UKRAINIAN CATHOLIC UNIVERSITY

BACHELOR THESIS

# Visual representation of music score

*Author:*
Yuliia POCHYNOK

*Supervisor:*
Andiy GAZIN

*A thesis submitted in fulfillment of the requirements for the degree of Bachelor of Science*

*in the*

Department of Computer Sciences
Faculty of Applied Sciences

Lviv 2022

# Declaration of Authorship

I, Yuliia POCHYNOK, declare that this thesis titled, "Visual representation of music score" and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: ________________________________________

Date: ________________________________________

*"Where words fail, music speaks"*

Hans Christian Andersen

UKRAINIAN CATHOLIC UNIVERSITY

Faculty of Applied Sciences

Bachelor of Science

**Visual representation of music score**

by Yuliia POCHYNOK

# *Abstract*

Music has been a significant part of human life from the beginning of time and grew even closer with the development and dissemination of streaming platforms in the market. Although the accessibility of classical music has reached its peak compering to old times when listening to famous musicians and composers was a privilege for the upper class, the visual representation of classical pieces remained fairly blank and unoriginal, with no references to actual music.

Therefore this project focuses on creating a visual illustration of the music score, that will serve as a good portrayal of classical composition, taking into account its properties such as instruments, music key, notes duration, etc. Four different approaches with a selection of additional attributes were implemented, encoding music score data into colour, opacity, width, size, amount, position, etc. As a result, an approximate amount of 480 slightly different variations of album/song covers can be generated for one music composition.

# *Acknowledgements*

First of all, I want to thank my supervisor Andriy Gazin for leading me in every step of this work, sharing his time and knowledge. His advice and enthusiastic feedback were a great help in making this project better and kept me motivated and eager to work during the whole course of writing it.

I would like to thank Ukrainian Catholic University for always challenging me, pushing me out of my comfort zone and teaching how to face and overcome difficulties.

I am infinitely grateful to my family. Thank you for being with me during every cry, every laugh, every busy day and every happy hour. Thank you for always believing in me, even when I didn't believe in myself.

Special thanks to my friends, who understood me and shared my worries as well as my successes. You made every sorrowful moment more bearable and every happy moment even happier.

Also, I would like to thank myself for carrying on despite everything. You've made it.

And finally, thanks to the Armed Forces of Ukraine, Territorial Defence Forces, medics, volunteers and every Ukrainian who fights for our freedom. The fact that I could safely sit and write this work any time of the day is owing to your enormous courage, the brilliancy of mind and inviolability of spirit. Thank you!

# Contents

# List of Figures

# List of Tables

# List of Abbreviations

| | |
|---|---|
| **MAM** | **M**usic **A**nimation **M**achine |
| **BPM** | **B**eats **P**er **M**inute |
| **D** | **D**enominator |

*Dedicated to my family, always and forever.*
*And to the music, that has always been my best friend, my refuge and my motivation.*

# Chapter 1

# Introduction

## 1.1 Motivation / Topic relevance

Music has been a friend of humans since the beginning of time, playing a significant role in communication, therapy, and entertainment. Through time not only humans' relationships and type of music have changed, but also its representation. In about 1400 BC the main method to represent music was making drawings and logo-syllabic scripts on stones and tablets. While now the most common visual representation of a song or a track is probably its album/collection cover created by designers specifically for certain compositions. It is working nicely in our generation for distinguishing songs between each other and connecting their sounds with the certain picture, colours, and forms portrayed on the cover. However, the accuracy and inclusiveness of this approach can be discussed.

- Firstly, often album/song covers are not related to the actual music it represents. Covers have a tendency to show an artist of the composition or a picture related to its name (or lyrics if it is a composition with vocals) and not an actual representation of the music the person is listening to.
- Secondly, the approach of album covers is used mainly for modern artists and compositions. While classic pieces are represented either by a picture of a musical instrument or a portrait of the composer, which makes it hard to distinguish them one from another. Having all 41 of Mozart's symphonies be represented by the author's photo is definitely not the most efficient method.

## 1.2 Goal

This work aims to create a visual representation of the music score, that will serve as a good demonstration and portrayal of classical music composition, taking into account its instruments, music key, notes duration, etc. In the future, these kinds of visualizations can be used as a representation of classical pieces in music streaming services such as Spotify, Apple Music, Deezer, etc.

## 1.3 Objectives

1. Explore existing approaches for music visualization
2. Explore different music notation programs for sheet music
3. Choose a format of data we will accept as input

4. Identify and specify which units of information are contained in this data and be used for visualization
5. Choosing musical composition to visualize
6. Data parsing and converting it into a convenient format
7. Cleaning the data, preparing it for visualization creation
8. Testing different approaches for visualization
9. Improving generated visualizations, trying different alterations
10. Creating final visualizations
11. Evaluating and describing the results

# Chapter 2

# Background information

## 2.1 Common musical notations

Since canonical notations are also a visual representation of music, it is important to explore and learn where and how data is presented in them.

### 2.1.1 Modern staff notation

The most common and well-recognized musical notation is modern staff notation, which is easily distinguished by 5 parallel horizontal lines with oval-looking notes placed on, between, below, or above them. Although similar approaches have been used before, the invasion of modern staff notation is assigned to Guido d'Arezzo - an Italian music theorist, who formulated his system around 1030 years A.C. [6] Throughout the years the notation has been modernized and modified until reaching its current state, widely used by musicians and composers to document music. The information about the composition's tone, sound, pitch, rhythm, etc is represented by various elements.

The main properties of music composition can be divided into **discrete** and **continuous**.

Continuous - is a type of property that remains constant for the whole composition without interruption or changes.

Discrete - is a type of property that can be defined for a certain part or period of time, it can change and evolve throughout the composition.

Some of the most important properties of music composition are **pitch**, **key**, **duration**, **timing** and **volume**. Let us review each of them individually and state what elements are used to hold this information in the notation according to Music Notation and Theory by Jono Kornfeld [9].

#### Pitch

Pitch is a position of a single sound in the complete range of sound. Sounds are higher or lower in pitch according to the frequency of vibration of the sound waves producing them [5].
The pitch is the discrete property as it changes with every new note played by the instrument and never stays continuous unless the whole music composition contains only one note/pitch.

In the modern staff notation a pitch is encoded by several elements, to be precise, by **a note** (sometimes with **accidental** before it) standing of one of the **staff** lines with a certain **clef**.

A note is an element that represents a sound, it holds information about the pitch and duration of the sound.

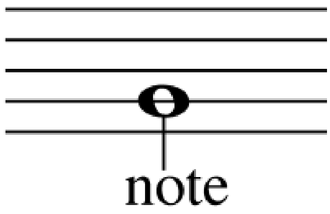


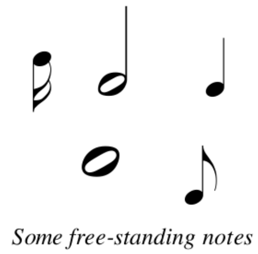

FIGURE 2.1: Single note [9]


FIGURE 2.2: Free-standing notes [9]

The notes are located on the staff(stave) - five horizontal lines.
Each note has its place on the staff, they can be placed on, between, above or below the lines. The placement of the note on the staff corresponds to pitch, the higher the note is on the staff, the higher the pitch.

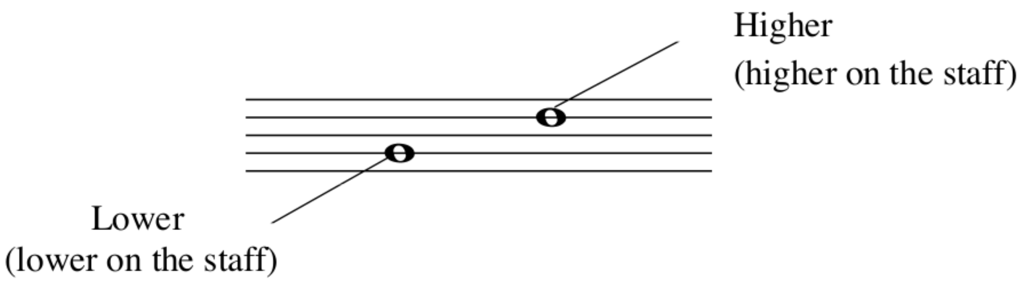



FIGURE 2.3: Notes on the staff [9]

At the beginning of the staff is located a clef - an element which indicates the specific diapason of sounds represented by notes on this staff, with that directly influencing the pitch of the notes on the staff. The three most common clefs are Treble, Bass, and Alto.

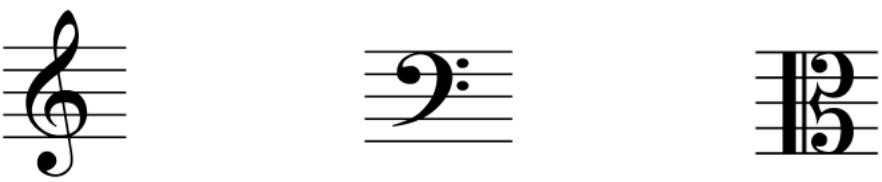



FIGURE 2.4: Treble, Bass and Alto clefs [9]

Treble clef diapason is 1-line octave Mi(E4) on the bottom line to 2 line octave Fa(F5) on the top line.
Bass clef diapason is a great octave Sol(G2) on the bottom line to a small octave La(A3) on the top line.
Alto clef diapason is small octave Fa(F3) on the bottom line to 1 line octave Sol(G4) on the top line.

One can notice that the notes on the Bass staff correspond to the lower, Alto staff to the middle, and Treble staff to the higher notes on the keyboard. For a better understanding here are the diapasons for the three represented clefs side by side.

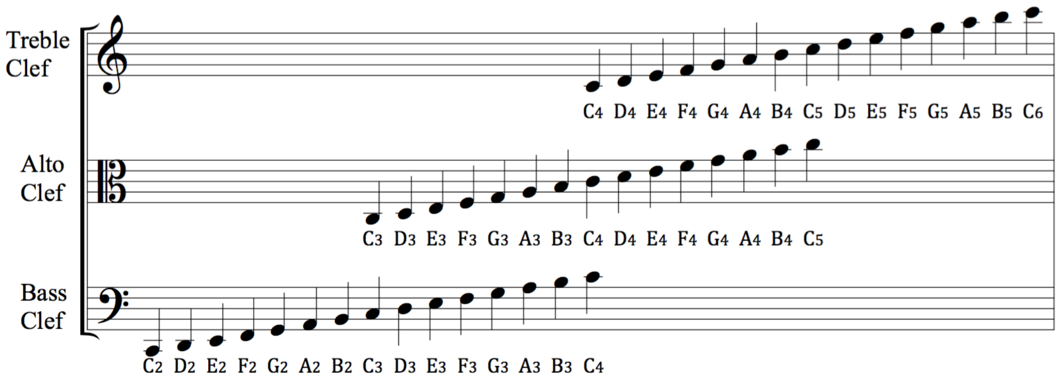



FIGURE 2.5: Notes on the staff with different clefs[22]

Occasionally the pitch of the note is altered by a special element called accidental. It raises or lowers the pitch of the note by a semitone(half step).
If we raise a note, we use a sharp sign, if we lower a note, we use a flat sign. To cancel or deactivate a previous sharp or flat, we use a natural sign.
Rarely the double accidental can be found, which lowers or raises the pitch of the note by the tone(a whole step).

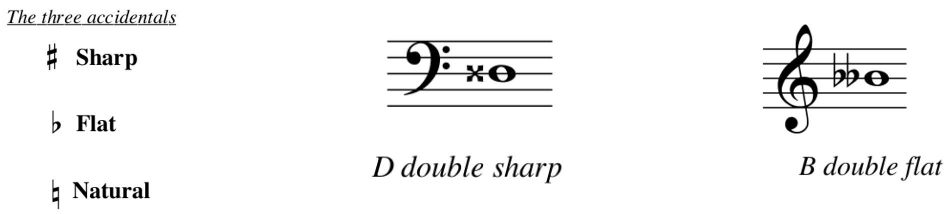



FIGURE 2.6: Singular and double accidentals [9]

**Key**

Key is the group of pitches, or a system of functionally related chords deriving from the major and minor scales, with a central note, called the tonic [4].
The key in most cases is the continuous property as it is defined at the start of the composition and remains the same continuously until the end. Seldom there are music scores where the key is discrete property when it can change a few times within a long composition.

There are 24 keys in music, 12 major and 12 minor, which are paired by the selection of common pitches used in both entries of a pair.
In modern staff notation, a pitch is encoded by an element called a **key signature**, represented by **accidentals** of a particular key.

A key signature is placed on each line of the staff after the clef, using accidentals to identify which notes will be raised or lowered to create a certain selection of pitched - a key.

FIGURE 2.7: Examples of key signatures [9]

**Duration**

Duration is the amount of time or how long or short a note, section, or music composition lasts, or in other words, - the length of time a pitch, or tone is sounded(*3.2). The duration is the discrete property as it changes with almost every new note played by the instrument and rarely stays continuous.

Modern staff notation holds the capability of describing the duration of a sound or a silence.

The duration of the sound is indicated by different **note values**, which are expressed as relative lengths to one another by a factor of two, as one whole note equals two half notes, one-half note equals two-quarter notes etc.

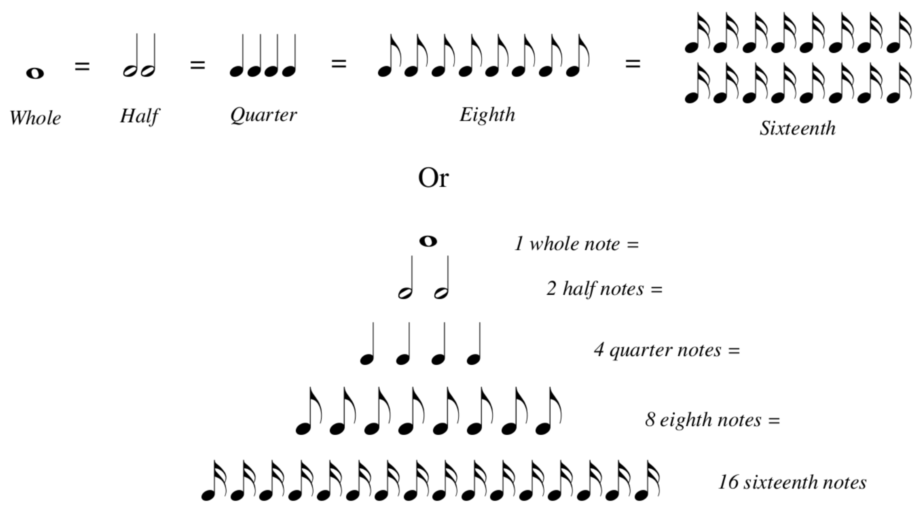



FIGURE 2.8: Note values relation [9]

If several notes of the eighth or sixteenth value are placed one after another they can be beamed together by their stems.

As the notes are indications of sounds, the rests are indicators of silence in the music composition. Same as note values there are identical rest values for different lengths and durations of the pause/silence.

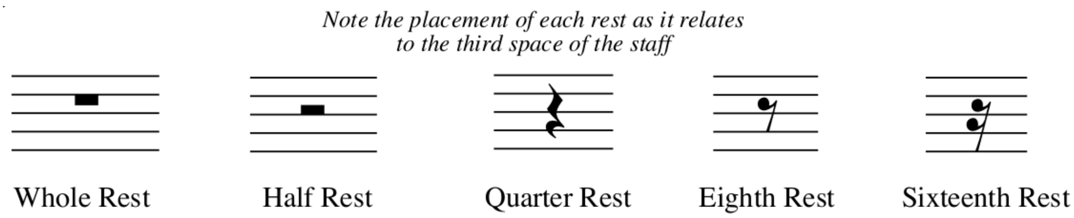



FIGURE 2.9: Rest values [9]

#### Timing

Timing in music refers to the ability to keep synchronized to an ensemble, as well as subtle adjustment of note or beat duration, or tempo [24].
Timing is one of the continuous properties as it is strictly defined at the beginning of the composition and defines one true timing measure for the whole music score.

