrangement.

From w_b and c_e it is possible to compute other descriptors of audience reaction to the music pieces, such as $divisiveness$, $popularity$, $shock$, and others (Equations in Section II-A).

The questions presented in each exercise were the following:

1) **Exercise 1**
   a) Do you like it?
   b) How would you rate the ce?
   c) Do you like the timbre of these samples?
   d) Do you think the Arrangement is creative?
   e) Do you consider this Music or just an experiment?
2) **Exercise 2**
   a) Which one did you like the most?
   b) Which one has/needs a higher level of ce?
   c) Which one is the most creative?
   d) How would you describe Arrangement A and B? [Human vs Machine generated]
   e) Do you consider this Music or just experiments?

Question $a)$ of Exercise 1 is relative to the "well-being" assessment. Question $c)$ is analogous but asks the surveyee to focus on the sample choice specifically.

3) Demography and Musical Background: 52 people answered the questionnaire, consisting mostly of 18 to 25-year-olds (88.5%). From our surveyees, 92.3% of them played instruments, and 69.2% composed music before. There was considerable diversity of musical knowledge levels within our studied population: 34.6% studied music formally for some years with $11,5\%$ at a university level. 5.8% are starting to learn an instrument/music theory, and the remaining 46.2% learned music theory or an instrument in the past.

Relatively to electronic music 92.3% said they like it, whereas 32.7% of the total actually compose in the genre. A majority of 53.8% using DAWs while composing, and also 48.1% using Synthesizers.

At last, it is possible to conclude that we were successful at gathering a "committee" that has the desired characteristics for this study: a great majority of musicians, mostly composing electronic music with DAWs and Synthesizers, at various levels of music knowledge.

B. Questionnaire Results

1) Exercise 1: Each surveyee was asked to give 4 quantitative ratings (per Music-Piece): w_b, c_e, creativity level, and likeableness of the samples' timbre (w_b specific to timbres). In accordance to Colton et al. [4] definition of w_b and c_e, these metrics had the range $[-1, 1]$ and $[0, 1]$, respectively. For creativity, we attributed the same range as c_e, and Timbre "Likableness" had the same range as w_b.

Tables II and III show the resulting Average and Median values of the above-mentioned metrics, for each example. Tables IV and V display the resulting compound measures.

First looking at the first 3 examples in Tables II and III that have no MIDI mod, Example 1 (Musician Arrangement) had the best score in terms of wb, and both Example 2 and 3 had the same score. Adversely, Example 1 rated significantly lower in terms of ce, whilst Example 3 (Random Arrangement) was the most "demanding" to listen to.

In terms of Creativity, Example 3 had the highest rating, followed closely by Example 2. Looking at the Timbre rating, we can see that there was no correlation with the w_b overall ranking: people rated the timbre lower than the w_b score, and the relationship between examples was different. Example 2 ended up having a higher "Timbre" score and Example 3 was rated significantly lower than the other two (on average).

Before analyzing Example 4's performance, let us take a look at Tables IV and V results for the first three Examples. We can see that Example 3 had lower scores in measures that value higher w_b, and better scores in those that award c_e: it was the most divisive and provocative. In turn, it was considered less indifferent (sparked a bigger reaction). Example 1 was the most popular, whilst the other two examples caused more disgust. Now looking at the compound measures in Table V, Example 3 caused the most Opinion Splitting (Op. Split.) and Forming (Op. Form.), and also, predictably, was considered more of an Acquired Taste (A.T.), possibly associated with being more unconventional, hence more subversive and less trivial. Less foreseeable, Example 3 was less shocking than the other two. Looking at the equation for this measure, we can explain this phenomenon given the smaller difference between "disgust" and "provocation": examples with bigger "disgust" and less "provocation" will have higher "shock" value. Other noticeable results are that the Machine generated example caused the least Opinion Splitting, while, in the other measures, sits between the other two examples.

