AN INTERACTIVE SYSTEM FOR GESTURE-CONTROLLED VOCAL IMPROVISATION WITH SOMAX2: CULTURAL NARRATIVE IN MINOTAUR'S LAMENT

Pingan YAO (patrickyao@campion.edu.gr) (0009-0007-6655-462X) [1]

[1]**Campion School**, Athens, Greece

ABSTRACT

This paper presents the design and implementation of a new system for human-computer interactive performances, integrating gesture recognition for live vocal improvisation with Somax2, particularly within cultural contexts. Using the MediaPipe framework in Python for real-time hand tracking, performer gestures are mapped to live signals, which are transmitted to the Max/MSP environment with the Open Sound Control (OSC) protocol. The received signals are mapped to parameters for controlling live vocal effects, which influence Somax2's generative output.

We then demonstrate the system's usage and performance workflow with *Minotaur's Lament*, an improvisation inspired by Greek mythology and Ancient Greek musical forms. Mapping gestures to pitch shift, we find that the proposed framework enables enhanced vocal expressivity and Somax2 musical variability. As an additional control layer for the Somax2 architecture, the system widens the improviser's creative flexibility, with potential for further applications in culturally-embedded interactive performances and computer music interfaces. The paper provides an overview of the system architecture, technical implementation of the performance, as well as the artistic background and intent.

1. INTRODUCTION

Since the development of AI-powered real-time computer vision tools in early 21st century, gesture control has become widely adopted in interactive music performances, particularly in the context of computer music. Performers began to experiment with the use of physical movement to shape sound synthesis, as exemplified by earlier studies of hand gesture-based musical parameter control using Time Delay Neural Networks (TDNN) [1].

Later developments saw the release of MediaPipe, an open-source computer vision framework. Based on Convolutional Neural Networks (CNN), it enables highly robust real-time gesture tracking [2]. Through the Open Sound Protocol (OSC) protocol [3], MediaPipe has been extensively integrated into the Max/MSP architecture for movement-based audio-visual interactions, such as SoniSpace [4] and NeoLightning [5]. Within the domain of finger tracking, works such as MuGeVI [6] have expanded gesture control into the creation of multi-functional virtual instruments.

Additionally, computer-based interactive music has been studied in cultural contexts, with the creation of performance systems such as LOERIC for Irish traditional dance music [7]. Gesture control, with significant potential in such applications, has also been integrated in digitising traditional instruments, such as the Chinese erhu [8].

However, most gesture-controlled computer music architectures have been designed for direct sonic interaction and control. Until recently, gesture control has not been widely incorporated in human-machine co-creative performances, especially in the context of real-time AI-based musical composition and improvisation agents such as Somax2. Such use cases are even more rarely integrated into culturally-embedded music performances. Kinesthesis 2.0 [13, 17] was proposed as the first gesture-controlled Somax2 improvisation architecture using a MediaPipe-OSC-Max/MSP pipeline. Drawing on ancient Greek tragedies and theatrical practices, it enabled interactive vocal improvisation embodying gestures and movements inspired by traditional schemata, while extending performance expressivity and cohesion through using Greek music as musical material for Somax2. Nevertheless, it primarily focused on the relationship between gestures and vocal effects, which were not specifically designed to influence Somax2 behaviour. The process by which gesture-controlled sounds can directly interact with creative AI musical agents remains unexplored.

On the other hand, existing computer music works inspired by the Greek culture have taken various forms, often featuring elements drawn from Greek mythology [12, 22]. However, such works have primarily been realised as fixed media, rather than real-time musical interactions and improvisations.

Thus, building on previous work in the fields of music gesture control and human-machine interaction, our study aims to present a new, interactive pipeline for vocal improvisation that connects sound, gestures and machine creativity into an interlinked and culturally-embedded performance. Expanding upon [13], our system structurally integrates gesture-modulated vocal signals into Somax2's generative decision process.