Timing is the main property for describing music through and within time, a certain period of time count. It is also important for coordination and synchronizing the sounds of two hands that are playing the instrument together as well as two or more instruments with one another.

Information about the score's timing is encoded into **time signature**, **measures** and **bar lines**.

Music staff is divided into equal time duration parts called Measures or Bars.
It is used as an indication of the beginning and end of the time period set by the time signature and for a better understanding of certain points of time in music composition.
Measures are separated by vertical lines called Bar Lines. The final of the musical composition is indicated by Double Bar, pair of thin and thick lines at the very end of the music staff.

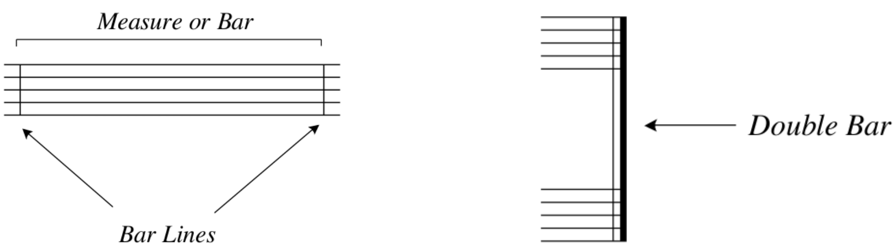



FIGURE 2.10: Bar, Bar lines and Double bar [9]

Time Signature is a timing element consisting of two numbers in the column, located at the beginning of the staff after the clef. It determines the beats in each measure, where the upper number indicates how many beats (or counts, or pulses) are in each measure and the lower number indicates which type of note value counts for one beat. This sets the rule for a composition, when(timing) and for how long(duration) certain sound or silence must be held.

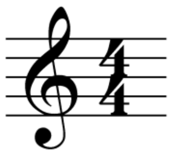



FIGURE 2.11: 4/4 time signature [9]

For example, in a 4/4 time signature, the quarter note counts for one beat and there are four beats per measure.
A lot of different time signatures are possible, but the most common ones are 2/4, 3/4, and 4/4.

**Volume**

Volume in music is the degree of loudness or the intensity of a sound [18]. In other words, how strong and loud the notes(pitches) must sound.
The duration is also a discrete property as it can change a few times through the composition. However in some small, easy music pieces, the volume can remain practically the same for a whole score, so in this case, it can be called relatively continuous.

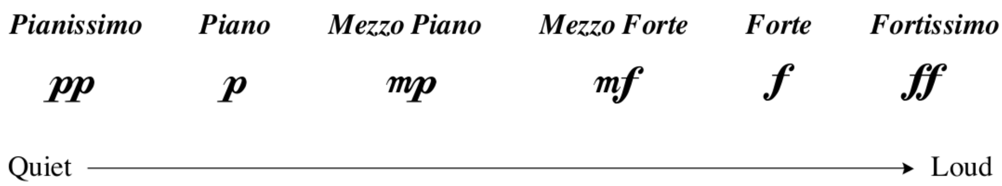
Dynamic is an element that is responsible for volume indication, it states how loud or quiet a certain part of music should be. A dynamic sign is placed under the staff at the beginning of the part, and it lasts until the occurrence of the next dynamic sign.
The quietest possible volume is Pianissimo and the loudest is Fortissimo.


FIGURE 2.12: The basic dynamic range [9]

These were the basic and main properties of music score described by elements of the modern staff notation system, however, there are more aspects of musical composition and elements used for its representation.

### 2.1.2 Tablature

Another widely-used musical notation is Tablature (or Tab), used mainly for guitar and harmonica, but also other string instruments and free-reed aerophones. According to Krenz [10] the difference from modern staff notation is that tablature is used to indicate instrument fingering (or air directing), which strings on the instrument are being played at any point and what frets need to be fingered.

Tablature same as modern staff notation also has a staff with the number of lines that equals the number of strings in the instrument. The numbers on the lines describe which frets the fingers needed to be put on and which strings to use.
So the main difference from modern staff notation is in encoding **pitch**, **timing** and **duration**. Rather than using notes for indicating pitch, tabs use numbers on the staff lines for every specific sound.
Timing and duration are also encoded slightly differently as tabs use numbers, their placements and spaces between these numbers to indicate the timing and duration of a specific pitch.

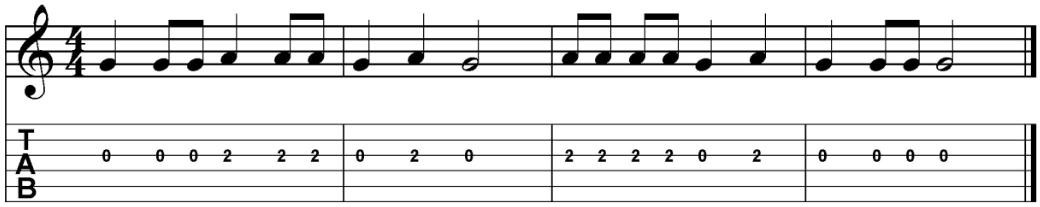



FIGURE 2.13: Example of lines and spaces as indication of pitch and duration in tabs [10]

### 2.1.3 Numbered musical notation (Chinese musical notation)

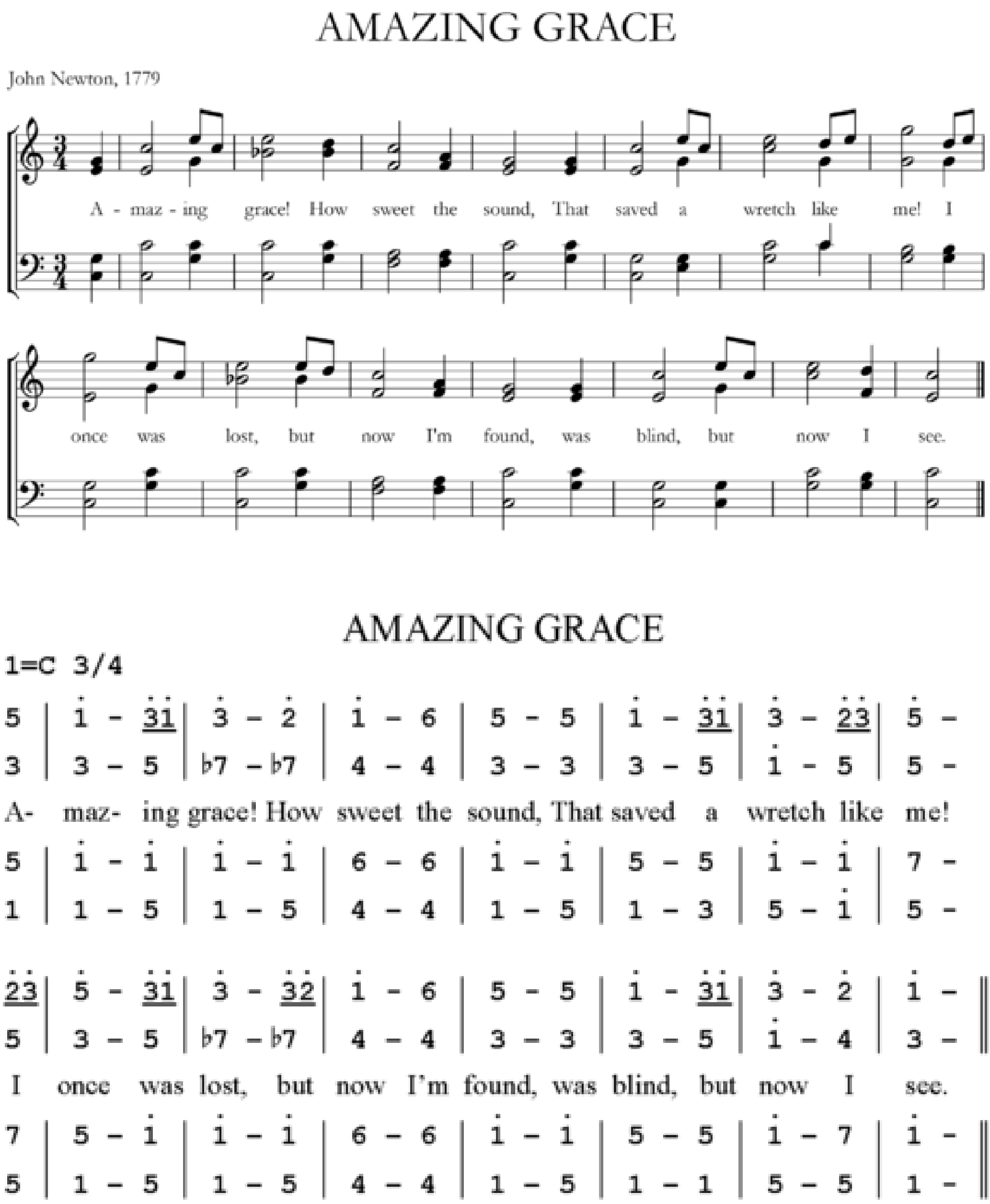
Numbered musical notation is a cypher notation widely used in China, and to some extent in Japan, Malaysia, Germany, etc [23].
The representation of almost all music properties is different from modern staff notation.
For example, for the pitch and key, this system is using numbers from 1 to 7 as a representative for seven notes in a chosen musical key. For example, in Do(C) major 1 is Do, 2 - Re, 3- Mi, 4 - Fa, 5 - Sol, 6 - La, and 7 - Si.
Also, this notation uses different elements such as the "X" symbol, dots, plain lines, and underlines of different durations and placement to represent octaves, note duration, rests, undetermined pitch, etc.


FIGURE 2.14: The example of numbered notation in comparison to modern staff [23]

# Chapter 3

# Related works

Conventional music notations are executing the task of effectively storing and presenting information about a variety of different elements perfectly well. It, indeed, is an easy way for a musician to quickly write down, keep, or read the musical piece with the most accuracy to the original sound produced by the instrument. However, these systems are surely not the only way to represent music, consequently, with the rise of the technology era, new methods and tactics for music visualization were developed.

## 3.1 Music Animation Machine by Stephen Malinowski

One of the examples among the existing work in the area of music visualization is an animated graphical score project called Music Animation Machine created by musician, inventor, and software engineer Stephen Malinowski [11]. It contains of music animation machine website [13] - all the information and documentation about the project, music animation machine software [15] - software used for creating the visualizations, and a music animation machine YouTube channel [17] - the portal for showing and sharing the visualizations.

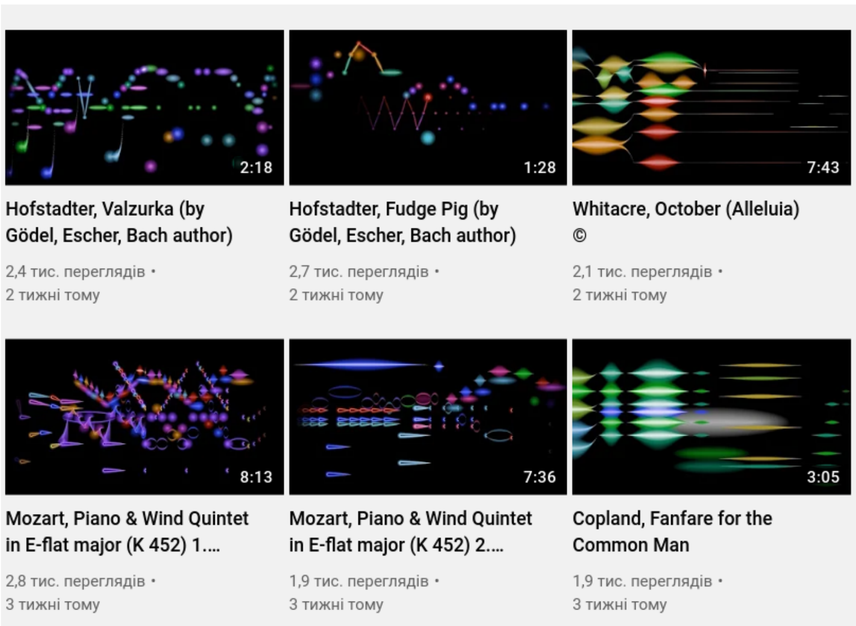



FIGURE 3.1: Videos on Stephen Malinowski's channel [17]

As Nick Romeo of The Daily Beast says in his article [21] about Malinowski and his project:

*"His software translates the primary formal elements of music—pitch, rhythm, dynamics, harmony, timbre, etc.—into a visual language of interconnected lines, colours, and shapes. Each visual feature models a particular element of the music; the size and position of the shapes model rhythm and pitch, for instance, which helps make the structure of even dazzlingly complex compositions intuitively apparent."*

## 3.2 Input Data

The Music Animation Machine MIDI Player - later, MAM Player - uses a Standart MIDI file to generate various kinds of graphical displays in response to (and in sync with) the MIDI data (Musical Instrument Digital Interface) as it plays. It will also produce these displays in response to live MIDI input [12].

The audio and video are processed separately. The edited performance is played back through a synthesizer (or sampler) and its output is recorded. The edited performance is also processed to produce individual video frames. The audio and video are then imported into video editing software and exported as a movie [14].

## 3.3 Information encoding methods

Stephen Malinowski in his MAM uses different ways and methods to encode units of information and different music properties.
For instance, he uses:

- **Colour** for encoding information about several melodic lines, playing hands or playing instruments, interval types, dynamics or rhythmic subdivision classes.
- **Connecting lines/notes** for encoding melody connections, extended canons, melodic motives, show repeating bass.
- **Note thickness** - dynamics.
- **Shape** - different pitches, different melodic elements, instrumental groups.
- **Size** - duration.

More detailed information about encoding can be found in MAM Techniques page, where the list of all approaches together with specific methods and meanings behind the result is located [16].

One example of the use of Malinowski's Music Animation Machine MIDI Player is the visualization of Beethoven's Fifth Symphony created by Kent Bye [7].
He uses three different display types to create this illustrating video representation of the piece, each of them encoding specific musical properties and elements.

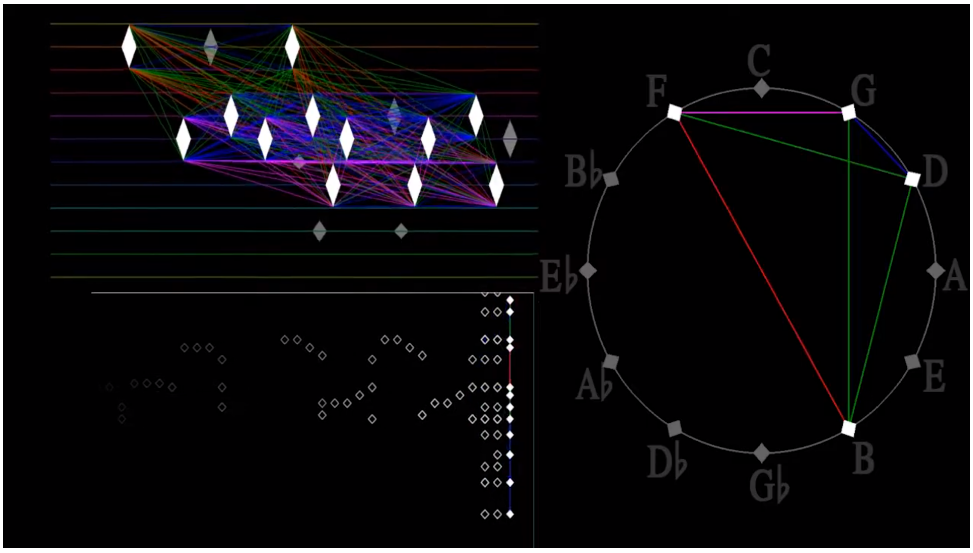



FIGURE 3.2: Visualization of Beethoven's Fifth Symphony [7]

On the right side is placed a coloured wheel display (DYAD) of Circle of Fifth[1]. It is responsible for showcasing the **intervals** and **tonality** in this arrangement.
Different intervals are displayed in different colours: blue for fifth, fourth and tonic octaves; green - thirds and sixths; violet - major seconds and minor sevenths; yellow - minor seconds and major sevenths; red - tritone.
Types of chords can be distinguished by the geometry of the triangle formed by "interval" lines. For example, major chords are large obtuse triangles with the Perfect Fifth 30° clockwise from the root and the Major Third 120° counterclockwise.

