Interactive Music Science Collaborative Activities

Team Teaching for STEAM Education

Deliverable 4.2
First Version of Pen-enabled multimodal interaction interface

Date:	30/9/2017
Author(s):	Israel Fenor (WIRIS), Dani Martín-Albo (WIRIS), Manu Bouillon (UNIFRI), Marcus Liwicki (UNIFRI)
Contributor(s):	
Quality Assuror(s):	Robert Piechaud (IRCAM), Renaat Frans (UCLL)
Dissemination level:	PU
Work package	WP4 – Core enabling technologies modules of iMuSciCA
Version:	1.0
Keywords:	pen interaction, multimodal interaction, music creation
Description:	First version of the demonstrator for pen-enabled multimodal interaction interface

H2020-ICT-22-2016 Technologies for Learning and Skills
iMuSciCA (Interactive Music Science Collaborative Activities)
Project No. 731861
Project Runtime: January 2017 – June 2019

Executive Summary

In this deliverable we present the first version of the demonstrator for gesture and VR multimodal interaction interface.
We will mainly describe the following interfaces:

- The WIRIS EDITOR allowing to edit mathematical equations. The editor is accessible at http://www.wiris.com/editor/demo/en/ and has a manual at http://www.wiris.com/en/editor/docs/manual
- The WIRIS GRAPH environment which allows drawing and interacting with graphs and functions. It is accessible at http://www.wiris.net/demo/graph/tests/en/test.html
- The UNIFRI DrAwME environment allowing to directly draw music and create sound in an innovative way. IT is accessible at https://imuscica-canvas.herokuapp.com

Version Log			
Date	**Version No.**	**Author**	**Change**
04-09-2017	0.1	Israel Fenor (WIRIS)	Initial content
19-09-2017	0.2	Dani Martín-Albo (WIRIS)	Sections 1, 2 & 3 written
25-09-2017	0.3	Manuel Bouillon (UNIFRI)	Section 4
28-09-2017	0.4	Marcus Liwicki (UNIFRI)	Summary
30-09-2017	1.0	Vassilis Katsouros	Submission to the EU

Disclaimer

This publication has been produced with the assistance of the European Union. The content of this publication is the sole responsibility of iMuSciCA consortium and can in no way be taken to reflect the views of the European Union.
iMuSciCA is an H2020 project funded by the European Union.

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

Abbreviation	Description
WYSIWYG	What You See Is What You Get
ATHENA	ATHENA RESEARCH AND INNOVATION CENTER IN INFORMATION COMMUNICATION & KNOWLEDGE TECHNOLOGIES
UCLL	UC LIMBURG
EA	ELLINOGERMANIKI AGOGI SCHOLI PANAGEA SAVVA AE
IRCAM	INSTITUT DE RECHERCHE ET DE COORDINATION ACOUSTIQUE MUSIQUE
LEOPOLY	3D FOR ALL SZAMITASTECHNIKAI FEJLESZTO KFT
CABRI	Cabrilog SAS
WIRIS	MATHS FOR MORE SL
UNIFRI	UNIVERSITE DE FRIBOURG

1. Introduction

In this deliverable we describe the different core-enabling technologies related to pen interaction contributed by different iMuSciCA partners.

2. WIRIS EDITOR

WIRIS EDITOR[1] is a powerful tool that allows composing any math expression with high-graphics quality. WIRIS editor is based upon standards like MathML for internal representation and the PNG image format for displaying formulas. It can also handle other formats like LaTeX, flash, SVG and EPS.

