HandMonizer: a case study for personalized digital musical instrument design

Davide Lionetti*, Antonios Pappas*,
Luca Comanducci†, Alberto Bernardini†,
Massimiliano Zanoni†, Augusto Sarti†
Dipartimento di Elettronica, Informazione e Bioingegneria
Politecnico di Milano
Milan, Italy
Email: *{name.surname@mail.polimi.it},
†{name.surname@polimi.it}

Matthew Yee-King and Mark d'Inverno
Goldsmiths, University of London
London, United Kingdom
Email: {m.yee-king, dinverno}@gold.ac.uk

***Abstract*—The rapid evolution in technology has found its way to introducing novelty in today's live music performances. In this context, the development of Digital Musical Instruments (DMIs) has obtained increasing attention in recent years. In this paper, we present the development of a DMI called *Handmonizer*, an interactive artist-oriented harmonizer for musical performance adapted to the needs of a specific singer. A key component of our work is the combination of hand motion recognition and audio signal processing to obtain a smoother interaction. We describe the development methodology, but we also focus on our collaboration with the artist to conceptualize and then refine this tool until the development of the final product. At the end of this paper, we define an evaluation strategy, collecting feedback with a questionnaire addressed to the singer. Our aim in presenting this evaluation strategy is to help other engineers keen to develop cutting-edge technologies by working in partnership with artists. While results are not definitive, we believe that the chosen methodology could be of interest to other DMI researchers. Moreover, the modular nature of the Handmonizer makes it easily adaptable to further developments concerning the Internet of Sounds (IoS) and Networked Music Performances (NMP).**

***Index Terms*—Music interaction design, human-centered design, hand gesture recognition, vocal harmonizer, live performance**

I. Introduction

The way of interacting with machines is continuously changing, since gestures, movements, and direct graphical user interface manipulation co-exist with physical interfaces like keys, buttons, and knobs. In the musical context, these new interface technologies have given us countless possibilities in the creation of Digital Musical Instruments (DMI) or New Interfaces for Musical Expression (NIME), artifacts that connect inputs (interface controllers) and outputs (sound synthesis modules) according to a mapping strategy [1]. A DMI comprises a control interface and an audio engine, which usually work separately [2]. The communication between these two components is made possible using some industry-standard communication protocols such as Musical Instrument Digital Interface (MIDI) and Open Sound Control (OSC). This is a considerable difference compared to traditional instruments, where the musical gesture and the sound generation take place on the same instrument [3]. In the case of a DMI, the design of the Human-Computer Interaction (HCI) component is important to give a sense of musical meaning not only to the artist but also to the audience to support their experience. The main goal in designing such a system is to make the HCI and software engineering tasks work together seamlessly [4].

While early DMI approaches were usually mostly leaded by (or with) the artists themselves [5]–[8], more recently the trend has inverted and DMI design has been usually practiced by researchers having backgrounds in human-computer interaction and/or music technology. Nowadays, far more rare are approaches where the design is led by or with performers [9]. A few works addressed this issue, such as [9] where percussionists can personalize a pre-designed adaptable DMI built through Arduino Uno, or [10], where a previously designed embedded DMI, the *Noisebox* [11], is redesigned through a user-driven procedure consisting of several workshops.

In this manuscript, we follow a similar yet different approach in the sense that the design of the DMI is tailored from the start around a specific musician through workshops, an approach partially similar to [12], where workshops with expert musicians are carried out in order to imagine design practices for DMI. However, the main difference in the work described in this paper is that the objective of the DMI design is to do the opposite of general purpose design and, instead, to create an instrument that, from conception to realization meets the particular needs of the single musician.

Specifically, we design a DMI performance tool tailored to the needs of the Italian jazz singer, Maria Pia De Vito[1] [13], [14]. In her recent musical practice, the artist experimented with the use of standalone loopers for solo vocal performances, yet she felt them as intrusive and not fulfilling all of her artistic needs[2].

Taking into consideration the artist's needs based on the nature of her live performance and following her performance desires and constraints, we built a custom harmonizer to

[1] Maria Pia De Vito Wikipedia page

[2] Personal communication between us and the artist, which created the motivation behind the realization of the Handmonizer.

enhance her solo performance, a DMI with which she can interact creatively and intuitively. While many definitions of DMIs are possible [15], in the case of the Handmonizer, we follow the definition provided in [16] where the instrument is structured into a *control interface* (hand detection), *processor* (audio engine) and *output* (harmonized voices).