On the higher left side is placed a Tonality Staff (V12). It is similar to the DYAD display, except the tonalities classes are shown on the lines, not in the circle, which are moving to keep them centred. The notes are also displayed according to their pitch, the left is lower and the right is higher. Additionally, the notes are connected by coloured lines, displaying intervals between them.

On the lower left side is placed a display for **intervals** and **pitch** (YARN). It displays all the notes being played, located lower or higher on the vertical axis according to their pitch. The coloured lines connect the played notes, to show the sounded intervals through the composition.

More detailed information about this visualization can be found in the description of the video [7].

## 3.4 Approach's restrictions and limitations

The discussed approach does an excellent job of representing the music score and a lot of its music properties. Especially well displayed and easy to understand are the visual representation of pitches and durations as well as the presence of different

[1]Circle of Fifth - is a way of organizing the 12 chromatic pitches in a sequence of perfect fifths, generally shown as a circle with the pitches (and their corresponding keys) in a clockwise progression.

melodic lines. However, in my opinion, many researched visualizations lack the display information about the volumes and dynamics. On the occasion when volume property is encoded by some method it is not as easily comprehended and recognized as, for example, durations or pitches.

Also, most of the display types and display options presented in MAM Player are oriented for the representation of only one or two music properties. While it is a good practice to not mix different types of information between one another and not create an information noise, it also means that from a given visualization the data about only one property can be consumed. Relying on that, it is necessary to make a few different visual representations of the same music score to fully get the grasp of some information in different aspects (for example, not only information about duration and intervals but also timing, melodic lines, instruments etc).

Malinowski's MAM visualization result is animation, a video representation that lasts for the same amount of time as the original music composition. It is tightly tied to data temporality, displaying and visualizing music and sounds at a specific moment in time, and then joining all these static visualizations together into continuous animation. This approach is great as the following visualization, which is consumed together with the composition's audio, for a better understanding of the sounds a person is hearing and generating displays in real-time from live performances.
However, the approach of animating visualization does not satisfy the goals of this project, as a complete visual representation of the whole music score together as one and not the combination of visualization at every specific moment of a given score. A better solution for described purposes would be a static visualization.

# Chapter 4

# Data obtaining and transformation

For this project, we will be working with the MIDI data format of music scores. It can be found on the Internet, generated and saved through MIDI input, or generated from sheet music and musical notations described in Chapter 2.
To produce and obtain the MIDI file from music sheets of the score, we will be using the software called MuseScore 3 [19] - a music notation scorewriter, that allows users to write, edit, import and export music compositions using modern staff notation. After importing music sheets into MuseScore, and editing some elements if necessary, we export the score in a MIDI format file.

## 4.1 mido package

To work with obtained MIDI data we will be using a special Python package for working with MIDI messages and ports called mido [20]. The music score is represented as an instance of an python object MidiFile:

```
MidiFile(type=1, ticks_per_beat=480, tracks=[
  MidiTrack([
    MetaMessage('time_signature', numerator=3, denominator=4,
    clocks_per_click=24, notated_32nd_notes_per_beat=8, time=0),
    MetaMessage('key_signature', key='C', time=0),
    MetaMessage('set_tempo', tempo=500000, time=0),
    Message('control_change', channel=0, control=121, value=0, time=0),
    Message('program_change', channel=0, program=0, time=0),
    Message('control_change', channel=0, control=7, value=99, time=0),
    MetaMessage('midi_port', port=0, time=0),
    Message('note_on', channel=0, note=67, velocity=50, time=0),
    Message('note_on', channel=0, note=67, velocity=0, time=719),
    MetaMessage('end_of_track', time=1)]),
```

From this object, specifically from MetaMessages, all the needed information about music properties and composition will be taken.

#### Time Signature

```
MetaMessage('time_signature', numerator=3, denominator=4,
   clocks_per_click=24, notated_32nd_notes_per_beat=8, time=0)
```

Numerator and denominator are upper and lower values of key signature, notated_32nd_notes_per_beat is the number of 32nd notes per one beat, clocks_per_click is the number of MIDI clock ticks per quarter of note.

#### Key Signature

```
MetaMessage('key_signature', key='C', time=0)
```

Key holds an information about score's key.

#### Tempo

```
MetaMessage('set_tempo', tempo=500000, time=0)
```

Tempo holds information in microseconds per beat (quarter note). One can use tempo2bpm() to convert to beats per minute.

#### Pitch & Velocity (Volume)

```
Message('note_on', channel=0, note=55, velocity=50, time=73)
Message('note_off', channel=0, note=84, velocity=0, time=4)
```

Note value determines the pitch, it goes from 0 to 127, with the middle Do(C) note being represented by the value of 60. The velocity value goes from 0 to 127, covering the volume range from Pianissimo to Fortissimo, discussed earlier in Chapter 2 [2]. The time value represents a time in delta time when a specific note(pitch) is played or stopped.

#### Instrument

```
MetaMessage('track_name', name='Flute', time=0)
```

It stores a track name which often is the same as the instrument or voice name.

#### End

```
MetaMessage('end_of_track', time=1)
```

This message marks the end of the track.

## 4.2 pretty_midi package

Another Python package that will be used in this project to work with MIDI data is pretty_midi [8]. Needed information is stored and can be obtained from the following objects:

#### Time Signature

```
TimeSignature(numerator=3, denominator=4, time=0.0)
```

Numerator and denominator are upper and lower values of key signature.

#### Key Signature

```
KeySignature(key_number=2, time=0.0)
```

Key holds information about the score's key, Key number according to [0, 11] Major, [12, 23] minor. For example, 0 is Do(C) Major, and 12 is Do(C) minor.

#### Tempo

```
(array([0.]), array([120.]))
```

Tempo holds information as two arrays, first - times when tempo changes, second - BPMs of the score.

#### Pitch & Velocity (Volume) & Duration

```
Note(start=3.000000, end=3.127083, pitch=81, velocity=69)
```

Pitch value determines the pitch, it goes from 0 to 127, with the middle Do(C) note being represented by the value of 60. The velocity value goes from 1 to 127, covering the volume range from Pianissimo to Fortissimo, discussed earlier in Chapter 2 [2]. The start and end values represent a time in seconds when a specific note(pitch) is played or stopped.

#### Instrument

```
Instrument(program=73, is_drum=False, name="Flute")
```

It stores an instrument name as a name parameter, and MIDI program number as a program parameter.

### 4.2.1 Additional functions

#### get_beats(start_time=0.0)

Returns a list of beat locations, according to MIDI tempo changes. This method returns every denominator notes time in seconds (every quarter note for 3/4 or 4/4 time).
For example:

```
[0.  0.5 1.  1.5 2.  2.5 3.  3.5]
```

#### get_piano_roll(fs=100, times=None, pedal_threshold=64)

Compute a piano roll matrix of this instrument. The function returns an array of 128 arrays - corresponding to 128 notes on a piano roll, each array represents a note throughout the whole composition.
For example:

```
[0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 80.0 80.0 80.0 80.0 80.0 80.0 80.0 80.0 80.0 80.0 80.0 80.0
80.0 80.0 80.0 80.0 80.0 80.0 80.0 80.0 80.0 80.0 80.0 80.0 80.0 80.0
80.0 80.0 80.0 80.0 80.0 80.0 80.0 80.0 80.0 80.0 80.0 80.0 80.0 80.0
80.0 80.0 80.0 80.0 80.0 80.0 80.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0]
```

## 4.3 Lost or hard-to-reach information

Although a MIDI format is very well structured and easy to obtain needed information with Python, there are a few pieces of data about elements of the music score that are lost or quite hard to reach or calculate.

### 4.3.1 Accidentals

In MIDI data information about accidentals is not represented on their own, meaning there are no specific MIDIMessages for stating a sharp of flat. However, accidentals influence on the pitch is not lost, just encoded within Notes messages and Key messages.
For better understanding, if we have a music composition of a Do major key, and on one occasion there is a sharp accidental on the Do note of 1 line octave, it will be represented as:

```
MetaMessage('key_signature', key='C', time=0)
Message('note_on', channel=0, note=61, velocity=50, time=73)
```

As Do on 1 line octave is 60 and Re on 1 line octave is 62, Do sharp is 61 (same as Re flat), as it is stated on the Note message. And looking into the Key message, we can obtain information that the key is Do major, and it has no sharps or flats as it represents, meaning if 61 note occurs in this composition - it is accidental.

### 4.3.2 Articulations (Legato/Staccato/Glissando etc)

Similar to accidentals, information about articulations of the music score is lost, but its effect on the sounds of notes is encoded into different properties. Legato, staccato, etc signify a note of shortened/longed duration, a passage from one note to another, a louder note etc. All these effects are encoded in other existing properties, such as note duration, note velocity and others, that is why there is no need for a separate MIDI Message to store information about articulations.

### 4.3.3 Note values

As the MIDI format is often used for writing down and working with live input, it does not have a property that states a certain note value, as when playing live one cannot tell in advance what value a note will have. However, form note_on and note_off messages that contain information about ticks of a specific note, it is possible to obtain information in seconds about how long a note is played. In a consequence of that, the information about specific note values is not reachable, however, similarly to previous elements, its influence on the music score remains encoded by other properties.

## 4.4 Data preparation and transformation

Data that is obtained from the MIDI format is raw and in need of selection and transformation before creating visualizations with it. Different units of data, sorted in a certain order are needed for different types of visualization, but the algorithm and logics behind preparation remain the same.

Information about units of data such as key signature, time signature, and tempo are extracted and encoded into visualization separately. While pieces of information about notes and their properties are obtained for each note, together forming an array that holds data about certain properties. Therefore after given preparation, a number of arrays are formed, and each of them holds information about certain properties (such as pitch, volume, time, duration, and colour). Data about a specific note is stored under the same index in all the arrays.

For better understanding, here is an extract of transformed data of the “Happy Birthday” song:

```
[60, 60, 62, 60, 65, 64, 60, 60, 62, 60],
[65, 71, 79, 67, 85, 75, 73, 73, 80, 71],
[0.30000000000000004, 0.30000000000000004, 0.6000000000000001,
    0.5999999999999996, 0.6000000000000001, 1.1999999999999997,
    0.3000000000000007, 0.2999999999999998, 0.5999999999999996,
    0.5999999999999996],
[[0.9803921568627451, 0.5372549019607843, 0.03529411764705882,
    0.16666666666666663], [0.9803921568627451, 0.5372549019607843,
    0.03529411764705882, 0.16666666666666663], [0.9803921568627451,
    0.5372549019607843, 0.03529411764705882, 0.33333333333333326],
    [0.9803921568627451, 0.5372549019607843, 0.03529411764705882,
    0.333333333333333], [0.9803921568627451, 0.5372549019607843,
    0.03529411764705882, 0.33333333333333326], [0.9803921568627451,
    0.5372549019607843, 0.03529411764705882, 0.6666666666666663],
    [0.9803921568627451, 0.5372549019607843, 0.03529411764705882,
    0.16666666666667], [0.9803921568627451, 0.5372549019607843,
    0.03529411764705882, 0.1666666666666665], [0.9803921568627451,
    0.5372549019607843, 0.03529411764705882, 0.333333333333333],
    [0.9803921568627451, 0.5372549019607843, 0.03529411764705882,
    0.333333333333333]]
```

Arrays above hold data about pitches, time, volumes, durations, and colours respectively. First values on the arrays hold information about one note, second - about another and so on.
Values of each note colour are chosen based on the score’s key and the opacity of chosen colour corresponds to one of the properties (for example, duration or volume).

# Chapter 5

# Data encoding

## 5.1 Available data

Before describing the possible encoding for each property of music, the unit of available information, it is important to indicate how each is represented, in what measurements, and the range of values.

### 5.1.1 Data about a whole score & distinct parts of the score

For a better and quicker overview of different units of data, the following table was formed.

| **Property** | **Measure** | **Range** | **Coverage** | **Data Type** |
|---|---|---|---|---|
| Key | String or Integer | C-F or 0-23 | General | Continuous |
| Instrument | String or Integer | 0-128 | General | Continuous |
| Tempo | BPM | 0-? | Timing | Continuous |
| Beat Duration | Seconds | 0-? | Timing | Continuous |
| Time Signature | Integer values | D. are even | Timing | Continuous |
| Note (Pitch) | Integer | 0-127 | Specific note | Discrete |
| Velocity (Volume) | Integer | 0-127 | Specific note | Discrete |
| Duration | Seconds | 0-? | Specific note | Discrete |

TABLE 5.1: Overview of music properties data

For the Pitch, Volume and Duration their data type is defined as discrete throughout the score, but can be considered continuous if examined in relation to specific note.

In addition the good practice will be to divide Time Signatures into groups, for better and easier classification:
- Common time and other signatures that “feel” very similar (4/4, 2/2, 8/8)
- 2-beat (2/4, 2/8, 2/16)
- 3-beat (3/2, 3/4, 3/8, 3/16)
- 4-beat (4/2, 4/8, 4/16)
- 6-beat (6/2, 6/4, 6/8, 6/16)
- Others (5/4, 7/8, 21/8, 15/16, 11/16)

## 5.2 Possible encodings

### 5.2.1 Key

Keys can determine the colour palette of the score or score parts if there are several keys in the composition. The minor keys can correspond to a cold-coloured palette and the major key - to a warm-coloured palette.

### 5.2.2 Instrument

Instruments values can correspond to different colours within a colour palette of the key. If necessary one instrument can also correspond to not only one colour, but different shades of that colour. For example, music score has Do Minor as a key, so the palette is cold, and two instruments - piano and flute, which are encoded as shades of green and blue respectfully.

Different instruments can also be encoded as:

- different shapes. Piano - quadrangle, flute - circle, etc.
- through different areas on the visualization that does not intersect with each other. Piano as the upper part, the upper line; flute - as lower, etc.
- different sizes or different thicknesses if it's a line. Piano - a big circle, flute - a small circle, etc.

### 5.2.3 Notes/Pitches

Piano roll with all the pitches can be encoded as a value on the line from 0 to 127, which for some visualizations can be presented as an X or Y axis, as well as a line folded into a circle or a ray on semi-circular/circular graph.

### 5.2.4 Volume

Volume can be encoded into the opacity of the colour. For example, the piano notes are green coloured, so each note's volume will determine the opacity of the green for this specific note.

It can also be encoded into:

- different sizes of notes(depending on how notes will be encoded)
- the intensity of shadow behind notes(depending on how notes will be encoded)
- heights of the notes. If the visualization is similar to the kind of bar chart of any other chart that has a vertical axis, the volume can correspond to different heights of the element, the bigger the volume the higher the element.

### 5.2.5 Duration

Duration can be encoded into different lengths, the longer the duration of the note, the bigger the length.