Now looking at the results comparing Arrangements with and without MIDI Mod (Example 1 vs 4), we conclude that the latter had higher w_b and c_e scores. This trend was also followed in the Creativity and Timbre "Likeableness" rankings. **While in the other examples (Example 1, 2, and 3), higher c_e and "Creativity" rankings were followed by lower Well-Being, Example 4, in turn, was far more creative (and as cognitively demanding as Example 3) but was also clearly the most likable, Timbre-wise and overall**. This is a very relevant conclusion, as we cited in Section IV-A2, "positive appreciation tends to peak as novelty increases", but the System's MIDI Mod output was capable of maximizing creativity (novelty) whilst also receiving more positive rankings (value/w_b).

The measures in Tables IV and V show the same result. Example 4 was the most popular while maintaining provocation; was the least indifferent and caused the least disgust. In terms of compound measures, Example 4 was considered to be more of an Acquired Taste, because it was popular while maintaining provocation. It caused more Opinion Forming but had marginally less Instant Appeal (I.A.). It had less Subversion (Subv.) that Examples 2 and 3, but had about the same ranking as Example 1. As expected, Example 4 was

clearly the least trivial, however, caused less Opinion Splitting than Example 1: Example 4 had higher provocation and about the same divisiveness.

As Colton et al. [4] suggests, we can consider all compound measures (except triviality) as being directly proportional to "impactfulness". With this in mind, Example 4 caused more impact, while also being more likable and less trivial. Considering these results, we may conclude that the MIDI Mod algorithm was successful in developing a musical output that was deemed creative and also likable when compared to other musical creations with the respective modifications.

In terms of Sample picking, the Arrangement made by a real musician was more likable but was deemed less creative. The random Arrangement caused the most *ce*, but also was the least likable, clearly surpassing the threshold of novelty and damaging the "likableness". Nonetheless, we can verify that the Sample Choosing algorithm had more positive reactions than a random choice and was deemed more creative than the Human example.

TABLE II: Well-Being and Cognitive-Effort results in Exercise 1 of the Questionnaire.

		Well-Being		**Cognitive-Effort**	
Arrangement	MIDI Mod	Avg	Med	Avg	Med
Musician	No	0.08	0	0.36	0.25
Machine	No	-0.04	0	0.45	0.5
Random	No	-0.04	0	0.48	0.5
Musician	Yes	0.3	0.5	0.48	0.5

TABLE III: "Creativity" and Timbre "Likeableness" results in Exercise 1 of the Questionnaire.

		Creativity		**Timbre "Likeableness"**	
Arrangement	Mod	Avg	Med	Avg	Med
Musician	No	0.52	0.5	-0.12	-0.5
Machine	No	0.57	0.5	-0.09	0
Random	No	0.58	0.5	-0.22	0
Musician	Yes	0.71	0.75	0.21	0.5

TABLE IV: Measures resulting of C_e and W_b scores for each Arrangement in Exercise 1.

Arrang.	Mod	Disgust	Divisive.	Indiff.	Popular.	Provoc.
Mus.	No	0.46	0.46	0.56	0.54	0.36
Mach.	No	0.52	0.43	0.58	0.48	0.45
Rand.	No	0.52	0.51	0.5	0.48	0.48
Mus.	Yes	0.35	0.47	0.47	0.65	0.48

TABLE V: Compound Measures resulting of C_e and W_b scores for each Arrangement in Exercise 1.

Arr.	Mod	A.T.	I.A.	Op. Split.	Op. Form.	Shock	Subv.	Trivial.
Mus.	No	0.45	0.59	0.55	0.41	0.55	0.41	0.6
Mach.	No	0.46	0.52	0.49	0.44	0.54	0.48	0.56
Rand.	No	0.48	0.5	0.51	0.49	0.52	0.5	0.51
Mus.	Yes	0.56	0.58	0.5	0.48	0.44	0.42	0.5

2) Exercise 2: Two Arrangements are directly compared: one with Musician's MIDI Mod and Synth Parameterization made by a real Musician and another made by the System.