Collaborating and discussing her necessities, we developed the Handmonizer, an unusual and (we believe) unique harmonizer controlled in real-time by hand gestures. Just by moving her hand, the artist can explore the different screen areas each associated to a specific system setting, creating different harmonic combinations by exploiting the smooth transition between the different voices. At the same time, she can control the number of different effects (e.g., Reverb or Delay) using fixed intuitive hand movements. This interaction strategy allows the artist to be focused on her moment-to-moment musical intention while performing, instead of having to think about switching between physical devices that slow the flow of the performance and disrupt musical improvisation.

The use of *air instruments*, i.e. musical instruments that are not physically touched by the user, has already been considered in the literature [17], [18] as well as in the Internet of Musical Things context [19], [20], but, to the best of our knowledge, has not yet been considered in user-centered personalized DMI design.

While the proposed Handmonizer can be still considered as a work-in-progress, its modular nature makes it extremely suitable in order to be extended to scenarios related to Networked Music Performances [21], since the control interface and the processor are connected via OSC.

The rest of the paper is organized as follows. In Sec. II we describe the process of cooperation with artist in order to set the desired design goals for the instruments. In Sec. III we give details on the technical aspects of the Handmonizer, presenting both the tools and methodologies we used to develop the final instrument. Sec. IV describes the evaluation procedure performed by the artist after the design of the Handmonizer in terms of a questionnaire and open comments through which she provided detailed feedback on every aspect of her experience with the Handmonizer after experimenting with it during two workshop sessions (each one of several days). Finally, in Sec. V we draw some conclusions from our collaboration to support future projects in this area. The full code used to implement the Handmonizer is available on the GitHub repository [3] and also a demo video of Maria Pia De Vito performing with the Handmonizer [4].

II. Artist-oriented system design

In this section, we introduce the development process based on the constraints posed by the needs of the artist, the discussions and the decisions taken based on her feedback. The Handmonizer system consists of a vocal harmonizer and of a hand gesture recognition interface able to control specific parameters in real-time depending on the movements of the singer's hand.

A. The Artist needs

The design of the handmonizer was not pre-prototyped. Instead, during the first meeting with the artist, she presented us with the technical, musical and performative challenges she encountered while using her previous setup, and her ambitions for using technology to fulfil a vision of her ideal performance scenario.

The previous setup used by Maria Pia De Vito (widely recognised as one of Europe's leading singers across jazz and improvised music) consisted of an Eventide® harmonizer guitar pedal and of a Gibson® Echoplex looping machine. Through the use of these two machines, she was able to create independently the backing tracks along which she used to improvise during her solo performances[5]. The main issues she faced using the described setup, were related to the unnatural and mechanic interaction due to the hardware tools. Her performances often require the modification of the parameters of the hardware through various knobs and switches, which is felt as a slow and cumbersome modality that reduces her creativity. At the same time, dealing with physical devices, like the Eventide® Harmonizer, implies the generation of clicking noises caused by the pressure of the knobs that disrupt the musical flow felt during the performance.

In the light of these considerations, the main challenge we mutually decided to consider was to develop an interface that would be easy for her to use so that she could mainly focus on her improvisation-based performance without having to struggle with the setup interaction. Therefore, we decided to implement a hand-controlled vocal harmonizer which gives her the opportunity to explore and experiment with the different harmonic configurations in a natural and simple way, just by moving her hand in front of a camera.

B. Interaction Design approach

Differently from acoustic instruments, where the physicality of the musical instrument imposes specific ways to control it, in digital musical instruments, it is extremely important to consider in detail the possible mapping strategies, due to the bigger amount of possible choices caused by the separation between the input gesture and output sound [22]. The choice of different mapping strategies also affects how the performer can react both musically and psychologically when playing the instrument. [1]. In general, when designing a DMI we need to follow three main principles [23]:

- **Control:** The way the interaction is mapped to the output, which should give the required amount of control to the artist who is using the system.
- **Legibility:** The importance of having a mapping that is easily understood by both the artist and the audience. A concept often referred to as "transparency" [24].