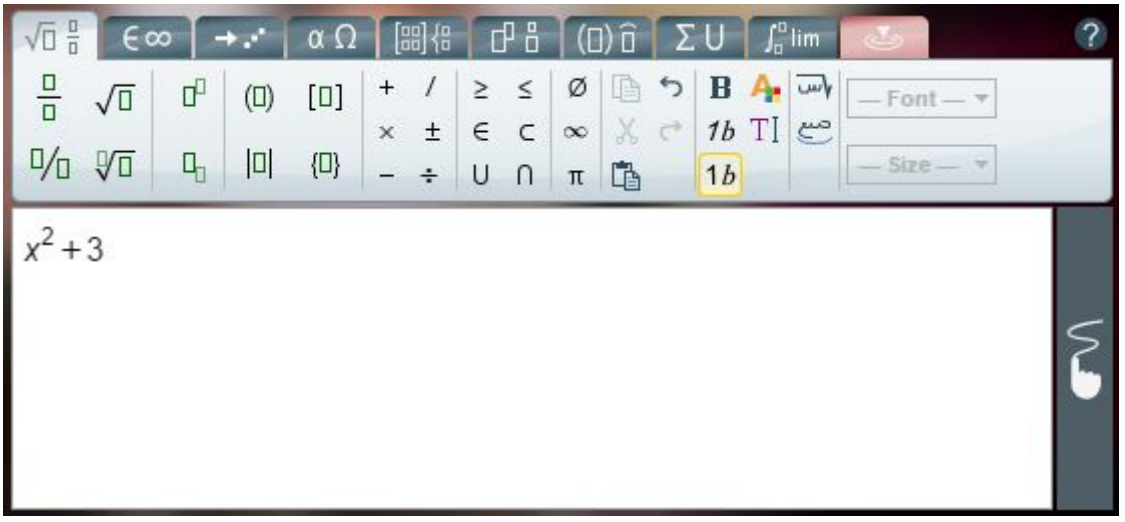

Figure 2-1: WIRIS EDITOR with the WYSIWYG interface. The device keyboard and the GUI toolbar can be used to include symbols and operators. You can switch to the pen-enabled input interface by pressing the rightmost big dark button (hand icon)

WIRIS EDITOR comes with two user input interfaces: a classic visual editor with a palette and icons (see Figure 2-1). The pen-enabled input method integrates a math expression recognition engine. This engine is based on an approach that integrates symbol segmentation and classification and structure recognition. The recognition process is guided by a probabilistic grammar that accounts for the structural probability between symbols and sub-expressions. The engine is completed with specialized modules for specific tasks: spatial relations classification, symbol segmentation and symbol classification. Mathematical symbol classification is performed using neural networks and a combination of several sets of online and offline features. The remaining models are also statistical classifiers such that all probabilistic sources of information are estimated from data.