Specifically, we will use the figure of Minotaur from Greek mythology as our thematic foundation. Using hand gestures for pitch-shift control, we attempt to replicate the voice of Minotaur, the half-man and half-bull monster [19]. Meanwhile, the mystery of the labyrinth [21] is recreated as a soundscape by Somax2, which is influenced by the

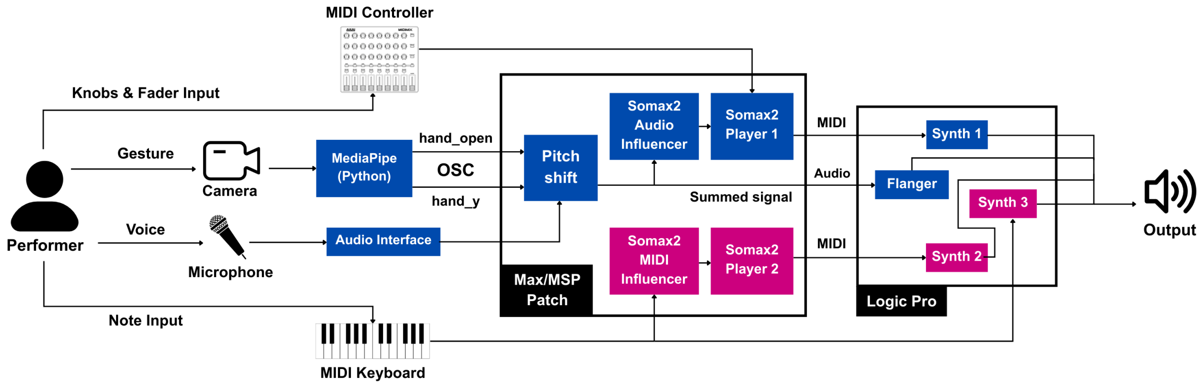

Figure 1. System architecture of the proposed interactive system for live improvisation with Somax2. The performer(s) carries out four main tasks: vocals, hand motion for gesture control, MIDI keyboard input, and knob & fader control for Somax2. The voice and gesture control branches are described in Section 2, while the other components of the architecture are detailed in Section 3.

modulated voice in real time.

Through this system, we aim to widen the application of gesture control in interactive computer music performances, and thus extend the use of Somax2 in playing a more active role for shaping cultural soundscapes, as inspired by Greek mythical archetypes. Nonetheless, the architecture used in this research is designed to be equally applicable in non-cultural contexts, and is aimed at contributing to the wider field of human-computer interaction for musical expression.

2. INTERACTIVE SYSTEM FOR GESTURE CONTROL

At the core of the system architecture is a pipeline linking real-time hand tracking to the pitch shift effect on live vocals. This is implemented through a pipeline running from OpenCV and MediaPipe in Python into a custom Max/MSP patch for further sound processing. Previous gesture control pipelines based on MediaPipe and Max/MSP have been found to yield reliable results, with [6] highlighting the system's low latency and flexibility as an advantage in its live performance applications.

2.1 Real-Time Hand Tracking

MediaPipe is an open-source, cross-platform framework for real-time computer vision. Specifically, we used MediaPipe Hands, a library that provides robust hand-tracking solutions, in combination with OpenCV for camera capture and image conversion. The MediaPipe layer is processed through Python, which provides cross-platform stability for our architecture. Our study was conducted using a webcam with 1080p resolution at 30 fps, connected to an M2 MacBook Air for MediaPipe tracking and all other software processing.

Using our custom Python program, we implement gestural design using MediaPipe's Hand Landmark Model, as shown in Fig. 2. Based on the landmark indices denoting various hand regions, we define two parameters, which are later used for gesture control mapping:

- Vertical hand position (`hand_y`): the y-coordinate of 0 (wrist), within the camera frame. This is computed through MediaPipe's normalised image coordinates, which return $y \in [0, 1]$ moving from the top to the bottom of the frame. To reduce edge sensitivity, only the middle 80% of the camera's vertical range is active, while the outside range is clamped to 0 or 1.

- Pinch distance (`hand_pinch`): the Euclidean distance between 4 (thumb tip) and 8 (index tip), computed using the two landmarks' respective x and y positions as a fraction of the frame size. Distances ≤ 0.02 return a value of 0, while distances ≥ 0.18 return 1. Distances between the two values are mapped linearly between 0 and 1.

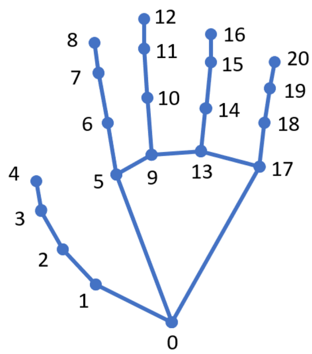

Figure 2. The hand's 21 anatomical landmarks, as tracked and processed by MediaPipe Hands in real time using numbers 0-20. [10]

In addition, we limit MediaPipe to track only one hand, as the performer's non-gesture hand may create disturbances to the program's output during performance.