[3] GitHub repository
[4] Handmonizer Video Performance

[5] Maria Pia De Vito solo performance

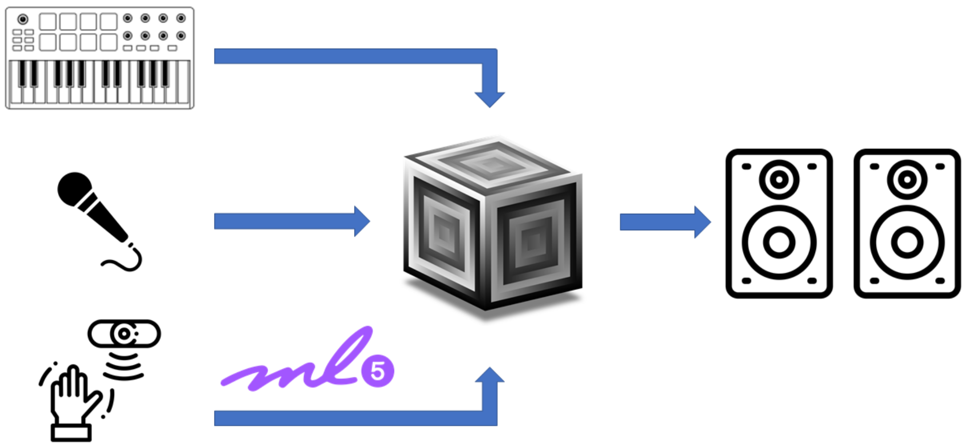

Fig. 1: Brief schematics of the Handmonizer signal flow. A simple MIDI controller (top left) is used in order to switch between the patches used in the Handmonizer by sending MIDI messages to the SuperCollider engine. A microphone (middle left) is used to acquire the voice of the singer, whose audio signal flow is then sent to the SuperCollider engine. On the bottom left, hand recognition is performed on the hand of the singer and the extracted values sent via OSC messages and used to control the voice of the singer through SuperCollider. Finally, the output audio signal processed by the SuperCollider part of the Handmonizer is sent to external loudspeakers (right).

- **Sound:** The choice of what kind of sounds the mapping is going to produce. This could mean creating new sounds, processing an input sound as well as using combination of both.

Moreover, when the design of a DMI interface must be tailored to the needs of a specific artist, the instrument designers should put themselves in the perspective of the artist as much as possible in order to develop something that is actually meaningful and usable in her performance.

A good starting point when designing the gesture-sound mapping is to use a bottom-up approach. Specifically, in the case of the Handmonizer, we started by considering very simple interactions, such as opening and closing the hand, in order to use them to trigger an intuitive response, like turning on or off the harmonizer, at the same time we thought how we can make these interactions meaningful for the audience. Audience engagement is strongly related to the mapping problem. Since the gestures and the produced sound have not necessarily a direct connection, the audience may not feel engaged during the performance if the mappings are not thoughtfully chosen [1], [18]. Previous work has tried to centre-stage the lived experiences of performers and audiences when working with improvising using technological systems [25].

III. Handmonizer Implementation

In this section we detail the design choices and the architecture of the proposed Handmonizer DMI. The general architecture of the Handmonizer structure is based on a Node.js®[6] server that allows the communication between the two main parts of the instrument: the hand gesture recognition engine and the real time sound harmonization algorithm, communicating through OSC messages.

We first present the general elements of the Handmonizer, specifically in Sec. III-A we outline how the hand gesture recognition is performed, while in Sec. III-B and Sec. III-B1 we discuss the harmonizer-part implementation and the corresponding pitch-tracking and pitch-shifting algorithms. We developed two different versions of the Handmonizer, the first one, the *patch-based* Handmonizer, described in Sec. III-B2 is based on a series of pre-defined patches that allow the artist to select via a MIDI controller the intervals considered by the harmonizer. Following the suggestions provided by the artist, we modified the Handmonizer in order to more closely follow her needs, specifically, we developed an *In-scale* version of the Harmonizer, described in Sec. III-B3, where a specific musical scale is chosen by the artist and is used in order to perform the harmonization of the voices. We present how the different voices of the harmonizer can be mixed together in Sec. III-B4 and finally, in Sec. III-C we describe how the various components of the Handmonizer are connected together.