[1] Demo accessible here: http://www.wiris.com/editor/demo/en/

2.1. Installation and technical requirements

WIRIS EDITOR is based on JavaScript, so it runs on any browser (Firefox, Internet Explorer, Chrome, Safari...) and operating system (Windows, GNU/Linux, macOS...). You can insert WIRIS EDITOR with the following code:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html>
  <head>
    <script src="http://www.wiris.net/demo/editor/editor"></script>
    <script>
      var editor;
      window.onload = function () {
        editor = com.wiris.jsEditor.JsEditor.newInstance({'language': 'en'});
        editor.insertInto(document.getElementById('editorContainer'));
      }
    </script>
  </head>
  <body>
    <div id="editorContainer"></div>
  </body>
</html>
```

2.2. Description of demonstrator and user manual

In this document we focus on the handwriting input interface. More details on the WYSIWYG input method here: http://www.wiris.com/en/editor/docs/manual.

The demonstrator can be accessed here: http://www.wiris.com/editor/demo/en/. First, you have to switch to the handwriting input. This can be done by pressing the rightmost button (hand icon). If you want to write a new formula you just need to draw some strokes in the canvas and the engine will transform them into the most probable math formula. For example, in Figure 2-2, the user has introduced some strokes that the engine has correctly decoded as "x^2+3".

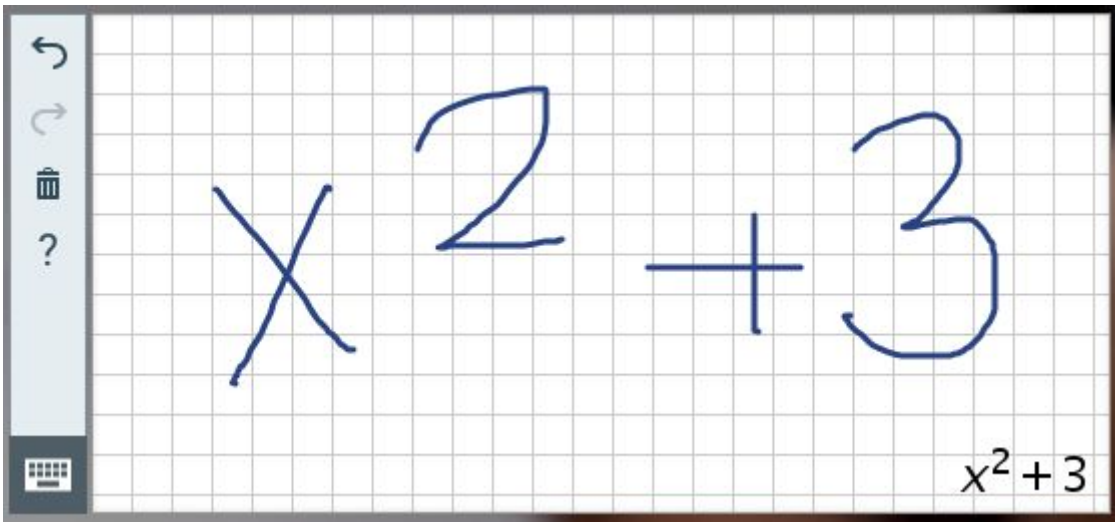

Figure 2-2: Pen-enabled WIRIS EDITOR input interface. In this interface, strokes can be drawn in the canvas and the math expression recognition engine will transform them into an equation in MathML format. You can switch to the visual editor input interface pressing the lower-left button (keyboard icon)

3. WIRIS GRAPH