It can also be encoded into:

- the opacity of the colour, the longer the note, the bigger opacity its colour has
- different sizes of notes(depending on how notes will be encoded)

### 5.2.6 Tempo

In the tempo/duration oriented visualizations tempo can be encoded into the darkness of the colour. As scores with bigger BPM are quicker and usually feel lighter, they can be represented by a lighter colour to correspond to that feeling. Identically smaller BPM feels heavier and can be represented by darker shades of colours.
It can also be encoded into:

- smoothness, the edginess of lines, the slower the tempo, the more smooth the lines transitions from one dot to another are

The same characteristics can be applied to beat durations and time signatures, as all of them contain information about the timing of the score.

Summarizing all the possible encodings stated above, the following table was formed, for a better and quicker understanding.

| **Property Encoding** | **Key** | **Instrument** | **Pitch** | **Volume** | **Duration** | **Tempo /Beats** |
|---|---|---|---|---|---|---|
| Colour Theme | + | | | | | |
| Colour /Opacity | | + | + | + | + | + |
| Shape | | + | | | | |
| Part | | + | + | | | |
| Size | | + | | + | + | |
| Position | | | + | | | |
| Shadow | | | | + | | |
| Length /Height | | | | + | + | |
| Smoothness of lines | | | | | + | + |

TABLE 5.2: Properties encoding

# Chapter 6

# Data visualization

## 6.1 Selection of music compositions

Music compositions exist in a wide variety therefore, a great algorithm must be able to work at least with several different types of scores. Hence it is considered a good practice to choose a few works of different complexity as input data for the approaches, to test and show the flexibility and variability of the algorithms.

For this project the following music compositions were chosen:

1. Air on the G String (Suite No. 3, BWV 1068) - Johann Sebastian Bach.
   For one instrument - piano, in major.
2. Toccata and Fugue in D minor, BWV 565 - Johann Sebastian Bach.
   For one instrument - organ or piano, in minor.
3. Bolero from 11 Pezzi Infantili, Op.35 - Alfredo Casella.
   For two instruments - flute and violoncello, in major.
4. The Four Seasons - Summer (3rd Movement) - Vivaldi.
   For five instruments - solo violin, violin 1, violin 2, viola and violoncello.
5. The Marriage of Figaro (Overture), K 492 - Wolfgang Amadeus Mozart.
   For orchestra - flute, oboe, clarinet in A, bassoon, horn in D, trumpet in D, timpani, violin 1, violin 2, viola, violoncello, contrabass. In major.
6. Symphony No.40 (1st movement), K 550 - Wolfgang Amadeus Mozart.
   For orchestra - flute, oboe, clarinet in B flat, bassoon, horn 1, horn 2, violin 1, violin 2, viola 1, viola 2, violoncello, double bass. In minor.

## 6.2 Colour encoding

The choice of colour is the crucial thing for an accurate visualization. In this project, the same logic and algorithm for encoding colour are used in all approaches.
The colour palette is chosen depending on the key of the composition. If the score is in a minor key - a cool colours palette will be used for visualization, if it is in major - a warm palette will be used. Later different musical instruments are encoded into different colours as well.
Shades for both palettes were carefully hand-picked and tested for perception and compatibility with one another inside the palette.

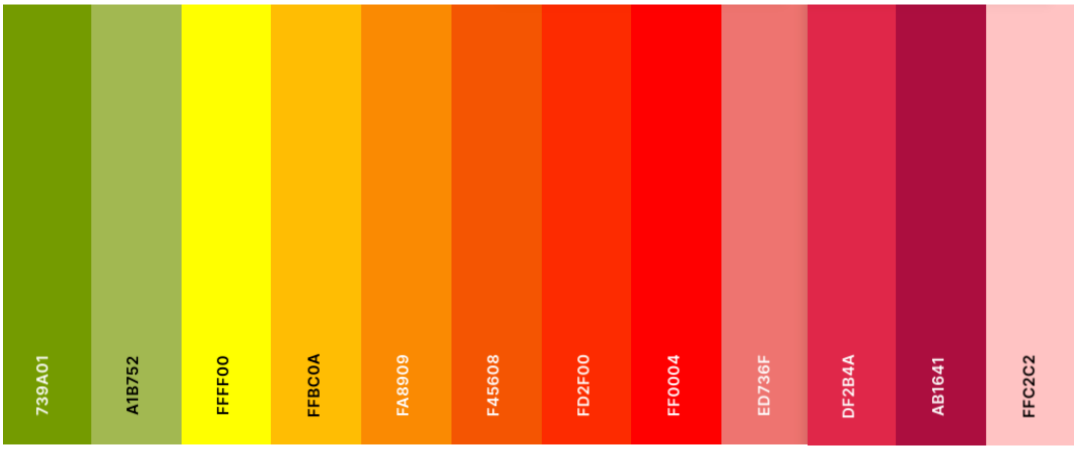



FIGURE 6.1: Warm Colour Palette

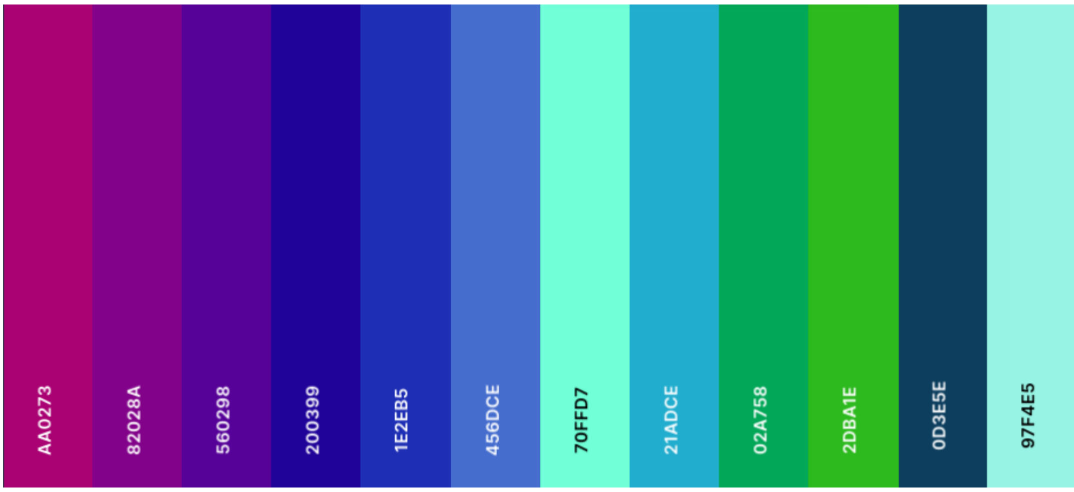


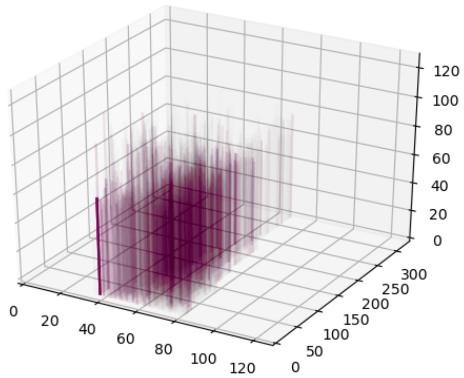

FIGURE 6.2: Cool Colour Palette

## 6.3 Approaches

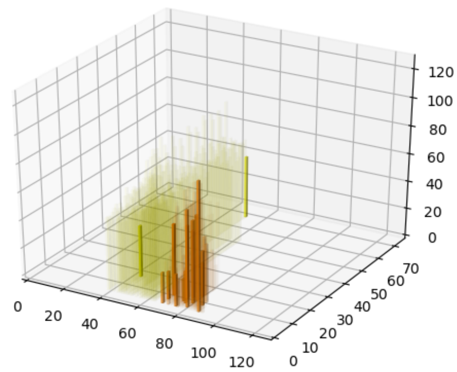
For better understanding and envisioning of data and possible results, it is important to try and explore a few different approaches and study their advantages and disadvantages to obtain the best possible visualization at the end.

### 6.3.1 Approach 1 - Notes popularity chart (6.3)

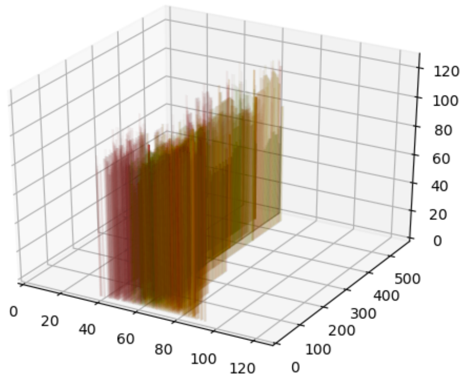
This approach is a kind of 3D "popularity chart" for notes in the score, it is focused on showing a relation between a placement of pitch on the piano roll and the number of times it appears in the composition.
The X-axis is a representation of the piano roll from the lowest note to the highest, encoded from 0 to 127 respectively. The Y-axis is a representation of amount, its range depends on the specifics of the score, the maximum value of an axis equals the maximum number of note appearances plus 10. And Z-axis representation has two different variations, in first it stands for the duration of the note, and second - for the volume of the note.
Depending on Z-axis variation, volume or duration is encoded into the opacity of the colour of the notes, the longer or louder the note is, the stronger is its representative bar's colour.



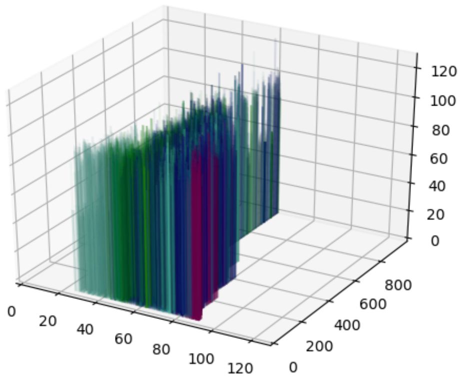

(A) Toccata and Fugue




(B) Bolero




(C) Summer




(D) Symphony No.40

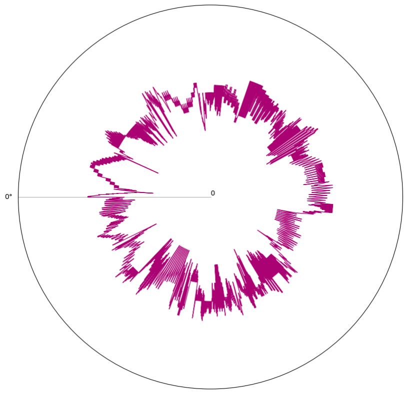
FIGURE 6.3: 3D notes popularity chart

### 6.3.2 Approach 2 - Circular chart (6.4)

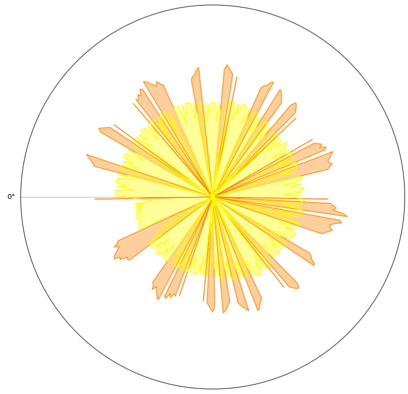
This approach is focused on specifying pitches and their timing. The axis that rotates clockwise represents specific moments during the music score. While the rays represent a piano roll, where points closer to the centre of the circle stand for lower notes on the piano roll and point closer to the circle-edge stand for higher notes. The start of the score is in the middle of the circle on the left side and it is invisibly connected to the end of the score in the same place.

### 6.3.3 Approach 3 - Notes' duration (6.5)

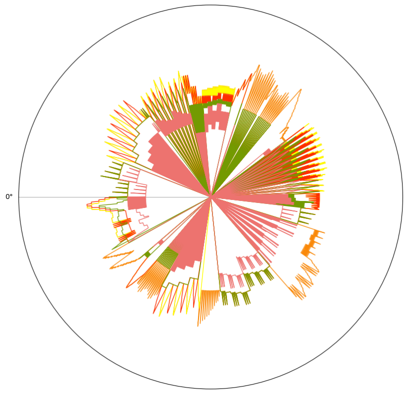
The X-axis represents the duration of one bar in seconds (for example, 0-2 seconds), however, if longer notes appear in the score, then the value of this note's duration is taken as the maximum of X. Y-axis represents the number of times this specific duration occurs in the music composition. Vertical lines are lines indicating the end of the bit duration (for example, if the time signature is 4/4, then there are 4 bits, that end at 0.5, 1, 1.5, and 2 seconds, respectively). This is important for linking the duration in seconds to the duration in bits, to better understand that, for instance, a duration of 1.78 seconds is between 3 and 4 bits in duration.



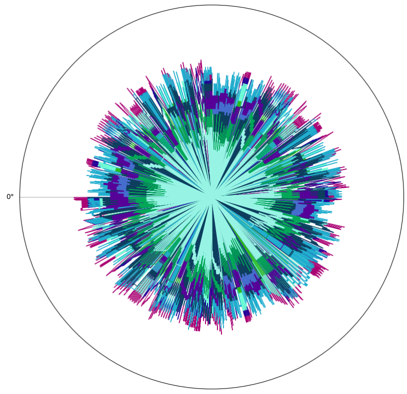
(A) Toccata and Fugue



(B) Bolero

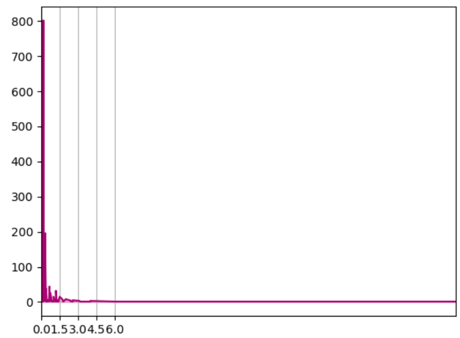


(C) Summer

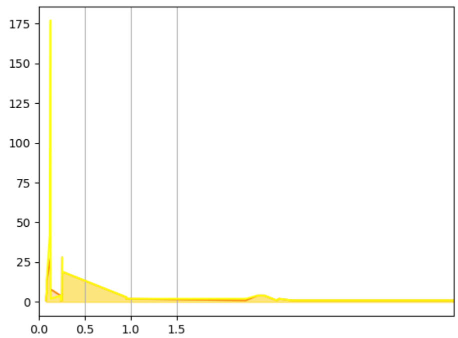


(D) Symphony No.40

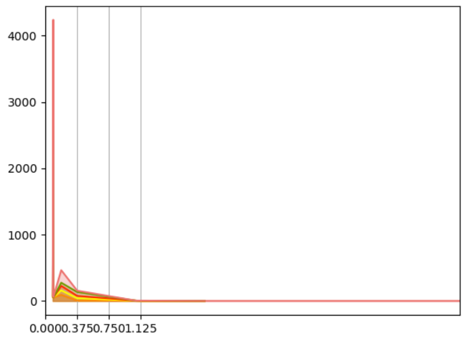
FIGURE 6.4: Circular chart



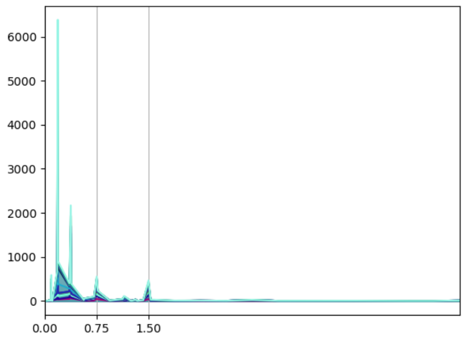

(A) Toccata and Fugue




(B) Bolero




(C) Summer




(D) Symphony No.40

FIGURE 6.5: Notes' duration line chart

### 6.3.4 Chosen approaches

While working and visualizing different units of data, it can be seen that although some properties can be vastly important and keep prime information about the score, not all of them are suitable for this kind of visualization.
The first approach has a distinct and relevant meaning and logic behind it, but 3D pieces are not adequate for the visualizations oriented for music composition covers. As for the third approach - the used data and output results are not diverse and look rather blank and empty, with a lack of engagement for potential viewers.