A. Hand gesture recognition

The harmonizer part of the Handmonizer can be controlled through hand gestures captured from a webcam. For the hand motion recognition, we use a pre-trained model from ml5.js

[6]https://nodejs.org/en/about

[7], more specifically, consisting of a JavaScript framework for creative coding built on top of TensorFlow.js that allows the use of GPU-accelerated machine learning algorithms in a web browser. The TensorFlow ecosystem provides an easy-to-use tool to convert pre-trained ML models trained in Python or C++ into web targets. Some of these models were specifically designed for creative applications to facilitate the development of real-time music related Web-applications [26]. We use the model called *Handpose*[8] that performs hand-skeleton finger tracking. It takes the video stream frame by frame and returns the 3D coordinates of $I = 21$ hand points $\mathbf{r}_i, i = 1, \ldots, I$ over the palm of the hand[9] as shown in Figure 2. Based on the coordinates of the I detected points we compute three main parameters:

- The hand centroid $\mathbf{r_c} = [x_c, y_c, z_c]^T \in \mathbb{R}^3$ which is computed as the arithmetic mean position of the I keypoints. It is displayed as the light-green central dot in Fig. 1.
- The palm length d defined as the distance between the keypoint corresponding to the tip of the middle finger and the keypoint corresponding to the base of the palm (as the length of the white line shown in Fig. 1).
- The hand orientation: computed as the slope of the line used to measure the palm length line, between $[0, \pi]$.

By applying a custom mapping to these three parameters we define 5 hand gestures that we can use to change the harmonizer settings in real time. Through the palm length parameter we can detect if the hand is open or closed, through this we control the fading out of the harmonics by closing the hand. Using the z_c coordinate of the hand centroid, which corresponds to the distance of the hand from the screen, we control the volumes of the harmonized voices. The palm slope controls the amount of effects (Reverb or Delay) that the user wants to add to the voice, similarly to a dry-wet knob; when the white line is perpendicular to the bottom border the voice has no effect, while it reaches its maximum value when the line is parallel to the lower border.

B. Harmonizer implementation

The audio signal processing part of the Handmonizer is developed entirely using the SuperCollider audio programming language. The algorithm of the harmonizer part is composed of separate sections, including pitch tracking and shifting, effects definition and communication protocol definitions (OSC, MIDI) for parameter mapping. Here we explain each component separately.

1) Pitch tracking and shifting: The first part of the (operation of the hand) harmonizer requires an analysis of the input singing voice signal in order to then generate the harmonized voices (i.e. harmonically pitch-shifted version of the input signal). Specifically, since we aim to use pitch shifting in order to create harmonic voices, the first part of the Handmonizer

[7]ml5.js: Friendly machine learning for the Web.https://ml5js.org

[8]https://learn.ml5js.org/#/reference/handpose

[9]This process is GPU intensive. To achieve the best performance a system with a dedicated GPU is advised.

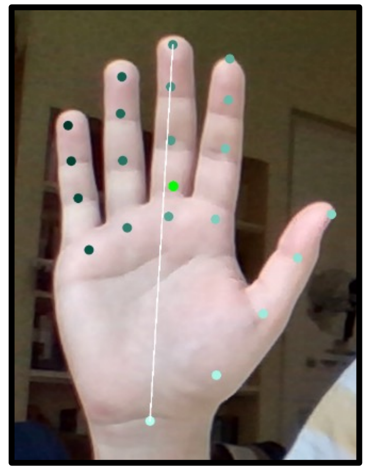

Fig. 2: Output of the hand recognition procedure used to control the Handmonizer. The hand is mapped by the Handpose model into 21 points, the white line connects the base of the palm to the top of the middle finger and is used to calculate the palm length.

audio engines requires us to track the fundamental frequency of the input audio signal. For this purpose we use an external class called *Tartini* instead of the standard Pitch class from SuperCollider, since it performs the pitch tracking more precisely and in a more efficient fashion [27].

To avoid obtaining vocal harmonies that may sound too unnatural to the singer, the pitch shifting is performed using the *Pitch-Synchronous Overlap and Add* (PSOLA) method [28]. One of the main advantages of this method is the preservation of the formant positions (i.e. spectral envelope) which allows us to maintain the original timbre [29]. In order to do this we use a SuperCollider class called *PitchShiftPA* which is based on PSOLA. The fundamental frequency value tracked by *Tartini*, as well as the pitch shift ratio, which we fix depending on the harmony we want to achieve, are given as input to the PSOLA pitch-shifting model. The same procedure is then used to generate six harmonic voices, namely three higher and three lower than the input signal. The harmonic voices are then mixed together in a separate channel, so that they can be further processed without modify the lead original singing voice.