For visualization and analysis tool we will use WIRIS GRAPH[2]. WIRIS GRAPH is a powerful tool that allows drawing geometric primitives or plotting equations with high graphics quality.

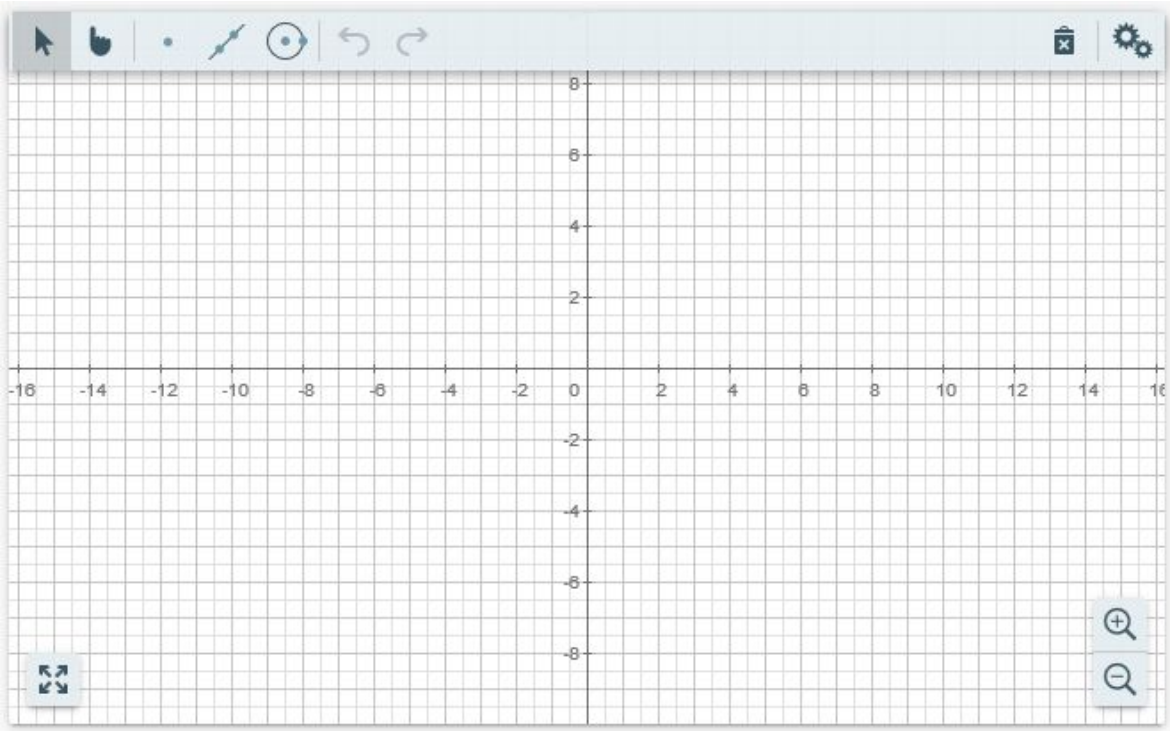

Figure 3-1: *WIRIS GRAPH interface*

[2] Demo: http://www.wiris.net/demo/graph/tests/en/test.html

Different geometric primitives can be drawn choosing the appropriate icon from the GUI:

- Points
- Lines
- Line segments
- Rays
- Polylines
- Circles (defined by a center and a point or by three points)
- Circle arcs (defined by three points)
- Parabolas

Moreover, WIRIS GRAPH can plot any univariate equation provided in Content MathML. For example, Figure 3.5-5 shows the WIRIS GRAPH plotting the equation $f(x)=x^4$:

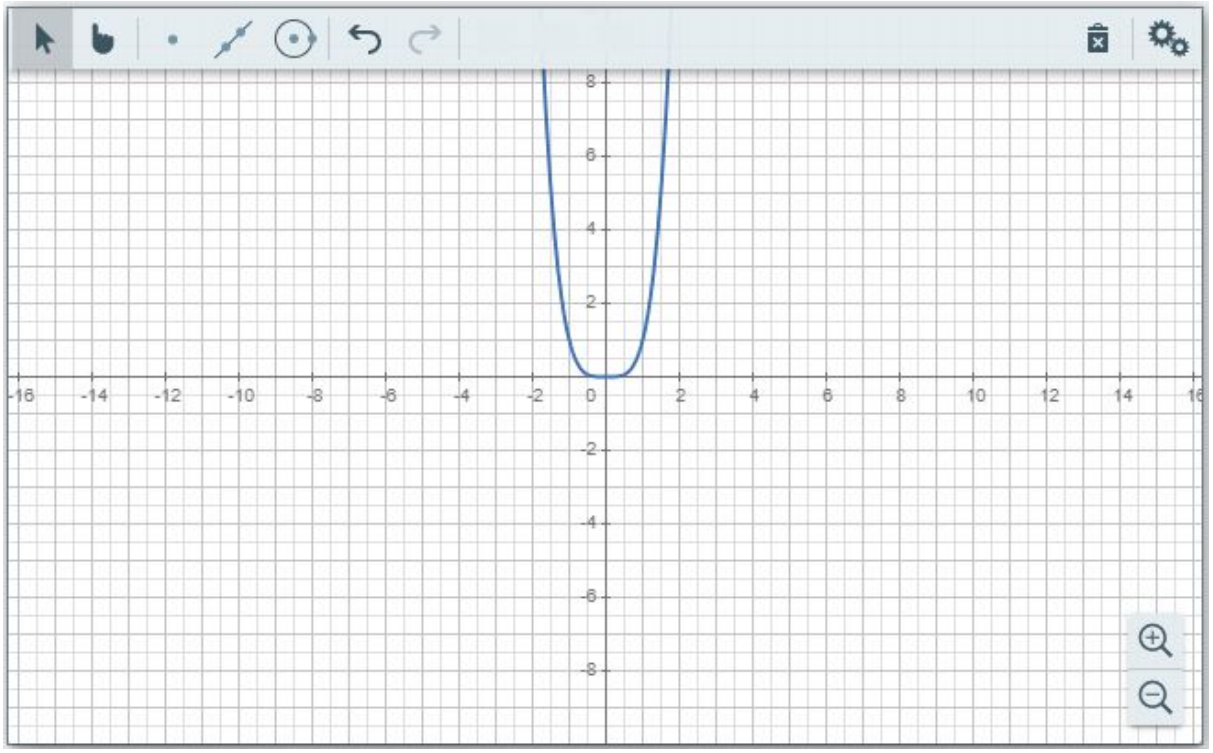

Figure 3-2: WIRIS GRAPH plotting the equation $f(x)=x^4$

In addition to traditional input, WIRIS GRAPH provides a pen-enabled input method that integrates a stroke recognition engine. This engine classifies a stroke into one of the following geometric primitives:

- Points
- Line segments
- Polylines
- Circles
- Circle arcs

- Parabolas
- Ellipses
- Ellipse Arcs
- Sinusoids
- Polygons

3.1. Installation and technical requirements

WIRIS GRAPH is based on JavaScript, so it runs on any browser (Firefox, Internet Explorer, Chrome, Safari...) and operating system (Windows, GNU/Linux, macOS...). You can insert WIRIS GRAPH with the following code:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html>
  <head>
    <script src="TBD"></script>
    <script>
      var editor;
      window.onload = function () {
        graph = com.wiris.js.JsGraph.newInstance();
        graph.insertInto(document.getElementById('graphContainer'));
      }
    </script>
  </head>
  <body>
    <div id="graphContainer"></div>
  </body>
</html>
```