The two parameters, `hand_y` and `hand_pinch`, are subsequently sent as Open Sound Control (OSC) messages via the User Datagram Protocol (UDP), transmitting values to Max/MSP in real time for gesture-based sound processing.

2.2 Pitch Shift

We choose to experiment with the voice in our study, as it draws on a traditional musical form, the Ancient Greek lament, a mourning song typically accompanying ancient burial customs [11]. Applying it to Minotaur's voice enables an extension of Greek musical traditions into mythological tropes. Furthermore, as the Greek lament is often sung by one solo performer, our choice of using voice mirrors such tradition, allowing a single user to engage with the entire performance through music and gestures, as their hands are not limited to playing a musical instrument.

We use a Max/MSP patch, shown in Fig. 3, as the central layer of our architecture, serving both as the effects processor and the user interface during performance. The live vocal input, V_{in}, is processed by Max/MSP through a pitch-shifting module, which is parametrised by the hand gesture input. Pitch shift widens the frequency range of the performer's voice, enhancing the improvisation's emotional expressivity [9], while mirroring Minotaur's vocality. In addition, it also influences the musical output of Somax2, which analyses the input sound's tonal and harmonic content as a major part of its musical decision-making framework [14].

We design the system to support pitch shifting within a frequency range of $\Delta f \in [-200, +200]$ hertz relative to the input voice. This is controlled by the vertical hand position, `hand_y`, where OSC values in the range $[0-1]$ are linearly mapped onto the frequency shift span. This mapping reinforces an embodied performance setup, whereby physical movement in height intuitively corresponds to pitch.

Meanwhile, the `hand_pinch` parameter is mapped onto the gain of the processed signal, $V_{\text{wet}} = \mathcal{P}(V_{\text{in}}, \Delta f)$, acting as a multiplier that controls the volume of the pitch-shifted voice. The Max/MSP patch then outputs a composite signal, V_{out}, which combines the wet channel with the clean signal with a 30% gain reduction, allowing the pitch-shifted vocals to be more prominent than the clean signal in the resulting combined output.

The overall signal processing in the Max/MSP module can be described by the following equation, where g is the gain multiplier dictated by pinch distance:

$$V_{\text{out}} = 0.7V_{\text{in}} + g \cdot V_{\text{wet}} \quad (1)$$

The composite signal is then received by Somax2's audio influencer for feature extraction, allowing gesture control to shape its corpus-based generative behaviour.

All source code used in this research is available in a GitHub repository. [1]

[1] https://github.com/pinganyao/minotaurs-lament

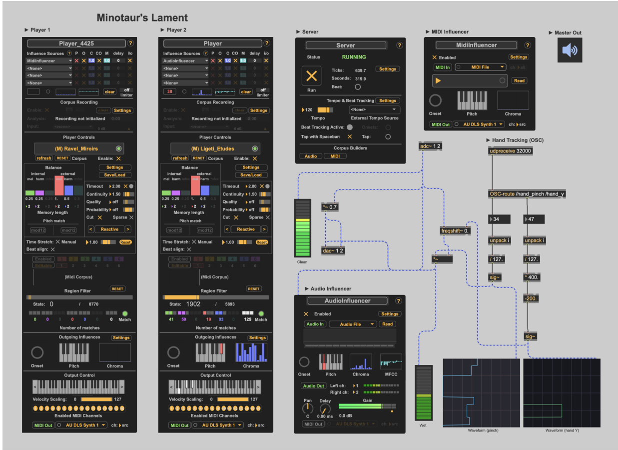

Figure 3. Max/MSP patch used for the performance. Gesture parameters are mapped to the pitch shift effect of the live vocals.

3. LIVE IMPROVISATION WITH SOMAX2 – PERFORMANCE ARCHITECTURE

To create an interactive and immersive experience while improvising with Somax2, as well as to maximise the degree of user control, a customised Somax2 setup is used, while incorporating additional components that are integrated into the performance architecture.

3.1 Somax2 Players

A Somax2 player is a module within the Somax2 system, which operates on a specific musical corpus [2], and autonomously generates its output based on an influence source. Each player acts as an independent improviser during the performance.