2) Patches-based Handmonizer : The first version of the Handmonizer, is based on a definition of a series of predesigned patches, which differ by which intervals are considered in order to generate the harmonized (i.e. pitch shifted) voices. Specifically, this version of the Handmonizer, first detects the fundamental frequency of the singing voice and considers this as the root of the scale, then, depending on number of intervals chosen it generates $2/3$ harmonics immediately higher than the root note and, in addition, the same harmonics pitched at an octave lower. We define 4 patches each considering fixed musical intervals for the harmonizer, following the specifications given by the artist: 3rd - 5th; 4th - 5th - 7th; 4th - augmented 4th - 7th; octaver. For example,

(a) (b)

Fig. 3: Testing the Handmonizer during the workshops at Politecnico di Milano, Milan, Italy.

if the pitch detected from the singing voice is a C4, then the system considers the C major scale, if the chosen patch contains the 3rd - 5th intervals, then the corresponding 3rd is a E and the 5th a G. The two higher voices will then correspond to E4 and G4, while the lower ones to E3 and G3.

All the patches also include a delay effect that can also be controlled by the user while rotating the hand.

Two additional patches where no harmonized voices are present are also include to fit the needs of the artist, specifically, one with only a reverb effect and another one with only delay, where both effects are controllable by simply rotating the hand. These two patches were designed due to the need of the singer to perform also single voice improvisations over looped backing tracks.

3) In-scale harmonizer: Following the suggestions provided by the artist during the first workshop, we also implemented a second version of the Handmonizer, where instead of simply considering fixed intervals, the Handmonizer can be used as a classic harmonizer following a specific scale, where the user can set the key and scale type (major, minor, etc). By hard-coding the first MIDI note for each key, we use an external class called *MiscFuncs* to retrieve the array of MIDI notes for the selected scale. Then we use another external class called *MyKFiddle* to retrieve in real time the precise MIDI note sang by the singer. Finally, the algorithm checks if the pitch of the sound emitted by the singer corresponds to a note that is part of the selected scale. If this is the case, it computes the pitch ratio and feeds it to the pitch shifter.

The two classes mentioned above[10], which contain all the intervals for each scale type, were used in order to perform all the scale-related calculations needed to create the in-scale harmonizer model.

4) Cross-fading: In order to make the Handmonizer as much musically usable as possible, we ensure a smooth transition between different generated voices, namely by avoiding sudden jumps in volume when the new harmonized voices are added to the mix by the artist. Here we manipulate the different amplitudes of all voice groups before mixing them. A similar procedure is used for the overall volume fader of the harmonies for which we use a dB scale mapping to approximate the behaviour of the human auditory system. We use the SuperCollider class *XFade2* as a cross-fade knob to control the dry/wet ratio used for the reverb and delay effects.

C. Communication protocols and architecture

To switch between patches, we use a simple MIDI controller where we assign each pad to a patch by changing the necessary parameters. In addition to the patches mentioned above, we use one pad as an ON/OFF toggle button and another pad as a bypass for the harmonic voices.

The user interface is hosted as a web page/application in an Express server, the connection is set up through the framework Socket.io[11]. All the control parameters mentioned above are computed on the laptop running the Handmonizer main engine and then sent to the server (remote host performing the machine learning calculations). The server, writes the parameters in OSC messages and forwards them to SuperCollider.

The hand motion recognition features are sent to SuperCollider as OSC messages in real time and are used to control different parameters that define the harmonizer's performance. As described before, The palm centroid is tracked into the 3D domain as $\mathbf{r}_c = [x_c, y_c, z_c]^T$. We use the x_c coordinate in order to add more harmonized voices as we move from left to right. The selection of the voices is mapped by subdividing the screen into two or three, depending on the number of chosen harmonics, columns of equal size. Each time the user visits a new column, another harmonic voice is added. The screen is also divided into two rows, such that the upper

[10]https://github.com/yeeking/myksupercollider

[11]https://socket.io/

TABLE I: Evaluation questionnaire provided to the artist. Some questions ask both a grade and a comment while a few of them are open questions.