3.2. Description of demonstrator and user manual

Let's imagine that we want to draw a particular geometric primitive, such as a circle, using WIRIS GRAPH. For that we need to provide some pen-strokes and the recognition engine will understand the user intention, transforming the pen-strokes into the set of parameters that best defines that geometric primitive. Once the recognition has been performed, the input stroke is replaced by the proper geometric primitive, as shown in Figure 3.3.

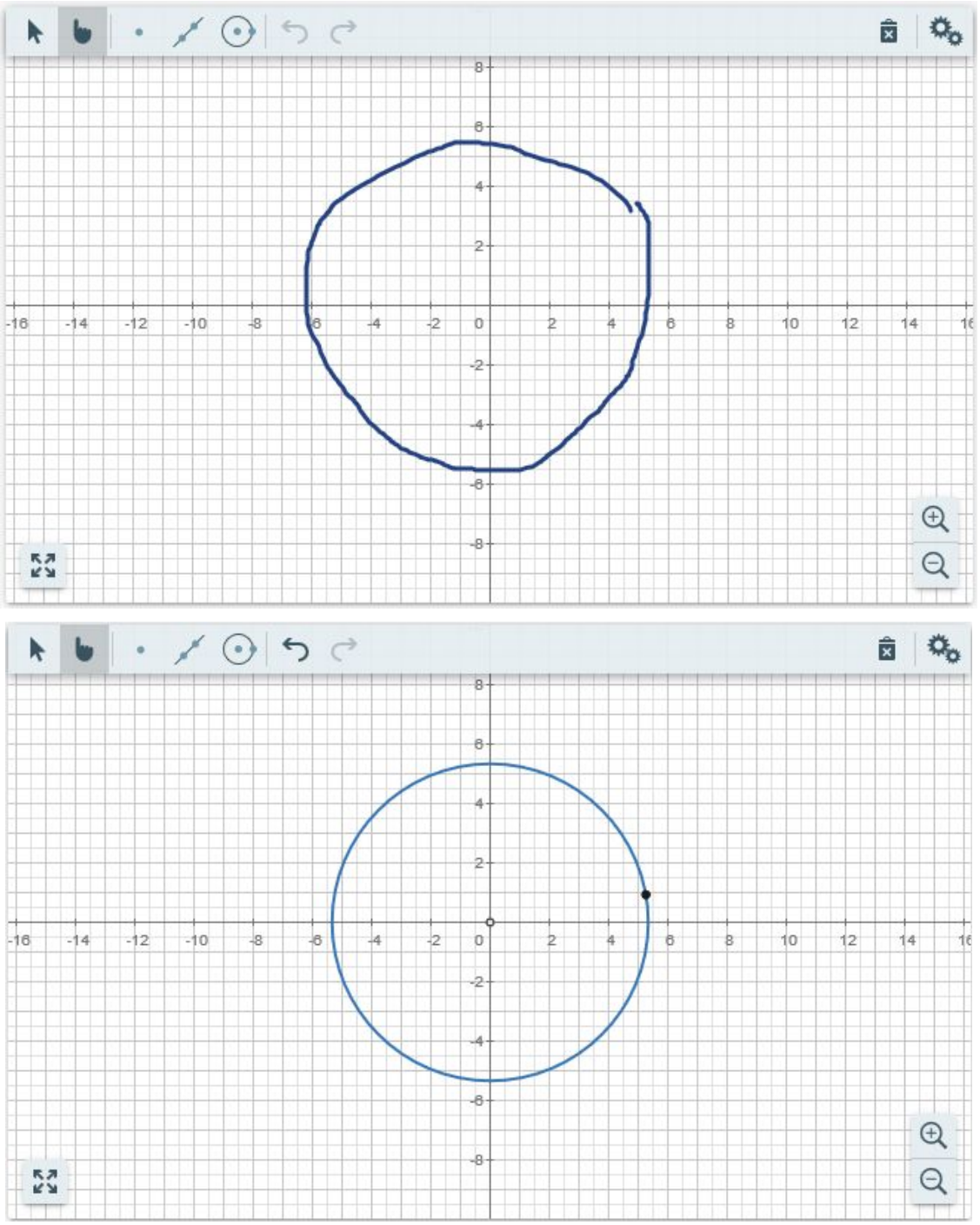

Figure 3-3: Example of WIRIS GRAPH recognizing user input strokes. Top: the user has drawn a stroke with the intention of drawing a circle. Bottom: the recognition engine transforms the stroke into the most suitable primitive, which is indeed a circle