Therefore the following development of this work will be focused on pitch specification, temporality and note distribution, which will be portrayed as further expansions and alterations on the circular chart approach and a 2D remake of the popularity chart.

# Chapter 7

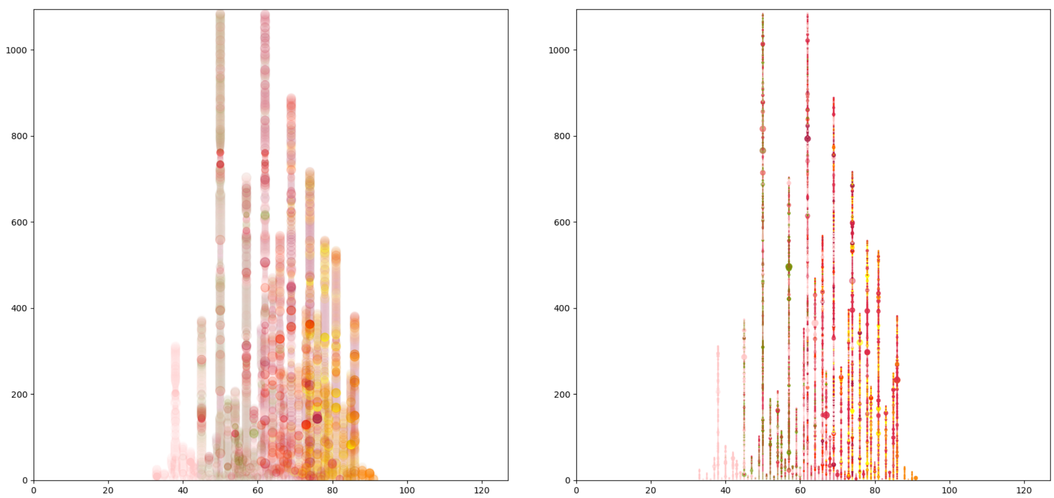
# Visualizations improvement

After choosing suitable approaches for final visualizations they should be worked on, improved and elaborated to produce the best possible results for score representations.

## 7.1 Improvements

### 7.1.1 3D to 2D

As was stated in the previous chapter, 3D visualizations are not adequate for music covers, thus they should be transformed into 2D for more proper results.
All the encoding, of the X-axis, Y-axis, colours and opacity will remain the same. Only the encoding of the Z-axis will be replaced by the encoding of the note's circle size. For the duration variation of this approach, the opacity of the circle's colour depends on the duration of the note and its size depends on the volume of the note. While in the volume variation - opacity depends on volume, size - on duration.



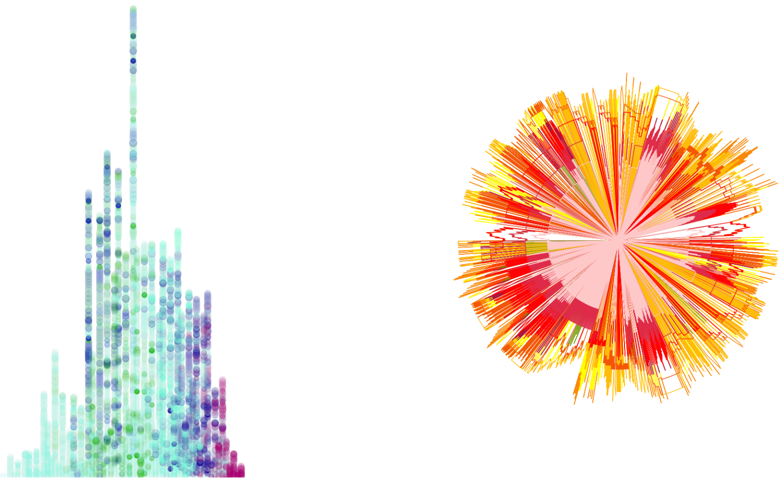

(A) Duration focus

(B) Volume focus

FIGURE 7.1: 2D notes popularity chart of "The Marriage of Figaro"

### 7.1.2 Markup removal

As the aim of creating these visualizations is to use them as music score representations for common music listeners and not scientific observation and research, all the unnecessary elements should be removed. They help comprehend and study the meaning and logic behind the visualization, but are not appealing or engaging to the potential target audience. All the markups: axes, ticks, labels, lines etc, should be abolished from the final results.



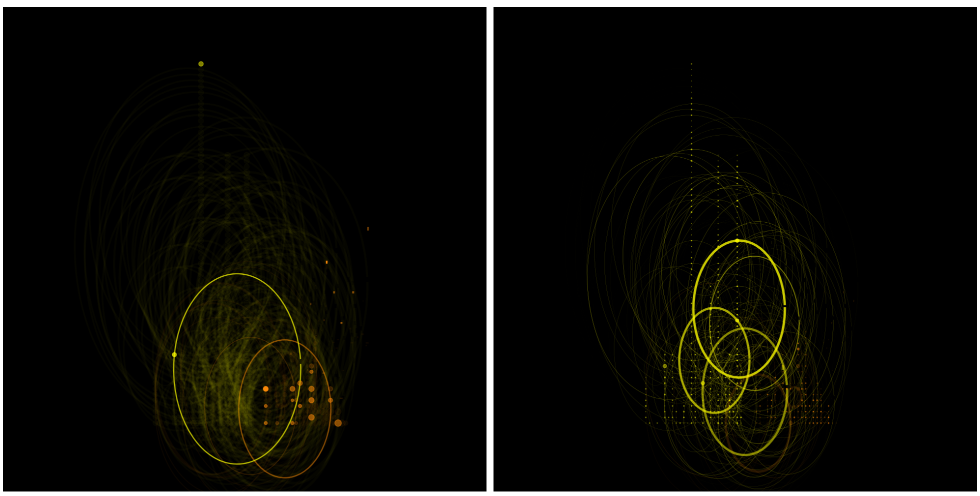
(A) Symphony No.40 (B) The Marriage of Figaro

FIGURE 7.2: Visualizations without axes

## 7.2 Alterations

### 7.2.1 Arc addition

The popularity chart as it looks at the current stage of development is to some extent blank and monotonous. To improve this visualization and make it more interesting and appealing to observe, the idea of adding arcs has arisen.
Already existing circles which represent notes will be connected with one another in a manner so that two notes which are played one after another will be connected. Each note, excluding the first and the last one, will be joined with the one that was played prior to and subsequent to it. The styling of the arcs will be the same as the styling of the circles, opacity and size of it correspond to volume and duration.

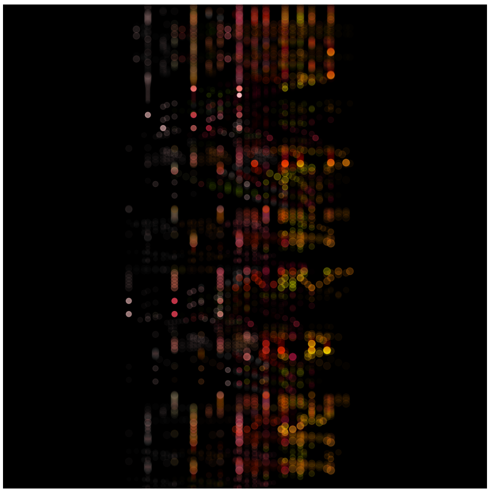


(A) Duration focus (B) Volume focus

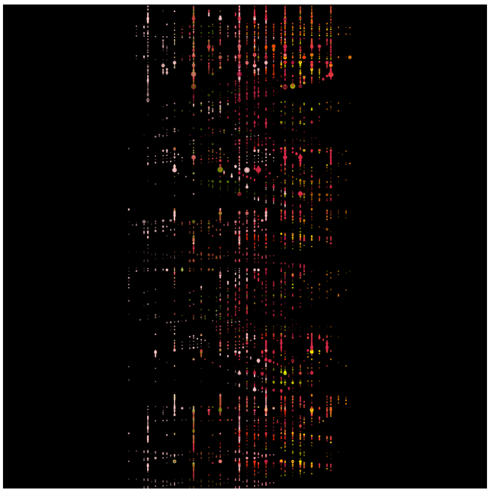
FIGURE 7.3: Arc plot of Bolero

### 7.2.2 Temporality

Another alteration that can be performed on the popularity chart is slightly shifting its focus from notes distribution to notes' timing and temporality. Thus encoding of the X-axis, opacity and colours will remain unchanged, but Y-axis now will hold information about the moment in time, when a specific note was played. Using described approach the viewers will be able to observe the timing of the score and the movement of the melodies throughout the score.



(A) Duration focus (Bolero)



(B) Volume focus (Bolero)

Furthermore, for a more appealing and pleasing visualization, the volume and duration variations can be joined together to create one piece with more information encoded into it and a more engaging picture.

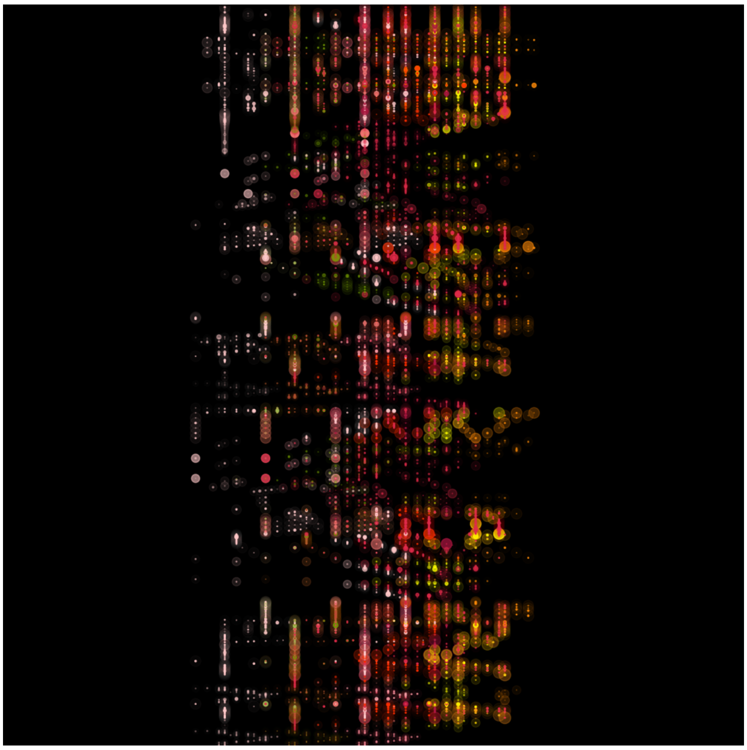
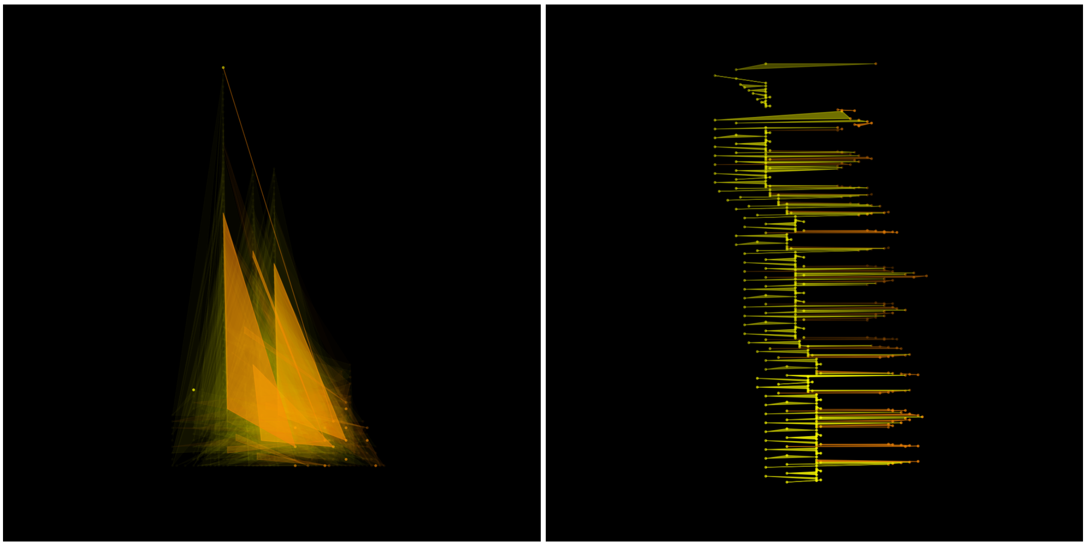


FIGURE 7.5: Temporality plot

### 7.2.3 Alternative linking

The point of this variation of the visualization is to try alternative linking of the notes. Previously the arcs were used, while in this case, the connection between the notes will be done by triangles. The points are divided into groups of three, depending on the time they are played in the score. Each point is serving as the vertex of the triangle. The colour of the triangle's edges and filling depends on the instrument, while opacity depends on the note's volume or duration.

Working with an offered approach, two variations are possible: first - focusing on the distribution and popularity of the notes, and second - on time and temporality.



(A) Distribution focus

(B) Temporality focus

FIGURE 7.6: Triangular plot of "Bolero"

## 7.3 Design & styling

Although the process of developing visualization is nearly finished, produced results themselves are not yet sufficient and qualified to use as music covers. Design and styling, title and author name additions, and background variations are crucial details to consider, for the work to be adequate to put forward on streaming services.

### 7.3.1 Background colour

Whereas the colours of visualizations are chosen based on the tonality, consequentially, the colour for the background will be selected relying on the same logic.
Two meticulously picked sets were chosen for major and minor scores, both of them consist of 6 pairs, while each pair has a background colour and edge colour.
Background sets have in them both neutral earth tones as well as bold chromatic colours so that they have options to suit everyone's liking. For major, they are black and pale spring bud as well as misty rose, key lime, xiketic and claret. For minor: black and cultured as well as deep taupe, morning blue, dark purple and illuminating emerald.
The edge colour is chosen either as a lighter/darker shade of the same colour or as a complementary colour to the one in the background.