The first three questions were about w_b, c_e, and creativity. 60% of people said they liked Arrangement "A" (Human) better, while 71% said Arrangement "B" had a higher level of c_e. On the other hand, 67% considered Arrangement "B" more creative. A relationship can be made between the level of c_e, and the ranking in creativity, even though there is a margin of people that possibly considered "B" more cognitively demanding, but did not consider it to be the most creative.

After that, people had to estimate if the Arrangements were made by a Musician or a Machine. Figure 8 shows the pie chart of guesses for Arrangement "A", where we find that 19% of people thought that this Arrangement was machine-generated, with a total of 40% "leaning" to this prediction. The larger portion of people (29%) could not tell if it was machine-generated or not, and only a small portion of 3.8% was sure it was human-written.

Before taking conclusions, let us compare it to Arrangement's "B" pie chart in Figure 8. This time, the clear majority of people (58%) classified it as being machine-generated, even though only 19% were sure. Fewer people (21%) could not make a guess, and the remaining 21.1% thought it to be made by a real musician.

Immediately, we can say that it was clearer to the audience that "B" was machine-generated when compared to "A"'s assessment: 58% guessing correctly in "B", and 30% for "A". Nonetheless, "A" still brought up considerable doubts, with 40% guessing incorrectly. A part of the "machine-generated" guesses can be attributed to the fact that both Arrangements have characteristics that can be interpreted as "machine-generated": even though "A" has synth parameters and MIDI mods made by a person, the "Playback" algorithm (inserts samples in the correct MIDI notes) is the same in both Arrangements and the same for the synthesizer.

However, it is important to note that "B" mislead a considerable amount of musicians (42%), meaning it showed signs of emulating a human's creative process to some degree. Given that the main objective of this System is to serve as a complimentary source of creative decisions that generate "likable" outputs, we consider these results satisfactory, in regards to the MIDI Mod and Synth algorithms.

On a final note, we also have to mention that the higher percentage of correct guesses on Arrangement "B" can also be a sign that the Synth algorithm still has room left for improvements: not only the parameters itself (ex: the range of the LFO or cutoff frequency) can still be optimized in a way to avoid harsher sounds, but also the FF is not sensitive to uncomfortable and outlandish sounds (it only tries to maximize the cost function relative to the divergence or convergence of sounds). In "A", the musician could more easily avoid setting parameters that led to harsher sounds, creating more "pleasant" sounds. Arrangement B was consequently more "flagrant".

Regarding classifying each Arrangement as Music, the majority (75%) considered both Arrangements to be Music.

7.7% did not consider "B" Music, and 3.8% the same but for "A". If we, for this purpose, dismiss the people that did not consider both Arrangements to be Music (13.5%), we see that a clear majority agreed that both Arrangements could be classified as Music. "B" had a lower acceptance rate, which corroborates with its lower w_b and higher c_e rating.

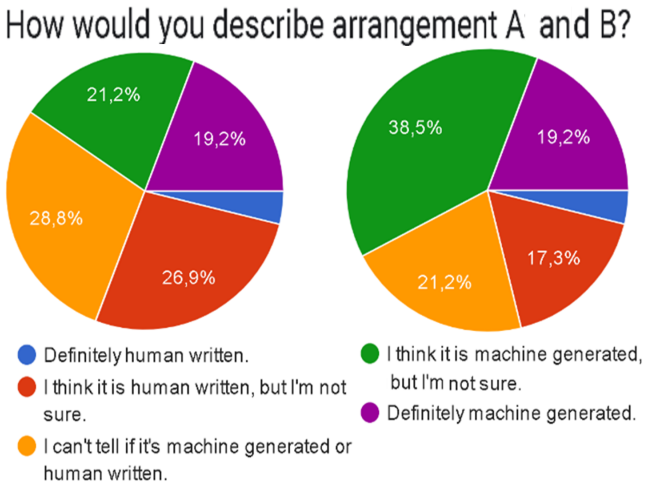

Fig. 8: Pie chart with the distribution of people's prediction about the nature of Arrangement A (Left) and B (Right) creative process.