Question	**Grade**	**Answer**
(1) Playability: How much control do you feel that you have over the tool while using it?	8/10	*I still depend on the presence of an expert to manage it.*
(2) Learnability: How easy was learning to use it?	9/10	*It takes a little practicing, but the use is pretty intuitive, so very good.*
(3) Expressiveness: How much does it help you to enhance your creativity and express your musical intention?	9/10	*With some practice the result could be amazing!*
(4) Enjoyability: How enjoyable was your experience while using the Handmonizer?	9/10	*Very enjoyable. I need practice.*
(5) Novelty: How much novelty does it introduce to your performance?	10/10	*The use of hand movements instead of just pressing buttons and turning knobs is pretty new and enjoyable for me. I move a lot while improvising!*
(6) Effectiveness: How much does this tool manage to solve the issue with your previous setup?	8/10	*The problem for me is still to enhance the previous, pretty analog setup! Other than that, it opens up a different field of action, pretty liberating!*
(7) Sound quality: Rate the perceived sound quality obtained through the Handmonizer setup.	9/10	*The perceived sound quality is very good. I still need to bridge some gaps with the previous setting with practice.*
(8) How much were we able to understand your needs and fulfill them?	9/10	*Very good job in this sense! We have been in touch and consulting each other on every step of the way.*
(9) How much were we able to improve the Handmonizer following your feedback after the first workshop?	10/10	
(10) Can you describe your experience of being involved in the development of your own custom instrument (expectations, results, impressions)?		*It has been a very inspiring experience. Still a work in progress for me, having to create instantly, improvising, new soundscapes. I hope to be able to use it in live performances.*
(11) How confident would you feel using it in a live performance?	7/10	*I will need the presence of an assistant. I am not a "nerd," but an intuitive musician. It would be reassuring to have somebody able to manage more complex issues.*
(12) What are the main limitations that you would face if you had to use the instrument on your own?		*As I said, I am afraid of dealing with the setup and possible issues, having for instance to re-launch the system in case of problems.*
(13) What should we carry on doing, and what could we do better, in order to work with artists like you to develop cutting edge performance technologies that you would use on concerts?		*To describe technicalities is a bit out of my strict competence. My wish is that the system becomes pretty "stable," so to become more and more independent from the help of others. I guess this is something that any creative artist would wish for himself/herself. But please, carry on doing "hands-on" work with artists!*

one corresponds to harmonic voices higher than the original voice and the bottom one to lower harmonics. We can thus imagine the screen as divided into two rows where the upper row represents the high octave harmonics and the bottom row represents the low octave harmonics. All these changes in number of voices and octaves are mixed together in a smooth way as explained previously, so that the artist can explore different sounds without any abrupt changes that would impact any sense of flow.

D. Additional Controls

Another feature used is the palm length represented by the white line depicted in Fig. 1. This feature is mapped to the harmony fader using a dB scale to control the volume of the shifted voices. There are two ways to exploit this feature. The first and most intuitive way is to open and close the hand in order to turn on or off the device, while the second one is to move our hand back and forth to change the volume mix of the harmonized voices with the lead one. If the artist wants to emphasise the harmonic voices they can simply move their hand closer to the camera, if they wants to fade out the harmonies they only needs to gently close their hand.

Finally, we use the hand orientation as an imaginary knob that controls the dry/wet level of the reverb or delay effects. When we keep our hand straight we have a fully dry signal (e.g.: no effect). While instead, if the hand is rotated either left or right it is possible to change the amount of the wet signal and decrease the amount of the dry signal using a cross-fade effect.

IV. Testing and evaluation

In this section we describe the evaluation procedure performed in order to understand how the custom-design of a DMI is beneficial to the musician. In Fig. 3 we presented some photos of Maria Pia De Vito testing the Handmonizer during our collaborative workshops. In order to properly evaluate the instrument and to understand the nature of the artists experience in co-designing with us and the resulting musical experience itself, we designed a questionnaire following the typical DMI evaluation methodologies [30], [31] focusing our attention on the final singer's performance and on the ability of the system to fulfill the specific initial ambitions and constraints. Since "*there is no one-size-fits-all solution to evaluating DMIs*" [32] it is hard to adopt an evaluation strategy consistent with similar type of works. Instead, we opted to design the questions by both analyzing some constructs typical of the DMI research community (e.g. playability, expressiveness, effectiveness, etc.) and by inserting some questions analyzing the specific case considered in our scenario, specifically the human-centered design procedure of the Handmonizer. On some questions we asked the artist to provide simple grade from 0 to 10 and a short comment, while on others she was asked to provide a longer feedback, by answering to open questions.

The questionnaire and the answers are contained in Table I.