We use two Somax2 players in our setup, enabling the generation of a rich and multi-layered soundscape. As each player can provide distinct behavioural and timbral characteristics, our system also allows for increased musical variations, inviting performers to freely engage with musical co-creativity. Player 1 uses the audio influencer as its source, taking the gesture-controlled vocals as input to its musical generation. In contrast, player 2 is designed to receive input from the MIDI influencer, which is controlled by the performer through a MIDI keyboard.

Therefore, the Somax2 improvisation layer uses a configuration that supports a complementary structure: player 1 is the continuation of the main gesture control architecture, acting as the primary AI improviser responding to the live vocals; player 2 can be triggered with MIDI note inputs, enabling the performer to introduce additional sounds when desired.

3.2 Live Somax2 Parameter Control

During performance, the output of the Somax2 players are highly adjustable using their parameters [15, 16], which can be tuned to the input and the corpus, and are controlled

[2] In the context of Somax2, a corpus is the musical material used by the system to generate its output, in the format of audio or MIDI.

interactively by the performer in real time. In our performance, we facilitate this process by connecting a hardware MIDI controller (AKAI MIDImix) to Max/MSP, mapping the hardware controls to various parameters that influence the player output. This addition ensures that Somax2 parameter control remains accessible in solo performance contexts, as the performer can now easily adjust these settings using a physical interface.

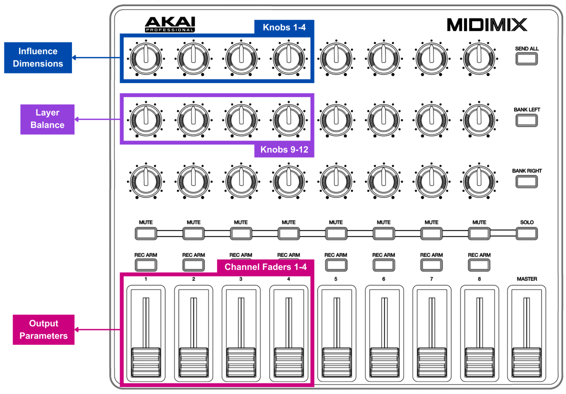

Figure 4. AKAI MIDImix controller, with hardware inputs mapped to corresponding Somax2 parameters: influence dimensions (knobs 1-4), layer balance (knobs 9-12), and corpus output (faders 1-4).

Fig. 4 illustrates the configuration we used on the AKAI MIDImix. For Minotaur's Lament, we mapped its knobs and faders to 12 different parameters controlling player 1, and can be additionally mapped to player 2 for alternative performance scenarios. For performative accessibility, the parameters are organised into three categories, each assigned to a distinct section of the MIDI controller: Knobs 1-4 are used to control the dimensions in the Influence Sources routing module; Knobs 9-12 are used to control the Layer Balance module; Channel faders 1-4 are used to adjust the main output parameters. The specific mapping of all 12 parameters are displayed in Table 1.

Control	Somax2 Parameter	Range
`Knob 1`	Pitch	$\{0, 1\}$
`Knob 2`	Onset	$\{0, 1\}$
`Knob 3`	Chroma	$[0, 1]$
`Knob 4`	MFCC	$[0, 1]$
`Knob 9`	Internal melodic match	$[1, 128]$
`Knob 10`	Internal harmonic match	$[1, 128]$
`Knob 11`	External melodic match	$[1, 128]$
`Knob 12`	External harmonic match	$[1, 128]$
`Fader 1`	Timeout	$\{0, 1\}$
`Fader 2`	Continuity	$[0, 127]$
`Fader 3`	Quality	$[0, 1]$
`Fader 4`	Probability	$[0, 1]$

Table 1. MIDI controller mapping for Somax2 player, configured for Minotaur's Lament.

The dimensions of the influence sources correspond to pitch, onset, chroma, and the Mel-frequency Cepstral Coefficient (MFCC), which are computed for the vocal input. Thus, knobs 1-4 allow the performer to select and mix between the dimensions, adjusting the influence level of each audio feature as received by the Somax2 player.

The Layer Balance module controls the relative weighting of various layers, including melodic and harmonic similarities, across both internal and external influences, thereby shaping the player's corpus selection process. Increasing the weight of the internal influences leads the player to prioritise musical similarities that correspond to its previous output, while increasing the weight of the external influences increases responsiveness to the performer input. By dynamically scaling these weights using knobs 9-12, the performer can shape the interactiveness of the player, and control the degree to which the vocal pitch shift influences Somax2's output generation.