V. Conclusion

Without any doubt, many challenges were met in this Thesis: there was no known study that explored Arrangements in this way: Combining "Timbre" classification models with MIDI Compositions, creating a complex sequence of sounds for more than one instrument. Consequently, no data set could fit our needs, both for Instrument and Arrangement Classification. A lot of time was spent figuring out the best way to represent an Arrangement for the NN's input, as we saw in Chapter III-E. we also had to develop an intricate algorithm to deconstruct MIDI into arrays of notes and apply mathematical functions to create effects.

Despite these setbacks, we can now contemplate the final product and realize that these challenges served as strengths and motivators for this Thesis, and for its relevance in CC exploration. We were able to successfully classify diverse sounds into new instrument categories, which can be useful in Music Information Retrieval and AC: Bass, Harmony, and Melody are the fundamental elements of the tonal structure of common modern Music. After that, four different NN architectures were developed (2 RNNs and 2 CNNs), that explore the possibilities of Music Classification in terms of Arrangement (sequence of samples). Even though some results suffered from the statistical barriers, one CNN model showed promising results, and many important conclusions came from this experimentation (Chapter III-E).

Moreover, we created three Synth Units from the ground up. One of the most remarkable features was the "evo" function, which emulated the *automation* effect of DAWs. This Synth also had the ability to choose its parameters, by criteria of a FF inspired in the works of Iánnis Xenákis such as *"Metastasis"*, modulating the evolution of the instruments' timbres by "converging" or "diverging" their spectral features throughout the music piece.

These accomplishments were further magnified in the questionnaire results. We were able to conduct a study of a well-represented group of musicians' w_b and c_e ratings, as well as evaluating measures of Creativity and Likableness of Timbre. In this study, we clearly saw that most musicians liked our System's output the most, in terms of MIDI Mod, even being considered the most "creative", with a great level of "novelty" and "value" simultaneously. We also explored Colton et al. [4]'s *"IDEA Descriptive Model"*, computing measures such as "popularity", "shock" and "subversion", which proved that our System's output was positively impactful. When asked to compare two Arrangements, one with synth parameters and MIDI mods chosen by a Human, and another by our System, musicians had difficulty deciphering their creative process nature. Even though the majority (60%) liked the Human-created example more, the one generated by our System was deemed more Creative.

Regardless, it is now important to mention the various axis that can be improved in future work. Immediately, a major improvement in the NN model's performance would come from gathering more data. Not only it would make the model generalize better but also enable the exploration of more complex architectures, capable of capturing more intricacies in the sequence of sounds' spectral characteristics. Efforts could also be made to develop other ways to represent an Arrangement and explore more spectral features. In terms of scoring the Arrangements for the NN, a new classifying method could be created, which could integrate ce and other metrics that would enrich the description of each creation.

At last, we can conclude that the overall objectives and functionalities were positively evaluated, and we hope our tool may come to be used by fellow musicians.

References

[1] iZotope, "Aiva," https://www.izotope.com/en/products/neutron.html, accessed: 2022-01-20.

[2] Algonaut, "Atlas 2," https://algonaut.audio/, accessed: 2022-01-20.

[3] L. Senni, "Scacco Matto," https://lorenzosenni.com/, accessed: 2022-01-20.

[4] S. Colton, J. W. Charnley, and A. Pease, "Computational creativity theory: The face and idea descriptive models." in *International Conference on Computational Creativity (ICCC)*. Mexico City, 2011, pp. 90–95.

[5] A. Roberts, J. Engel, Y. Mann, J. Gillick, C. Kayacik, S. Nørly, M. Dinculescu, C. Radebaugh, C. Hawthorne, and D. Eck, "Magenta Studio: Augmenting Creativity with Deep Learning in Ableton Live," in *Proceedings of the International Workshop on Musical Metacreation (MUME)*, 2019.

[6] Amper Music, "Amper," https://www.ampermusic.com/, accessed: 2022-01-20.

[7] Open AI, "Musenet," https://openai.com/blog/musenet/, accessed: 2022-01-20.

[8] S. Sterken, "Music as an art of space: interactions between music and architecture in the work of iannis xenakis," *Resonance: Essays on the intersection of music and architecture*, pp. 21–51, 2007.