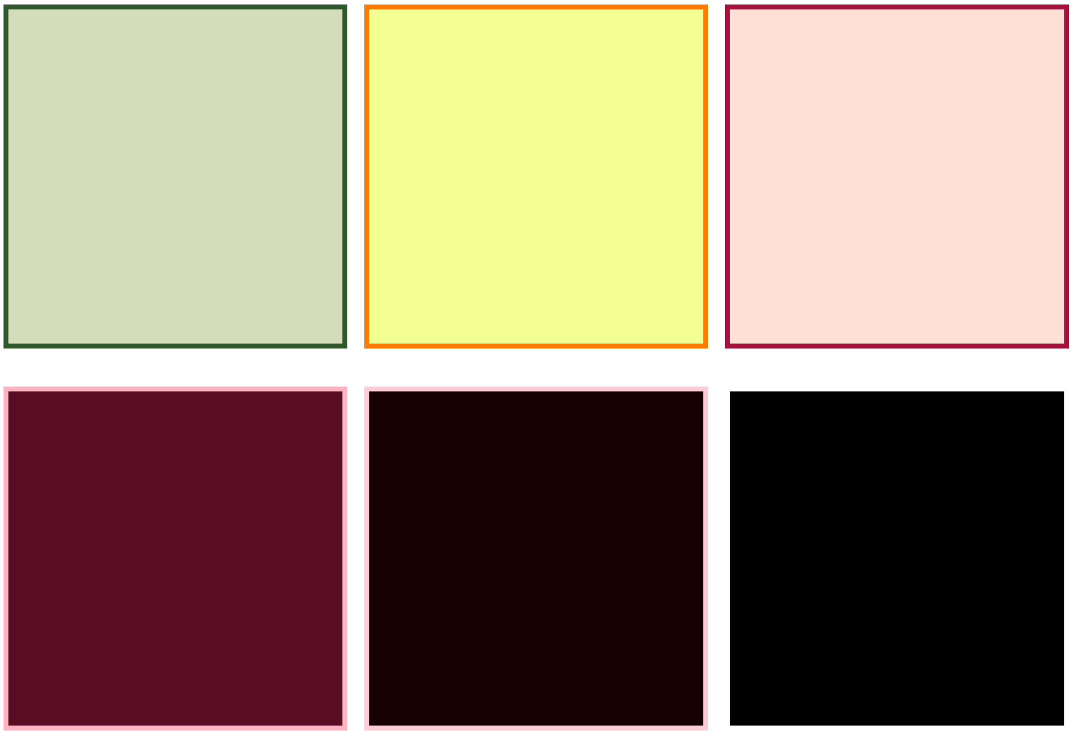


FIGURE 7.7: Background and edge colours for a warm palette

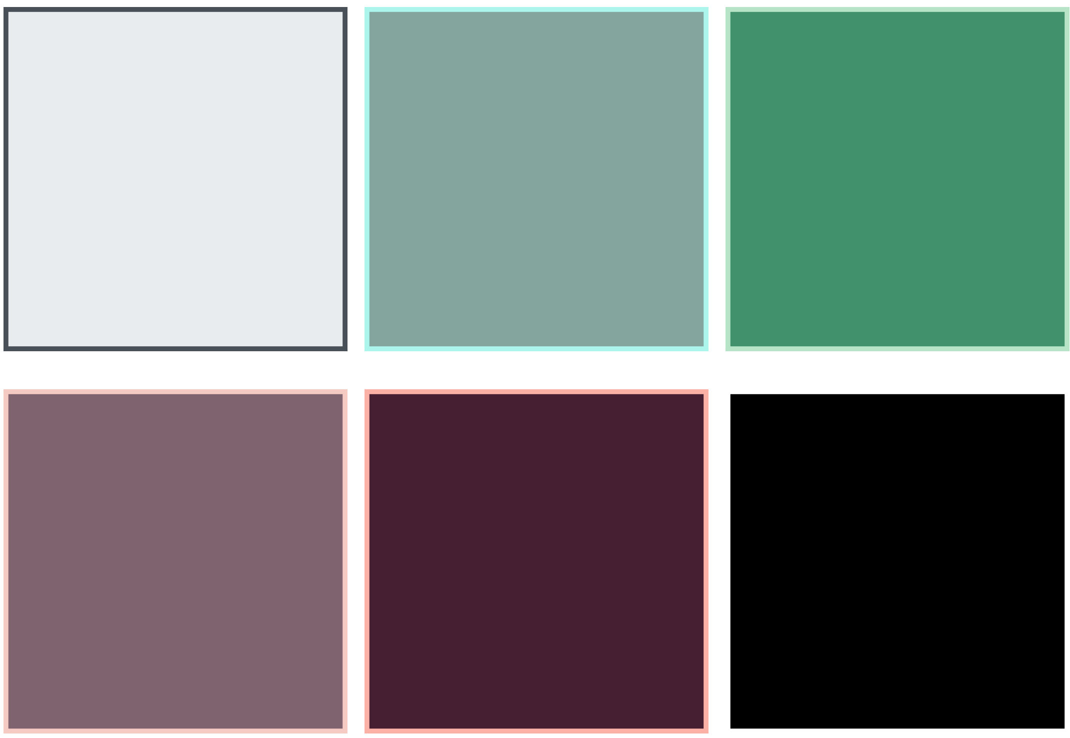


FIGURE 7.8: Background and edge colours for a cool palette

### 7.3.2 Title font

For the titles on the visualization five different fonts were selected - 3 more sharp and strict, and two more rounded and artistic. The list consists of DejaVu Sans, Oswald, Alata, Smooch and Zen Loop.


FIGURE 7.9: Examples of fonts

### 7.3.3 Placement of the title

After finishing improvement and alterations, and defining background colours and title fonts, the final thing to perform is arranging all the elements together and completing the visualizations.

#### Circular chart

From now on the start and the end of the score in the circular chart will not be connected anymore. On the contrary, the small space on the left side of the visualization's "flower" will indicate the start of the composition, which will last going clockwise and end on the left side at the bottom of the space.

Three options for the placement of the title will be available:

- Author and composition title in one line on top of the visualization's "flower" in the middle
- Piano roll will not start from the center of the "flower" but from one of the outer circles, creating a space in the middle of visualization, where the title will be placed
- The space that indicates the start of the score will be expanded, creating a sector, where the title will be placed

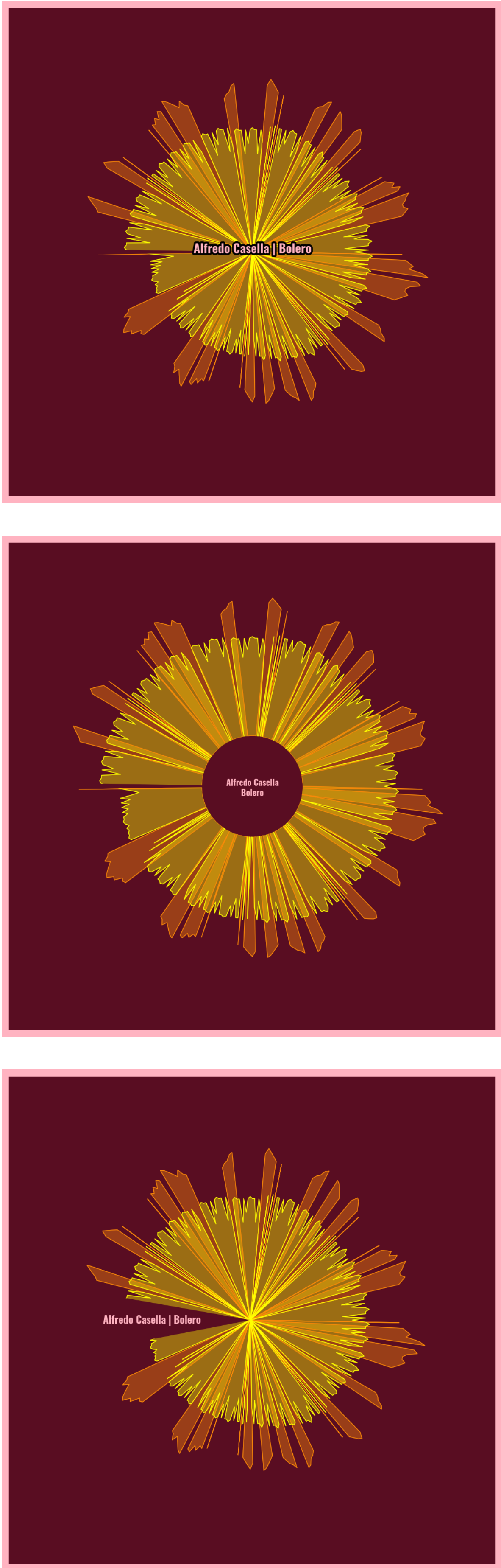



FIGURE 7.10: Arrangements of circular chart

**Popularity arc chart**

Four options for the placement of the title will be available for the popularity arc chart:

- Title placed in the upper left corner of visualization in two lines, author and composition
- Author and composition title in one line on top of the visualization's in the middle
- The author's name is in the upper left corner, while the composition title - in the upper right corner
- The author's name is in the lower left corner, while the composition title - in the lower right corner

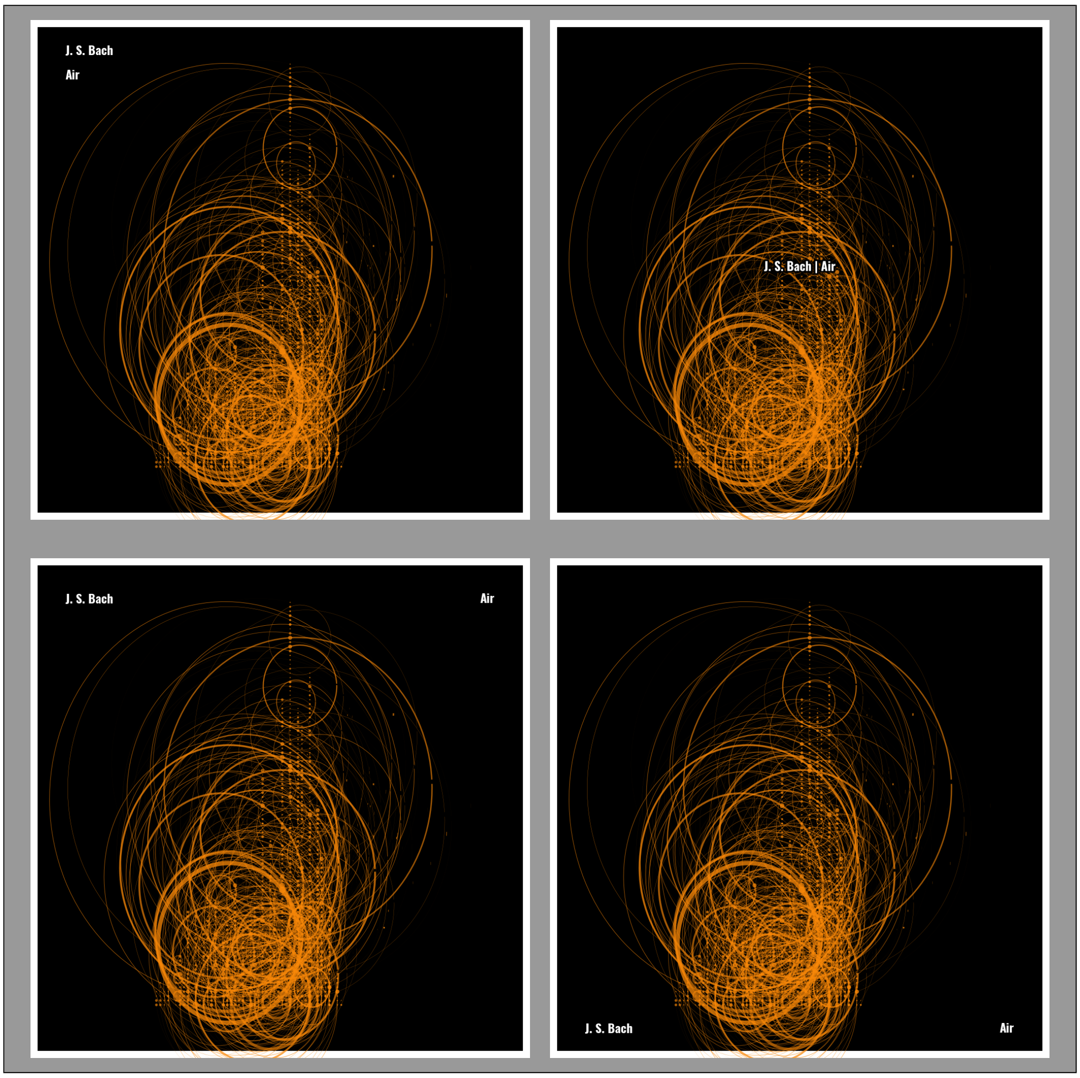



FIGURE 7.11: Arrangements of popularity arc chart

### Temporality plot & Triangular plot

Similar to the previous arc plot, temporality and Triangular plots have three identical options for the placement of the title:

- The author's name is in the lower left corner, while the composition title - in the lower right corner
- Title placed in the upper left corner of visualization in two lines, author and composition
- The author's name is in the upper left corner, while the composition title - in the upper right corner

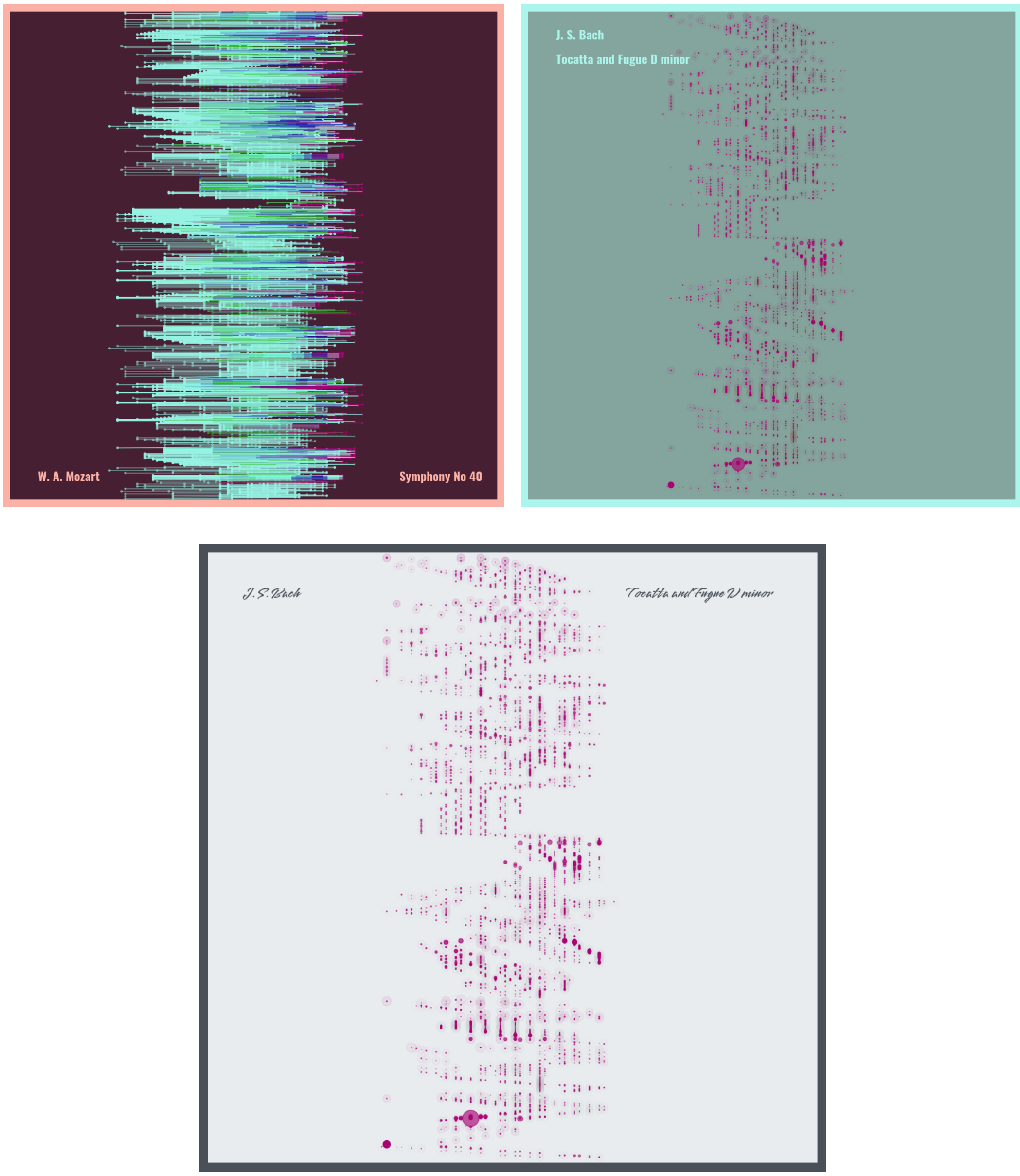



FIGURE 7.12: Arrangements of temporality plot & triangular plot

# Chapter 8

# Results

## 8.1 Visualization results

As a result of this project, four different approaches for visualizing music scores were created: Circular chart, Arc chart, Temporality plot and Triangular plot. Having six possible backgrounds, from three to four title placements and five text fonts, an approximate amount of 480 visualizations can be created for one music composition. Although not all of them are unique and interesting enough to become a worthy variant for an album cover, there is still a wide selection of alternatives to choose from, suitable for every taste.