Many different geometric primitives can be drawn using handwriting as input. For example:

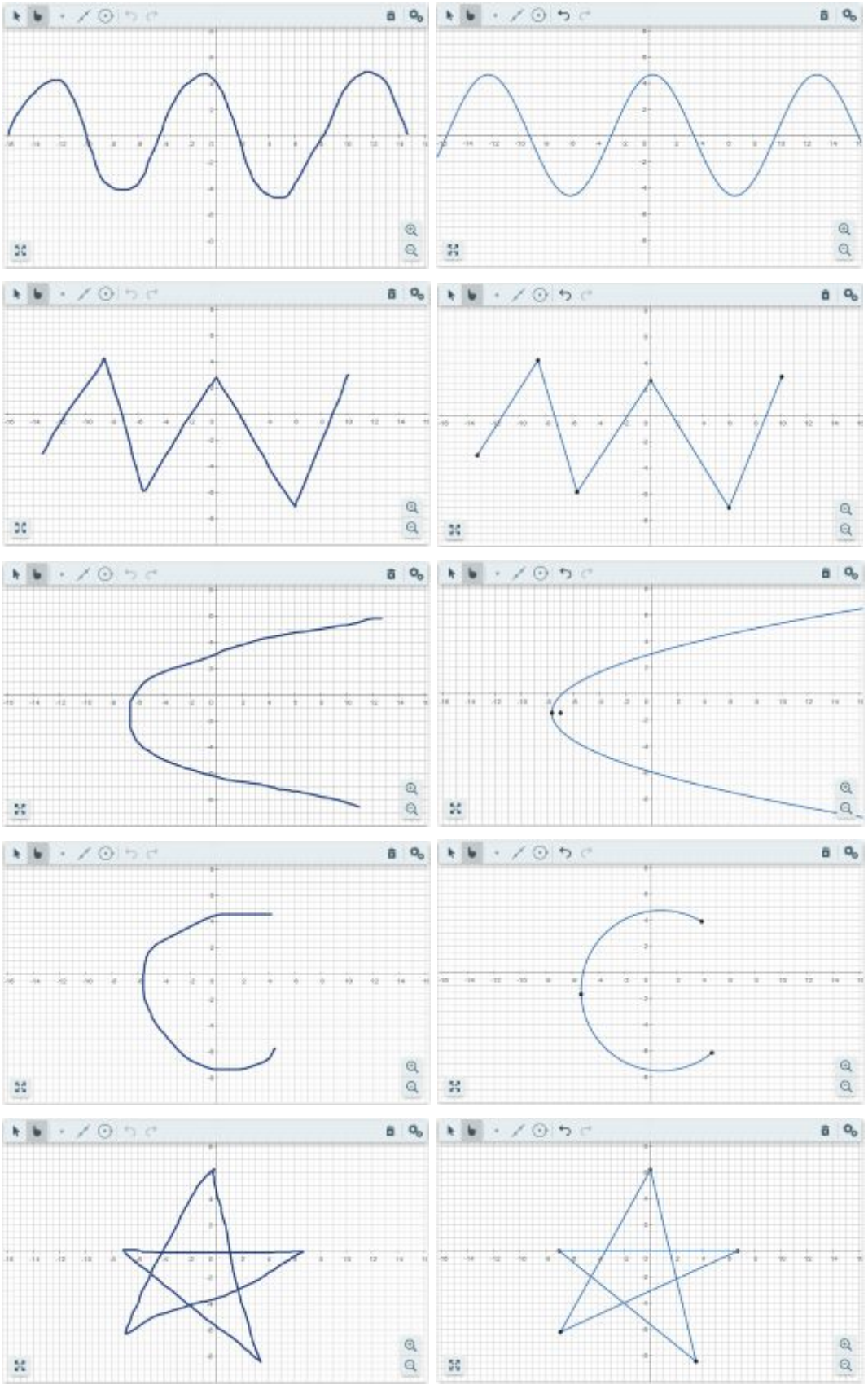

Figure 3-4: Different geometric primitives drawn using handwriting input on WIRIS GRAPH

4. UNIFRI DrAwME (Drawing cAnvas for Music crEation)

UniFri Dame is a powerful tool that allows composing music and exploring sound properties visually.