The answer to the questionnaire provide extremely useful feedback that could be applied in any future project looking to design of digital musical instruments. Specifically, they demonstrate that the personalize human-centered design is appreciated by the artist which feels as part of the whole instrument creation procedure, suggesting that focusing on the artist needs is a viable and desirable option when designing DMIs. While the questions proposed in Table I are specific to the work presented in this paper, we believe that the focus on aspects concerning the relationship of the artist with the instrument would be useful to the wider DMI-making community. For example, understanding if not only the instrument is easy to use, but also if it suits the needs of the artist (cfr. question 8) is an important aspect which we believe should be one of the main focuses when designing DMIs.

During the two workshops, we noticed just how quickly the artist learned to interact with the tool and within a few trials she was able to come up with extremely interesting vocal solos. She was not impeded by the interaction provided by the Handmonizer, instead she was stimulated to find new ideas and patterns by moving freely her hand.

V. Conclusions

In this paper, we have presented the design of a hand-controlled digital musical instrument, specifically tailored for the Italian jazz singer Maria Pia De Vito. The interesting aspect of our approach is that the design of the instrument is not made by the creators of the instrument, but instead is based on a human-centric design approach by following from start to finish by the guidelines provided by the musician, starting from the conceptualization of the instrument and continuously adapting it according to her requirements.

We have explained in detail how we approached the design steps, and how we centred-staged the needs of the artists and were continually open to discussion with, and feedback from, the artist. We also presented the methodology used in order to build the Handmonizer in terms of technology and architecture. The instrument can be described as a fully working harmonizer which is controlled by following the hand gestures of the musician. Through a questionnaire and discussions with the singer, we evaluate the effectiveness of the Handmonizer and of the chosen special-purpose design procedure.

There are a number of technological improvements that could be made in order to make the Handmonizer more usable, specifically, right now it is difficult for the artist to run the application on her own. To solve this problem, now the singer is supposed to have a technician to set-up and run the tool, while future developments will include the development of a standalone tool. However, we believe that this work shows that a possible path for the future creation of DMIs might not be focused on general-purpose design techniques, but, instead on the creation of instruments that are specifically personalized for the requirements of the artist, which in our opinion is a very promising way of fruitfully merging the power of the DMI technology with the subjective needs posed creative endeavours of the artists.

References

[1] R. Medeiros, F. Calegario, G. Cabral, and G. Ramalho, "Challenges in designing new interfaces for musical expression," in *International Conference of Design, User Experience, and Usability*. Springer, 2014, pp. 643–652.

[2] E. R. Miranda and M. M. Wanderley, "New digital musical instruments: Control and interaction beyond the keyboard (computer music and digital audio series)," 2006.

[3] I. Bergstrom, A. Steed, and R. Lotto, "Mutable mapping: gradual rerouting of osc control data as a form of artistic performance," 01 2009, pp. 290–293.

[4] J. Borchers and M. Muhlhauser, "Design patterns for interactive musical systems," *IEEE MultiMedia*, vol. 5, no. 3, pp. 36–46, 1998.

[5] M. Waisvisz, "The hands: A set of remote midi controllers," in *Proceedings of the International Computer Music Conference: International Computer Music Association, 1985*, 1985.

[6] R. Fiebrink and L. Sonami, "Reflections on eight years of instrument creation with machine learning," 2020.

[7] T. Machover, "Hyper-instruments: Musically intelligent and interactive performance and creativity systems," in *ICMC Proc.*, 1989, pp. 186–190.

[8] D. Morrill and P. R. Cook, "Hardware, software, and composition tools for a real-time improvised solo trumpet work," in *Proc. 1989 International Computer Music Conference*, 1989, pp. 211–214.

[9] T. Roth, A. Huang, and T. Cunningham, "On Parallel Performance Practices: Some Observations on Personalizing DMIs as Percussionists," in *NIME 2021*, apr 29 2021, https://nime.pubpub.org/pub/226jlaug.

[10] J. Sullivan, J. Vanasse, C. Guastavino, and M. Wanderley, "Reinventing the noisebox: Designing embedded instruments for active musicians," in *Proceedings of the International Conference on New Interfaces for Musical Expression*, 2020, pp. 5–10.

[11] J. Sullivan, "Noisebox : Design and prototype of a new digital musical instrument," in *Proceedings of the International Computer Music Conference, Denton, TX, USA*, 2015, pp. 266–269.

[12] J. Sullivan, M. M. Wanderley, and C. Guastavino, "From fiction to function: Imagining new instruments through design workshops," *Computer Music Journal*, pp. 1–46, 2023.