Additionally, the main output parameters, mapped to channel faders 1-4, are central to the player's generation process. *Timeout* dictates the duration of a corpus that the player outputs when no new performer triggers are detected. *Continuity* controls the adjacency of the next selected corpus event relative to the previous output. *Quality* sets a similarity threshold that determines whether a corpus match is considered valid for output. *Probability* conditions each generation with a stochastic triggering likelihood, regulating the density of the output. Assigned to a MIDI interface, these parameters enable the user to continuously fine-tune Somax2 behaviour, thereby supporting improvisations with shifting intensity and evolving momentum, central to the performance of Minotaur's Lament.

3.3 DAW Integration

As the final layer of the architecture, we integrate a digital audio workstation (DAW) as the output engine of the performance. The vocal signal routed from Max/MSP is processed in Logic Pro with a flanger effect. This enhances the voice's stereo depth, and introduces phase interference, creating spatial ambiguity and instability that reinforces the labyrinthine theme of Minotaur's Lament.

Furthermore, the sonic realisation of the Somax2 output is fully implemented within Logic Pro. As MIDI corpora are used for both players (see Section 5.2), we transmit the live output of each player into Logic Pro via a virtual MIDI input. Each player thus triggers a virtual synthesiser within the DAW, thereby constructing the multi-layered soundscape. A third synthesiser track is controlled by the aforementioned MIDI keyboard, which functions both as the influence of player 2, and as an additional instrument within the sonic texture.

4. CULTURAL BACKGROUND

The performance of Minotaur's Lament is deeply grounded in culture. Therefore, the artistic style and aesthetic framework of the improvisation are informed by both Greek mythological narratives and cultural traditions, as discussed in this section.

4.1 Minotaur and the Labyrinth

In Greek mythology, Minotaur is portrayed as a monstrous figure, born as a result of divine punishment, when King Minos of Crete failed to sacrifice the bull sent to him by Poseidon [23]. An unnatural creature with a human body and head of a bull, Minotaur was of violent nature, and fed on human flesh. Thus, he was imprisoned at the centre of the labyrinth, a maze constructed to conceal him from the outer world, with impossibility of escape [24]. King Minos ordered that seven boys and seven girls are chosen each year to enter the labyrinth as tributes to be devoured by the Minotaur. Thus, the archetypal image of the monster establishes the underlying theme of our improvisation, with suppressed guilt, chaos, isolation and repression positioned at the centre of its musical narrative. Moreover, Minotaur's voice is described as a "low methodical roar" that instils fear [23]. Accordingly, the performer predominantly sings in the lower register, reinforcing the dark tone of the improvisation.

Beyond the figure of Minotaur, the labyrinth has become a recurring metaphor in modern culture, emerging as symbolic tropes in arts and literature. The labyrinthine form has been described as a symbolism of "unconsciousness" [21], and as a structure of "non-linearity" [25], which is reflected in Minotaur's Lament through the constantly evolving vocals, and the randomness of the Somax2 player output. Furthermore, [26] characterises the experience of navigating the labyrinth as "fragmented". Artistically, we mirror this idea through frequent pauses in the vocals, articulating the music with a sense of detachment. Consequently, this alters Somax2's onset detection through its low-latency spectral flux algorithm [13, 18], hence intensifying the musical effect of the pauses. The improvisation is performed without a definite structure or resolution, emphasising the labyrinth's unpredictable atmosphere.

4.2 Lament in Ancient Greece

In Ancient Greece, the lament was a musical genre which originated as a ritual mourning song performed in funerary contexts [11]. Such lamentation rituals have been recorded by Homer in "The Iliad", when the Trojans mourn the loss of Hector. The poem depicts two forms of mourning songs: *gôos*, which is a highly improvised and personal lament sung by family members of the deceased; *thrênos*, which is a more formal and structured dirge [27]. Lamentations have also been featured in ancient Greek theatre, such as *kommós* in Athenian tragedies, emphasising the dramaturgical, dialogic representation of the mourning customs.

The Greek mourning ritual often involves heightened emotional expression, and is commonly accompanied by gestures and movements, including raising hands to the head [29]. Moreover, *gôos*, which most closely resembles the style of Minotaur's Lament, was traditionally sung solo with exaggerated gestures. Thus, our use of hand gestures in a solo performance context embodies the Ancient Greek rituals, exploring the intersection between cultural traditions and human-computer interactive performance.