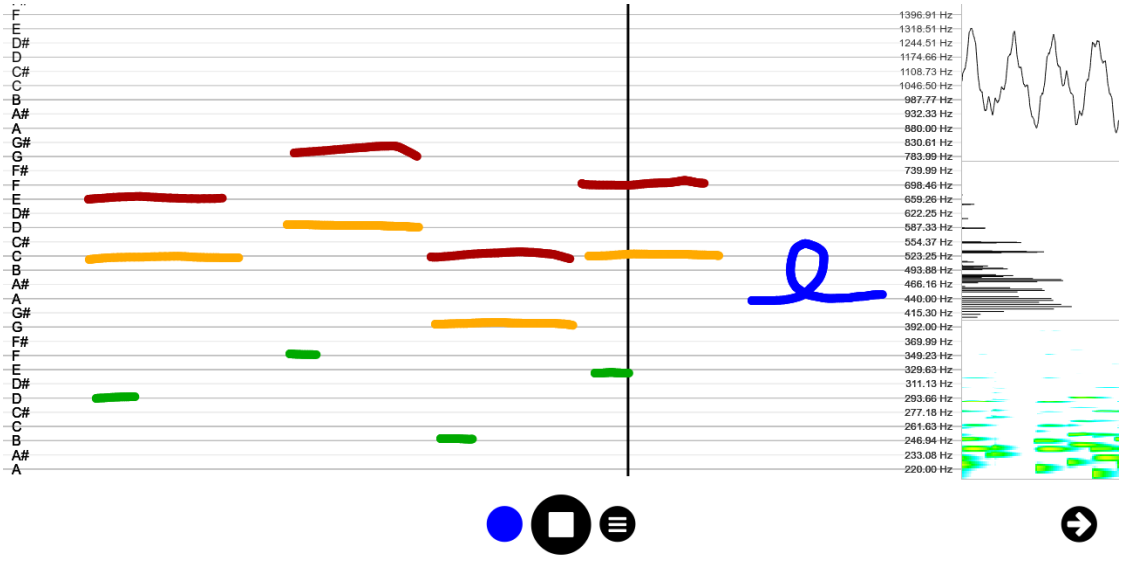

Figure 4-1: UniFri Dame with the drawing canvas, the control buttons at the bottom and the three visualization views on the right side