[13] S. Marino, "Jazz, libertà ed emancipazione femminile (con un'intervista a maria pia de vito)," *Jazz, libertà ed emancipazione femminile (con un'intervista a Maria Pia De Vito)*, pp. 341–370, 2018.

[14] M. P. De Vito, "The jazz life of maria pia de vito," pp. 17–17, 2022.

[15] E. Frid, "Accessible digital musical instruments—a review of musical interfaces in inclusive music practice," *Multimodal Technologies and Interaction*, vol. 3, no. 3, p. 57, 2019.

[16] J. Pressing, "Cybernetic issues in interactive performance systems," *Computer music journal*, vol. 14, no. 1, pp. 12–25, 1990.

[17] Y. Ikawa and A. Matsuura, "Playful audio-visual interaction with spheroids," in *Proceedings of the International Conference on New Interfaces for Musical Expression*, R. Michon and F. Schroeder, Eds. Birmingham, UK: Birmingham City University, Jul. 2020, pp. 188–189. [Online]. Available: https://www.nime.org/proceedings/2020/nime2020_paper36.pdf

[18] J. Leonard and A. Giomi, "Towards an interactive model-based sonification of hand gesture for dance performance," in *Proceedings of the International Conference on New Interfaces for Musical Expression*, R. Michon and F. Schroeder, Eds. Birmingham, UK: Birmingham City University, Jul. 2020, pp. 369–374. [Online]. Available: https://www.nime.org/proceedings/2020/nime2020_paper72.pdf

[19] D. Cocchiara and L. Turchet, “Democratizing access to collaborative music making over the network using air instruments,” in *Proceedings of the 17th International Audio Mostly Conference*, 2022, pp. 211–218.

[20] L. Turchet, C. Fischione, G. Essl, D. Keller, and M. Barthet, “Internet of musical things: Vision and challenges,” *IEEE access*, vol. 6, pp. 61 994–62 017, 2018.

[21] L. Turchet and C. Rottondi, “On the relation between the fields of networked music performances, ubiquitous music, and internet of musical things,” *Personal and Ubiquitous Computing*, pp. 1–10, 2022.

[22] J. B. Rovan, M. M. Wanderley, S. Dubnov, and P. Depalle, “Instrumental gestural mapping strategies as expressivity determinants in computer music performance,” in *Kansei, The Technology of Emotion. Proceedings of the AIMI International Workshop*. Citeseer, 1997, pp. 68–73.

[23] T. West, B. Caramiaux, S. Huot, and M. M. Wanderley, “Making Mappings: Design Criteria for Live Performance,” in *NIME 2021*, may 3 2021, https://nime.pubpub.org/pub/f1ueovwv.

[24] S. Fels, A. Gadd, and A. Mulder, “Mapping transparency through metaphor: towards more expressive musical instruments,” *Organised Sound*, vol. 7, no. 2, pp. 109–126, 2002.

[25] M. Yee-King and M. d’Inverno, “Experience driven design of creative systems,” in *Proceedings of the Seventh International Conference on Computational Creativity (ICCC 2016)*, F. Pachet, A. Cardoso, V. Corruble, and F. Ghedini, Eds., June 2016, pp. 85–92.

[26] A. Correya, P. Alonso-Jiménez, J. Marcos-Fernández, X. Serra, and D. Bogdanov, “Essentia tensorflow models for audio and music processing on the web,” in *Web Audio Conference (WAC 2021)*, 2021.

[27] E. Kermit-Canfield, “A comparison of real-time pitch detection algorithms in supercollider,” *Journal of the Audio Engineering Society*, october 2014.

[28] F. Charpentier and E. Moulines, “Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphones.” in *Eurospeech*, vol. 89, 1989, pp. 13–19.

[29] N. Schnell, G. Peeters, S. Lemouton, P. Manoury, and X. Rodet, “Synthesizing a choir in real-time using pitch synchronous overlap add (psola),” in *ICMC*, 2000.

[30] J. Barbosa, J. Malloch, M. M. Wanderley, and S. Huot, “What does” evaluation” mean for the nime community?” 2015.

[31] P. J. C. Reimer and M. M. Wanderley, “Embracing Less Common Evaluation Strategies for Studying User Experience in NIME,” in *NIME 2021*, apr 29 2021, https://nime.pubpub.org/pub/fidgs435.

[32] S. O’modhrain, “A framework for the evaluation of digital musical instruments,” *Computer Music Journal*, vol. 35, no. 1, pp. 28–42, 2011.