Ancient Greek laments were often of high vocal intensity and emotional display, with screaming and wailing among

Figure 5. Ancient Greek mourning ritual, as depicted by a Terracotta funerary plaque, ca. 520-510 BCE. [28]

the most prominent features of lamentation [11]. Our use of the pitch shift effect, controlled by the gestures, thereby serves the purpose of amplifying the performer's vocal expressivity, recontextualising the lament while preserving its traditional characteristics. Particularly, an upward shift enhances the effect of human wailing, while a downward shift draws the performer closer to Minotaur's low voice and his animalistic nature. The gain control functionality further enables varying vocal intensity during the performance.

Generalising Minotaur as a non-verbal creature, the vocals in Minotaur's Lament also refrain from using spoken words, taking the form of an extended vocalisation. This allows the musical style of the Greek lament to be adapted to channel the mythological narrative of the monster. In addition, mourning rituals also served to alleviate the pain of the living [11], making the lament a fitting musical style for Minotaur's emotional expression – the children sacrificed as the "deceased", and his own imprisonment in the labyrinth as the source of inner suffering. This serves as the central artistic value of our work.

4.3 Lament in Modern Greece

Our performance also draws on musical elements of modern Greek lament, which are direct continuations of Ancient Greek lament forms. Specifically, *moirologia* is a lament genre popular in 19th and 20th century Greece. The vocals are often characterised by melismatic embellishments [30], as well as the use of microtones as opposed to a diatonic scale [31]. Integrating these elements, the vocals in Minotaur's Lament also place an emphasis on the usage of melisma. Meanwhile, the soundscape created by Somax2 is designed to be fully atonal, reflecting the unconventionality of the lament.

5. OBSERVATIONS & DISCUSSION

5.1 Gestures

Our interactive system for gesture-controlled improvisation with Somax2 has been tested by users with varying degrees of musical and Somax2 experience. Feedback indicates that most users found the usage of gesture control to have created a highly engaging musical experience, with several participants noting the enhanced interactiveness of the performance compared to standard Somax2 improvisations. In addition, users also reported that the pitch shift effect expanded the expressivity of their voice, allowing them to produce vocalisations outside of their natural vocal range. When used with the setting and vocal style of Minotaur's Lament, gesture control embodies not only the movements and sounds of the Ancient Greek mourning rituals, but also supports the creation of an immersive soundscape that pictures the imagery of Greek mythology.

5.2 Somax2 Output

Our system uses a MIDI corpus for both Somax2 players. Central to the labyrinth trope, not only does the versatility of such corpora facilitate higher performance variability, this MIDI-based setup also allows the performer to redesign the Somax2 output rather than relying on preexisting sonic material, hence enabling the real-time creation and synthesis of unique soundscapes, and maximising the creative potential of Somax2. Our design thus promotes human-machine co-creation and interaction, allowing users to shape the AI-generated soundscape through gestures and voice. In order to capture the labyrinthine atmosphere, the atonal corpora are used in combination with synthesiser sounds that produce a hollow timbre.

Influence	Voice	$\bar{x}$	s
Default	Clean	6.8	0.837
	Pitch shift	15.0	1.871
Chroma off	Clean	7.0	0.707
	Pitch shift	20.0	1.225
Pitch off	Clean	4.6	1.140
	Pitch shift	18.4	2.191

Table 2. Comparison of Somax2 output behaviour with and without pitch shift under reactive mode, across 5 trials.

To test the impact that pitch shift exerts on the Somax2 player output, and thus gain a better understanding of gesture control as an interactive medium, we recorded two 10-second audio files, in which the same note is vocalised. One recording contains the clean voice, while the other is processed with pitch shift. We then measured, under varying influence source settings, the number of events triggered by the Somax2 player (loaded with the same corpus) when each recording is used as the audio input. Each case was repeated across five trials, and the mean number of triggered events, $\bar{x}$, is recorded in Table 2. As shown, the use of pitch shift consistently increases the number of events generated by the system, regardless of the influence dimensions. Furthermore, the higher standard deviation produced by the pitch shifted audio suggests that such vocal modulation increases the randomness of Somax2 output. This confirms that our addition of gesture control places a notable influence on Somax2 behaviour via pitch shift, expanding the degree of user interaction with AI.