For a display in this thesis, some of the most interesting and engaging covers were chosen. The election was based on the originality, appeal and uniqueness of the cover, as well as on the type and complexity of the input music score.

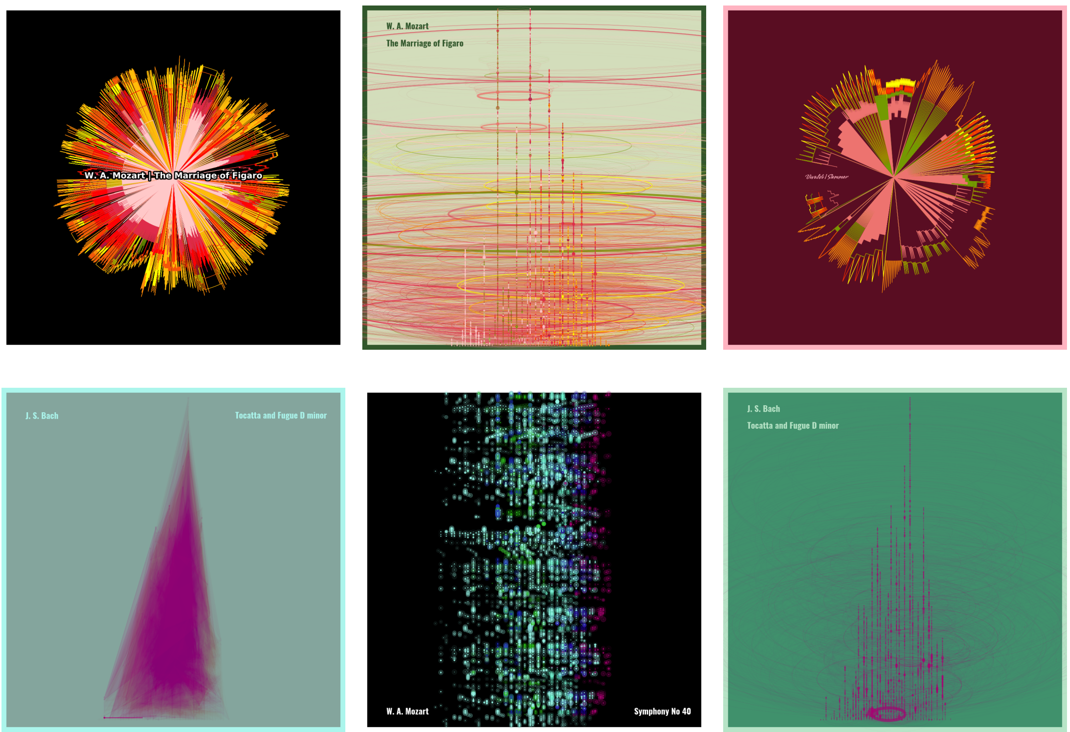



FIGURE 8.1: Example of result covers 1

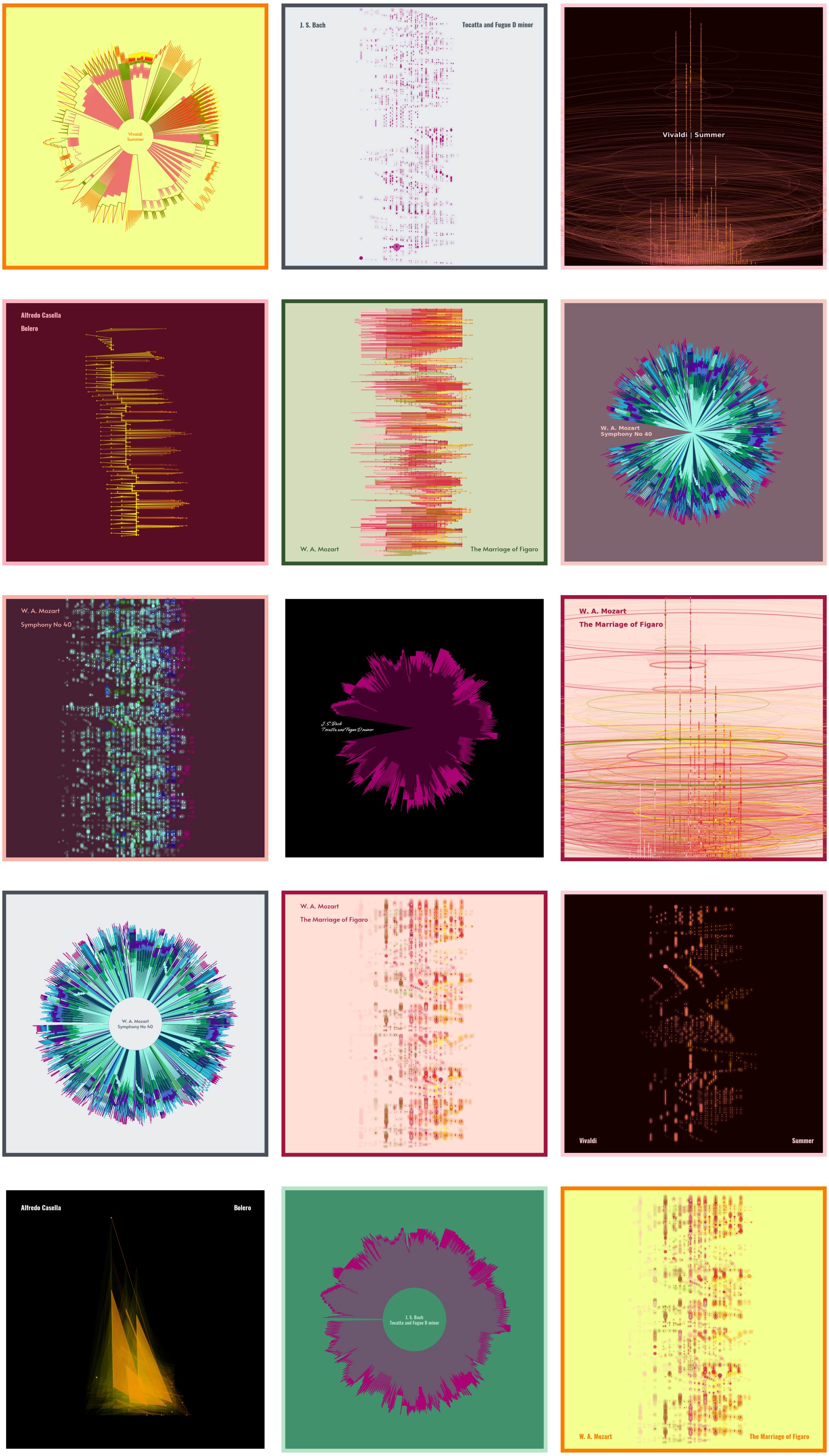



FIGURE 8.2: Example of result covers 2

## 8.2 Covers for streaming platforms

Taking into account that the goal of this work was to create a visual representation of the music score, that will serve as a good demonstration and portrayal of classical music composition and potentially can be used as a representation of classical pieces in music streaming services, the examples of possible outcomes were created. Using screenshots from a streaming platform Deezer [1], and adding some of the created visualizations led to the construction of some examples of how these pictures will serve as album/song covers.

Four examples were made: circular chart of Summer by Vivaldi (8.3), temporality plot of Symphony No. 40 by Mozart (8.4), arc chart of The Marriage of Figaro by Mozart (8.5) and triangular plot of Bolero by Casella (8.6).

## 8.3 Conclusion

This work serves as a confirmation that a decent replacement of general basic album covers for classical music indeed exists.
Four approaches were implemented based on data about music composition, which assure the unique and accurate demonstration of the music relying on its interior parts. Music properties such as key, instruments, timing, pitches, durations, and volumes were encoded into visualization properties - colour, opacity, width, size, amount, position etc. The existence of a wide variety of variations for future covers has been accomplished by adding a selection of additional attributes for more captivating results.

The examples of covers appearances on streaming services can be found in Streaming Services Covers.
All the covers shown in this thesis can be found in Visualizations From the Thesis.
Additional variations and alternatives of covers that were not included in the thesis, however may be interesting can be found in More Visualizations Variations.
The code of this project can be found in Visual Representation of Music Score GitHub repository.

In conclusion, this work shows how visualizations created by processing information about music are not only a good portrayal of the structure and essence of the composition, but also unique and interesting song covers, that can use their visual appeal as an attraction for people to listen and enjoy classical music on the streaming platforms.