4.1. Installation and technical requirements

UniFri DrAwME is based on JavaScript, so it runs on any modern browser (Chrome, Firefox, Edge, etc.) and operating system (Windows, GNU/Linux, macOS). It is accessible at the following URL: https://imuscica-canvas.herokuapp.com. You can insert UniFri DrAwME in an iframe with the following code:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html lang=en>
<head>
    <meta charset=utf-8>
    <title>UniFri Dame</title>
</head>
<body style="margin:0px;padding:0px;overflow:hidden">
       <!-- DAME iframe -->
       <iframe   id="tool"   src="https://imuscica-canvas.herokuapp.com"   frameborder="0"
height="100%" width="100%"></iframe>
</body>
</html>
```

4.2. Description of demonstrator and user manual

UniFri DrAwME is composed of three parts, the drawing canvas in the middle, the control buttons at the bottom and the visualization views on the right.

4.2.1. Drawing canvas

The user can draw on the canvas and hear the corresponding sound at the same time. Multiple colors, corresponding to different pitches, are available for drawing.

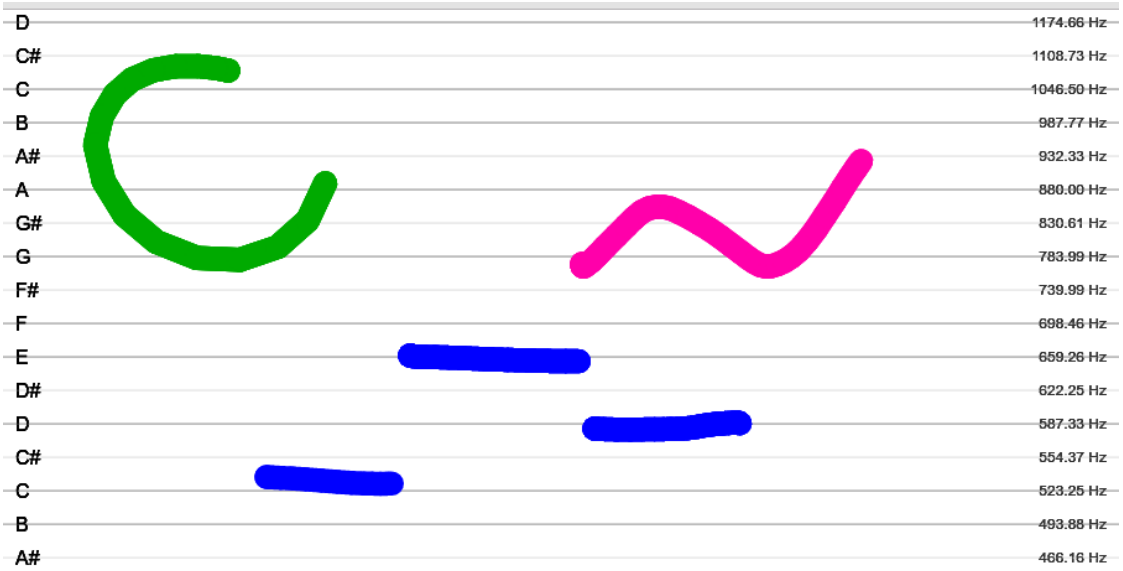

Figure 4-2: The drawing canvas, where the user can pen- or hand-draw with different colors. Pen erase button (and mouse right click otherwise) can be use to delete strokes. Pinch-like gesture (when enabled in option menu, and mouse scrolling button otherwise) can be used to zoom in and out. Two scales are displayed: a note scale on the left and and frequency scale on the right

4.2.2. Control buttons and settings

Below the drawing canvas, the control buttons enable to play the whole drawing by clicking on the "play" button, and to edit different settings (see Figure 4-4). A color picker enable to change the drawing colors, which correspond to different pitches. The menu button give access to differents options:

- Activate hand gesture interactions
- Activate stick to line option (to force drawing on the note/frequency lines)
- Clear the canvas
- Open settings window

Figure 4-3: The control buttons with the color picker on the left, the play/stop button in the middle and the option menu on the right. The option menu permits to active the hand gesture interactions (zoom, etc.), to activate the stick to line feature, to clear the canvas and to open the settings window.

The settings window enable to:

- Adjust volume
- Adjust playback speed
- Enable/disable loop playback
- Smoothen strokes
- Display debug information
- Export/import drawing in JSON format

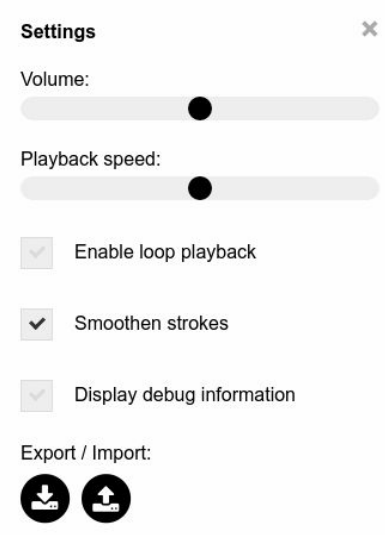

Figure 4-4: The setting menu permits to adjust the volume and playback speed, to enable/disable loop playback, to (de-)activate stroke smoothing, to display debug information and to import/export the drawing in JSON format

4.2.3. Visualization panel

Anytime sound is produce, it can be visualized in the visualization panel on the right. The visualization panel includes three views:

- A waveform view
- A Fourier transform view
- A spectrogram view

The visualization panel can be hidden by clicking on the arrow at the bottom right, and can be opened in an external window by clicking on them.

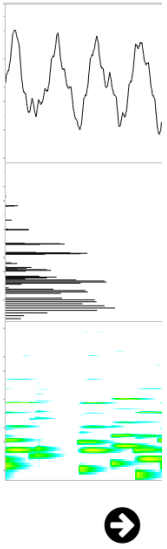

Figure 4-5: The visualization panel include three different views: a waveform, a Fourier transform and a spectrogram. The visualization panel can be hidden by clicking on the arrow on the bottom right, and can be opened in an external window by clicking directly on the views