5.3 Challenges

A number of technical and practical challenges occur during the performance, and could be addressed in the future to further improve the system's user experience:

- Latency: the complexity of our architecture results in a perceptible latency during the improvisation, both between the gestures and effects, as well as vocal I/O. On modern devices, reducing the buffer size can partially resolve this issue.
- User limitations: in solo performances, the degree of interactive control is slightly reduced, as the performer is unable to make Somax2 parameter adjustments while simultaneously controlling the music with gestures and keyboard input. This can be addressed through the use of a MIDI foot controller instead of a tabletop interface.

6. CONCLUSIONS

Real-time gesture control driven by computer vision has presented diverse applications in human-computer interactive performances. Contributing to the field, we presented a novel computer system that supports the use of gestures in augmenting musical improvisation with Somax2, a multi-agent AI co-creative system in Max/MSP. Using gestures to modulate the performer's live vocal input with a pitch shift effect, our architecture not only supports amplified musical expression, but also enhances sonic interaction between the user and the AI engine, as we found the Somax2 player to exhibit more diverse output selection under gesture influence. Our study further pushes the boundary of culturally-grounded computer music by situating the system in the context of Greek traditions, thereby creating an embodied improvisation which renews ancient musical forms, traditional rituals, and mythological narratives within a modern computational framework. This approach exemplifies that gesture-controlled improvisation with AI can be effectively linked to cultural heritage, and opens new gateways for performers and audiences to connect with traditional practices and customs.

Future work in this field could involve the extension of gesture control interfaces towards full-body motion tracking, enabling a broader range of performer movements to interact with AI creativity. This may also be explored in the direction of multi-performer setups. Additionally, gestural design could be adapted to serve more culturally-embedded musical forms and movements, allowing human-AI musical co-creativity to engage with a wider range of performance traditions.

Acknowledgments

The author would like to thank Tilemachos Moussas for his novel research in the use of kinesthesis-based gesture control for Somax2 in the context of Greek musical theatre.

7. REFERENCES

[1] P. Modler, T. Myatt, and M. Saup,"An Experimental Set of Hand Gestures for Expressive Control of Musical Parameters in Realtime," in *Proceedings of the Conference on New Interfaces for Musical Expression (NIME)*, 2003, Monetral, Canada.

[2] F. Zhang, V. Bazarevsky, A. Vakunov, A. Tkachenka, G. Sung, C.-L. Chang, and M. Grundmann, "MediaPipe Hands: On-device Real-time Hand Tracking," *arXiv preprint arXiv:2006.10214*, 2019.

[3] M. Wright, A. Freed, A. Lee, T. Madden, and A. Momeni, "Managing Complexity with Explicit Mapping of Gestures to Sound Control with OSC," in *Proceedings of the International Computer Music Conference (ICMC)*, 2001, pp. 314–317.

[4] R. Sun, A. V. Wallop, G. Leslie, and E. Y.-L. Do, "SoniSpace: Expressive Movement Interaction to Encourage Taking Up Space with the Body," in *Companion Publication of the ACM Designing Interactive Systems Conference (DIS Companion)*, 2023, pp. 279–283.

[5] Y. Kim, S. Park, M. Parker et al., "NeoLightning: A Modern Reimagination of Gesture-Based Sound Design," *arXiv:2505.10686v1*, 2025.

[6] Y. Yang, Z. Wang, and Z. Li, "MuGeVI: A Multi-Functional Gesture-Controlled Virtual Instrument," in *Proceedings of the International Conference on New Interfaces for Musical Expression (NIME)*, 2023.

[7] M. Amerotti, B. L. T. Sturm, S. Benford, H. Maruri-Aguilar, and C. Vear, "Evaluation of an Interactive Music Performance System in the Context of Irish Traditional Dance Music," in *Proceedings of the International Conference on New Interfaces for Musical Expression (NIME)*, 2024.

[8] W. Wu and H. Qu, "Gesture-Driven DDSP Synthesis for Digitizing the Chinese Erhu," in *Proceedings of the International Conference on New Interfaces for Musical Expression (NIME)*, 2025.

[9] K. R. Scherer, "Vocal communication of emotion: A review of research paradigms," *Speech Communication*, vol. 40, no. 1–2, pp. 227–256, 2003.