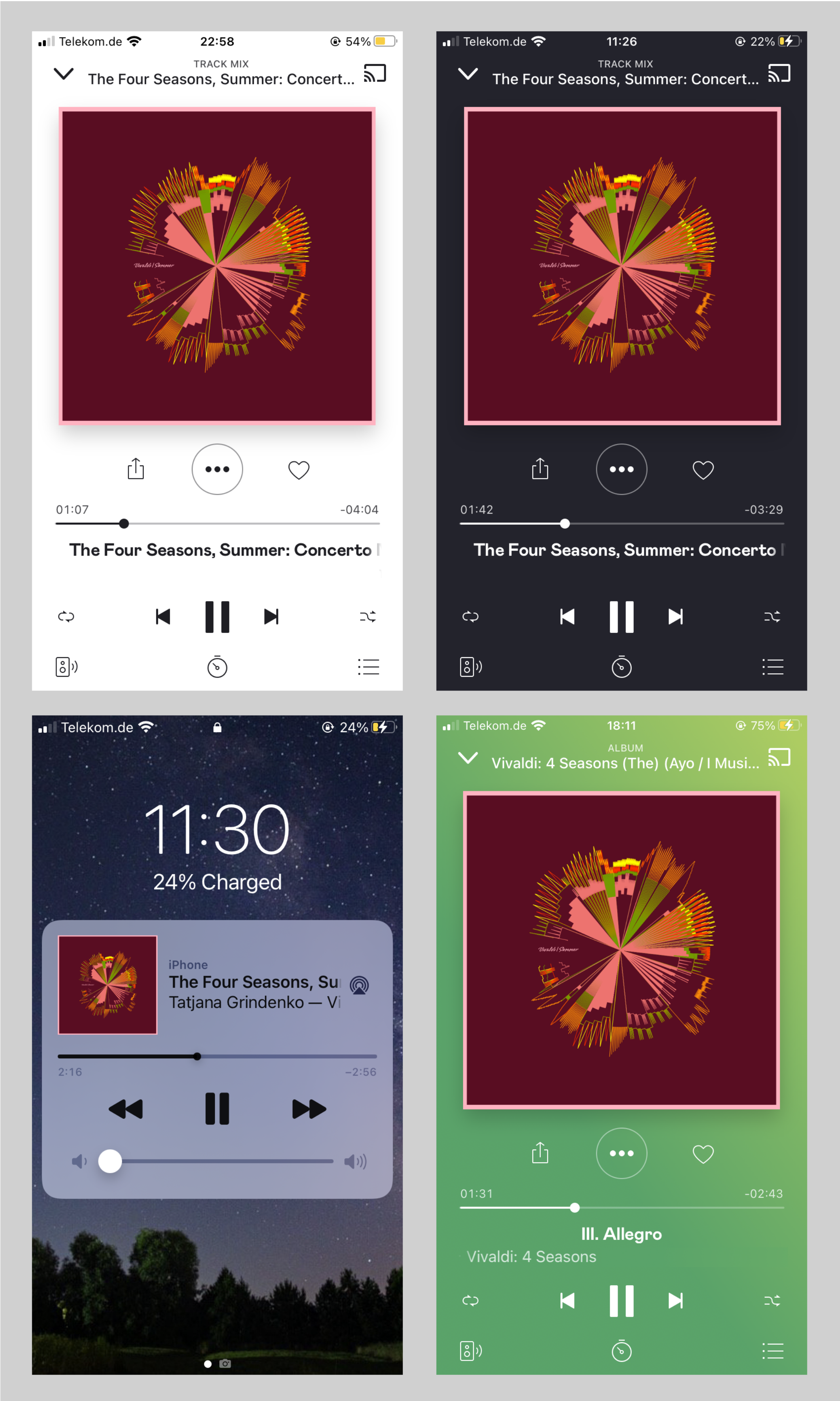



FIGURE 8.3: Example 1 of possible cover appearance on streaming services

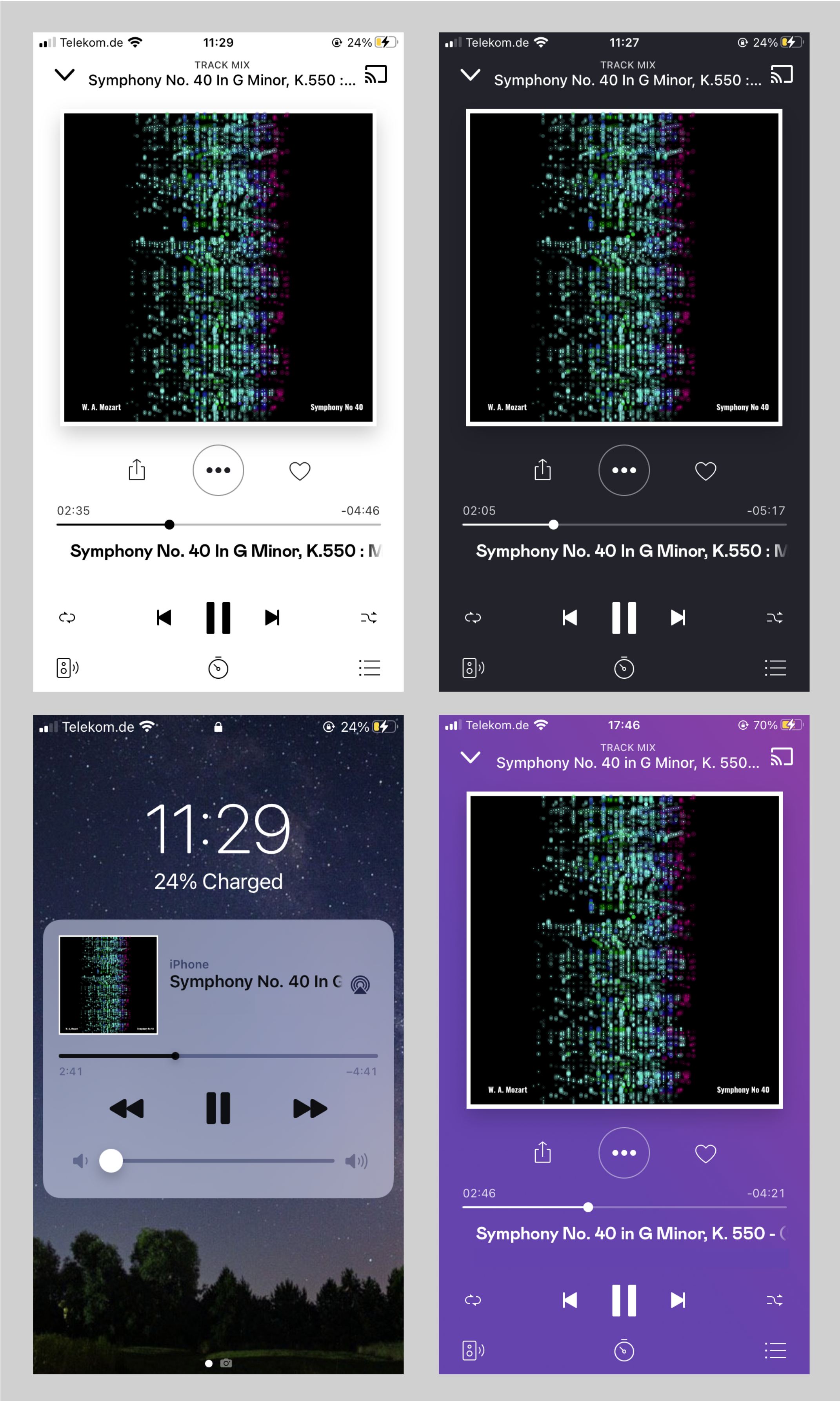



FIGURE 8.4: Example 2 of possible cover appearance on streaming services

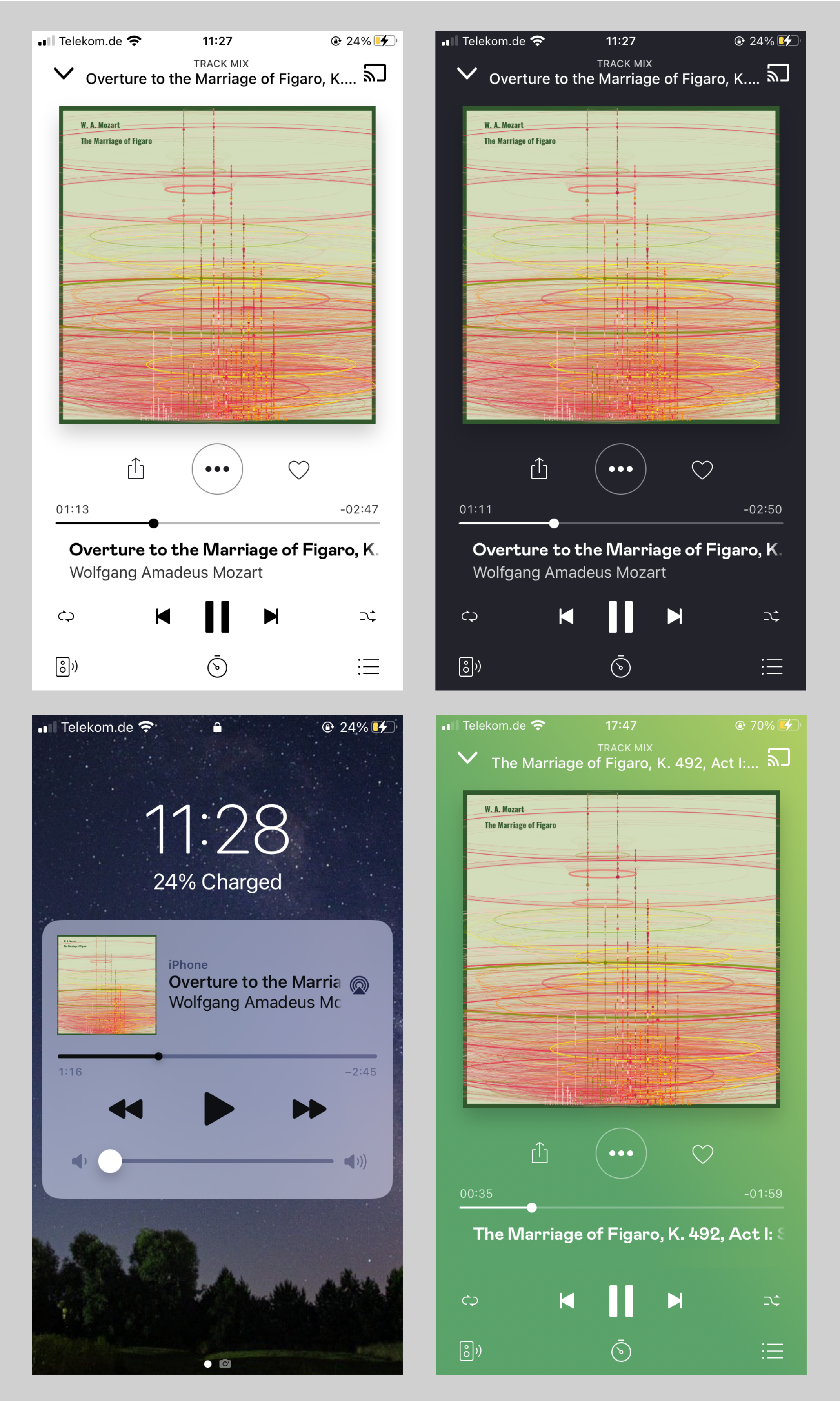



FIGURE 8.5: Example 3 of possible cover appearance on streaming services

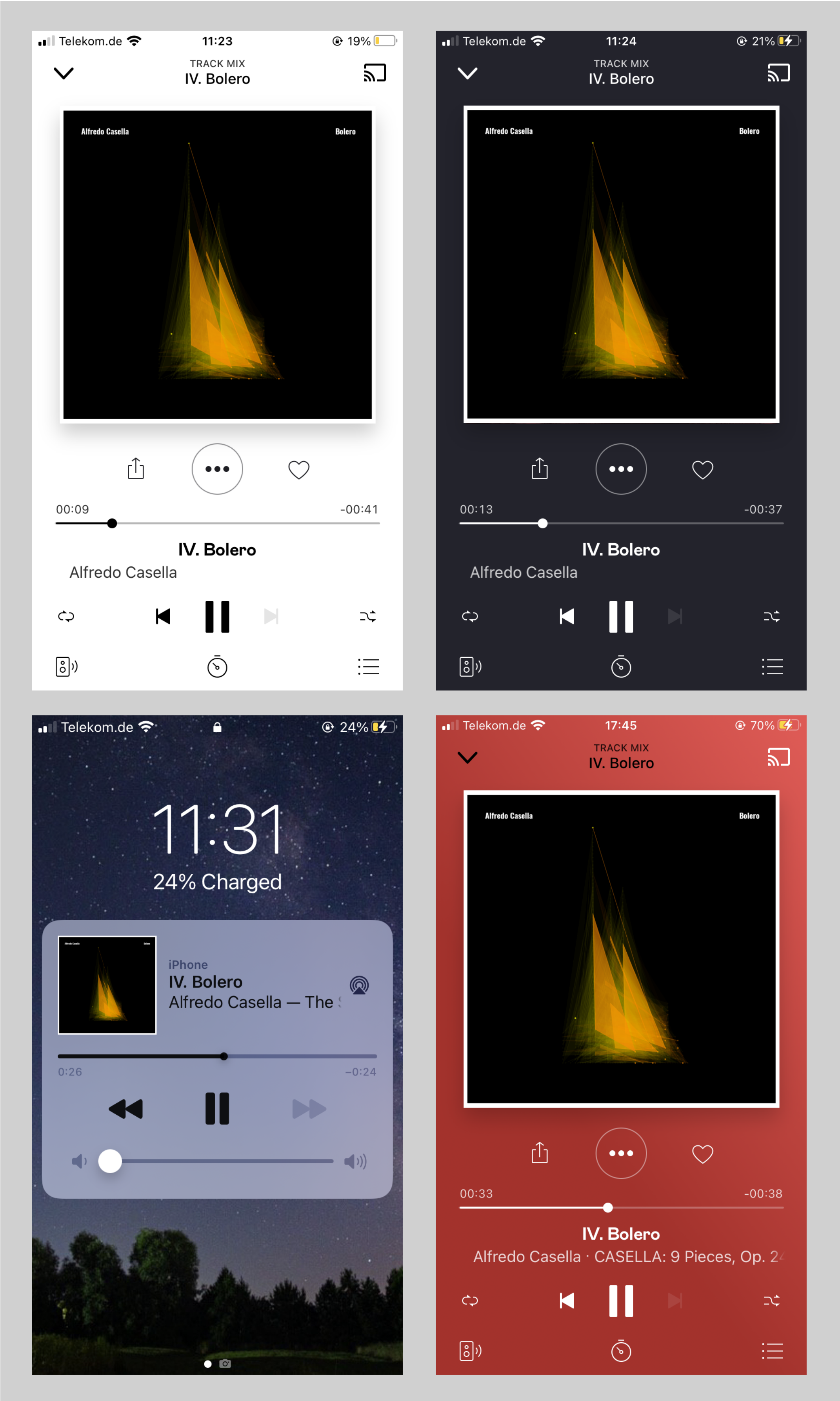
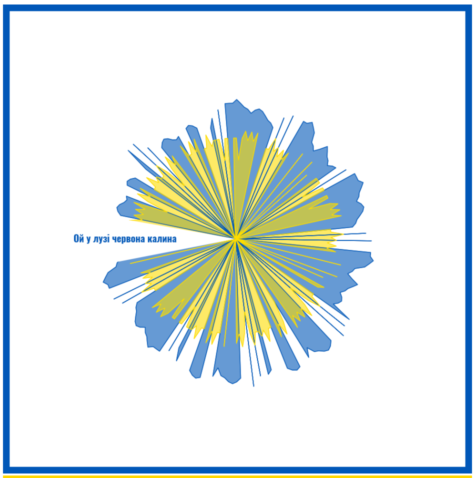


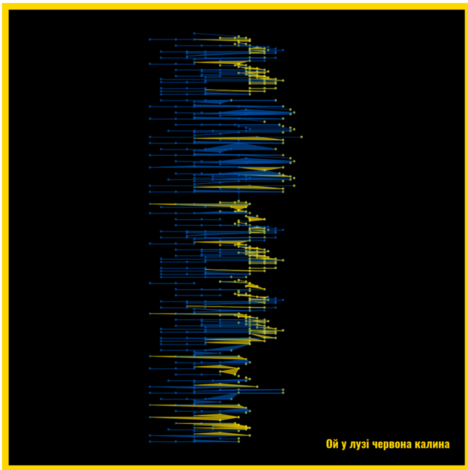

FIGURE 8.6: Example 4 of possible cover appearance on streaming services

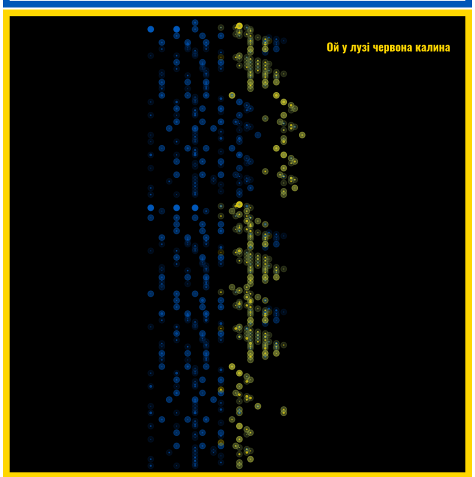
# Appendix A

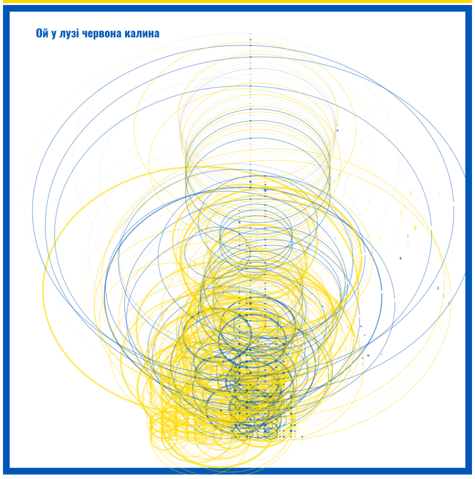
# Oh, the Red Viburnum in the Meadow

Most of this work was written during the time of the full-scale war in Ukraine. Now, when such horrendous efforts to destroy Ukrainian identity and culture are being made, it is of great importance to keep our culture close to our hearts, embrace it, celebrate it and show its beauty, authenticity and uniqueness to the whole world.
Therefore as the main focus of this work is music, which is a big part of the culture as well, I have decided to include a special addition to this thesis, by creating a visualization of the Ukrainian patriotic folk song “Ой у лузі червона калина” (transl."Oh, the Red Viburnum in the Meadow").
The arrangement for piano and voice by Andriy Bondarenko was taken as input for creating the covers [3]. Instead of the warm/cool palette scheme the colours of the Ukrainian flag, blue and yellow, were used as a representation of piano and voice.









# Bibliography


[1] Access Industries and Rotana Group. *Deezer*. Version 9.14.0. 2007. URL: hhttps://www.deezer.com/uk/.

[2] Dominique Vandenneucker Arpege Music. *Geometry of Music: Visualizing Beethoven's Fifth Symphony*. 2018. URL: http://www.music-software-development.com/midi-tutorial.html.

[3] Andriy Bondarenko. "Ой у лузі червона калина" *for piano and voice*. 2019. URL: https://www.facebook.com/ukrnotes.in.ua/posts/2610849848936995/.

[4] Encyclopaedia Britannica. *Key*. URL: https://www.britannica.com/art/key-music.

[5] Encyclopaedia Britannica. *Pitch*. URL: https://www.britannica.com/art/pitch-music.

[6] Encyclopaedia Britannica. *Staff*. URL: https://www.britannica.com/art/staff.

[7] Kent Bye. *Geometry of Music: Visualizing Beethoven's Fifth Symphony*. 2018. URL: https://www.youtube.com/watch?v=XTv8g_DEH5Q&ab_channel=KentBye.

[8] Colin Raffel and Daniel P. W. Ellis. *pretty_midi package*. 2014. URL: https://craffel.github.io/pretty-midi/.

[9] Jono Kornfeld. *Music Notation and Theory for Intelligent Beginners*. Jason Dullack, 2005. DOI: http://jkornfeld.net/complete_theory_text.pdf.

[10] Steve Krenz. *Learn and Master Guitar Lesson Book*. Legacy Learning System, 2010. ISBN: 9781450721493. DOI: https://www.learnandmaster.com/resources/Learn-and-Master-Guitar-Lesson-Book.pdf.

[11] Stephen Malinowski. *Home of Stephen Malinowski and the Music Animation Machine*. URL: http://www.musanim.com/.

[12] Stephen Malinowski. *MAM Player User Guide*. URL: https://www.musanim.com/Player/MAMPlayerUserGuide.pdf.

[13] Stephen Malinowski. *Music Animation Machine*. URL: http://www.musanim.com/all/.

[14] Stephen Malinowski. *Music Animation Machine Frequently-Asked Questions*. URL: http://www.musanim.com/mam/mamfaq.html#tantrum.

[15] Stephen Malinowski. *Music Animation Machine MIDI Player*. URL: http://www.musanim.com/Player/.

[16] Stephen Malinowski. *Music Animation Machine Techniques*. URL: http://www.musanim.com/Techniques/.

[17] Stephen Malinowski. *Stephen Malinowski YouTube channel*. URL: https://www.youtube.com/user/smalin.

[18] Merriam-Webster. *Volume*. URL: https://www.merriam-webster.com/dictionary/volume.

[19] MuseScore BVBA. *MuseScore*. Version 3.6.2. URL: https://musescore.org/uk.

[20] Ole Martin Bjørndalen. *Mido - MIDI Objects for Python*. Version 1.2.10. URL: https://mido.readthedocs.io/en/latest/.

[21] Nick Romeo. "This Amazing Machine Lets You 'Paint' Mozart's Music". In: *Daily Beast* (2015). DOI: https://www.thedailybeast.com/this-amazing-machine-lets-you-paint-mozarts-music.

[22] Wikipedia. *Clef*. URL: https://en.wikipedia.org/wiki/Clef.

[23] Wikipedia. *Numbered musical notation*. URL: https://en.wikipedia.org/wiki/Numbered_musical_notation.

[24] Wikipedia. *Timing*. URL: https://en.wikipedia.org/wiki/Timing_(music).