[10] A. Vysocký, T. Poštulka, J. Chlebek, T. Kot, J. Maslowski, and S. Grushko, "Hand Gesture Interface for Robot Path Definition in Collaborative Applications: Implementation and Comparative Study," *Sensors*, vol. 23, no. 9, art. 4219, 2023.

[11] A. Ntousiopoulou, "The Exaggeration in Greek Lament: The Diaspora of the Phenomenon Through Time on a Diachronic Level, from Antiquity to the Present Day, in the Context of Greek Civilization," *Series Musicologica Balcanica*, no. 6, 2025.

[12] K. Karathanasis, *Medusa In Somno v2.9*, electroacoustic composition, commissioned by Madeleine Shapiro, 2019.

[13] T. Moussas and A. Georgaki, "Enhancing Vocal Improvisation Through Interactive Media in Drama Performance — Insights from Kinesthesis Tools 2.0 Synergized with Somax2," in *Proceedings of Sound and Music Computing (SMC)*, 2025.

[14] J. Borg, *Somax 2: A Real-Time Framework for Human-Machine Improvisation*, technical report, IRCAM, 2019.

[15] J. Borg, M. Malt, G. Assayag, B. Sluchin, and S. Conforti, "Somax 2, a Reactive Multi-Agent Environment for Co-Improvisation," in *Proceedings of Sound and Music Computing (SMC)*, 2022.

[16] J. Borg, *The Somax 2 Software Architecture*, technical report, IRCAM, 2021.

[17] T. Moussas, N. Kotsani, and A. Georgaki, "Kinesthesis: An Interactive Voice Software for Intertextual Improvisation in Theatre Performance," in *Proceedings of Sound and Music Computing (SMC)*, 2022.

[18] J. Borg, *Onset Detection in Somax*, technical report, IRCAM, 2022.

[19] M. S. Olivier, *An Interpretation of Theseus and the Minotaur*, essay, 2002.

[20] D. I. Kalashnikova, "Metaphor of the Labyrinth in the Musical Culture of the Second Half of the XX Century: Ballet "Labyrinths" by A. Schnittke," *Philosophy and Culture*, no. 5, pp. 38–45, 2022.

[21] R. McDonough, "Wittgenstein's and Borges' Labyrinth-Imagery," *Athens Journal of Humanities & Arts*, vol. 5, no. 4, pp. 425–446, 2018.

[22] M. Larrieu, "An Analysis of "nyx" (2017), a Computer Music Work by Kerry Hagan," *Computer Music Journal*, vol. 44, no. 2–3, pp. 118–132, 2020.

[23] N. Tessmer, "Myth, Ritual, and the Labyrinth of King Minos," *Armstrong Undergraduate Journal of History*, vol. 5: Iss. 1, Article 1, 2015.

[24] M. M. Winkler, "The Labyrinth: Narrative Complexity, Deadly Mazes, and Ovid's Modernity." in *Ovid on Screen: A Montage of Attractions.* Cambridge University Press, pp. 119–158, 2020.

[25] T. Šosterič, *Labyrinths in Postmodernism: Danielewski, Pynchon, and Wallace*, Master's thesis, Univerzita Karlova, Filozofická fakulta, Prague, Czech Republic, 2019.

[26] P. R. Doob, *The Idea of the Labyrinth from Classical Antiquity through the Middle Ages*, Ithaca, NY: Cornell University Press, 2019.

[27] M. C. Pantelia, "Helen and the Last Song for Hector," *Transactions of the American Philological Association*, vol. 132, no. 1, pp. 21–27, Dec. 2002.

[28] Terracotta funerary plaque, Greek, Attic, ca. 520–510 BCE, terracotta; black-figure, The Metropolitan Museum of Art, New York, Rogers Fund, 1954, Object no. 54.11.5 (public domain).

[29] S. Houby-Nielsen, "Women and the Formation of the Athenian City-State (The Evidence of Burial Customs)," *Mètis. Anthropologie des mondes grecs anciens*, vol. 11, 1996, pp. 233–260.

[30] M. Athanasiou, "The Modern Greek Lament in the Twentieth Century: A Summary of Studies," *Humanities Bulletin*, vol. 4, no. 2, pp. 9–23, 2021.

[31] M. Andronikou and F. Sallis, "Centring The Periphery: Local Identity in the Music of Theodore Antoniou and other Twentieth-Century Greek Composers," *Intersections*, vol. 33, no. 1, pp. 11–34, Fall 2012.