Visual Representations to Stimulate New Musicking Strategies in Live Coding

R A U L   M A S U[1] a n d   F R A N C E S C O   A R D A N   D A L   R Ì[2]

[1]CMA, Hong Kong University of Science and Technology (Guangzhou), China
Email: raul@raulmasu.org
[2]Department of Information Engineering and Computer Science – DISI, University of Trento, Italy; Conservatory F. A. Bonporti, Trento, Italy
Email: francesco.dalri-2@unitn.it

In live coding, the code can be considered as an archetypal form of score that notates formal processes. We aimed at investigating the possibility of using graphic visuals as a complementary form of descriptive score by visualising sound events using different time representations. To this end, we devised two visualisation systems (Time_X and Time_Z). Time_X represents time along the *x*-axis, while in Time_Z the objects overlap along an imaginary *z*-axis. Based on our previous personal experience with the system, such forms of visual scores can help to develop new musicking strategies while live coding. In this article, we wanted to broaden such reflections, and we used them as probes in a study with three live coders. After tailoring the two systems to the usual practice of the three participants, we asked them to use the systems for three weeks and keep a diary. At the end, we interviewed them. Based on their comments, we present some reflections on the use of graphic forms of visualisation in live coding, on how they can support musicking process, and to what extent such visuals can be considered scores.

1. INTRODUCTION

In the last few decades, live coding has thrived as a novel performative practice where performers write musical instructions in the form of code interpreted in real time. Such a practice is generally based on improvisational processes, in which the musical discourse is built from scratch and developed during the performance. As discussed by Magnusson, the code can be considered as an archetype of notation of formal processes, and therefore, as a score (Magnusson 2011). Graphic forms of live visuals acting as scores to complement the code have been proposed, examples include Magnusson's Threnoscope (Magnusson 2014b) and Abreu's Didactic Pattern Visualizer. In both systems, the graphic components are designed with the aim of supporting the understanding of the code counterpart (further details will be provided in section 2.2).

In this article, we further investigate the possibility of using graphic visuals as a form of score complementing the code. In particular, we investigate if and how such forms of notation can stimulate musical ideas and support awareness about the musicking of a live coder. To this end, we used two visualisation systems as probes in a study with three live coders. The two visualisation systems (Time_X and Time_Z), showing sonic events in the form of graphic objects, manages time representation in two different and complementary manners: in Time_X, time is represented along the *x*-axis, similarly to a traditional score, while in Time_Z, the objects overlap along an imaginary *z*-axis, gradually fading away. The two systems were initially developed based on the specific needs of one of the authors of this work, who was able to develop new strategies for the musical structural development of his live coding practice thanks to the visual feedback provided by the graphic scores. In the study presented here, we adapted the visualisation systems to the live coding environments of three live coders (MrReason, Etol, u-mano u-dito), using an ideographic design approach – a design method which suggests designing technology targeting the specificities of one single user (Hook, McCarthy, Wright and Olivier 2013).

After tailoring the two visualisation systems to the live coding environments of each of the three participants, we asked them to integrate the systems into their usual rehearsing practice over three weeks and to keep a diary about their experience. Finally, we interviewed them individually. Based on their answers, comments and suggestions, in the final part of this article we propose some reflection on the use of graphic forms of visualisation in live coding and how it can stimulate creativity. In particular, we discuss the generation of new musical ideas using different theoretical lenses: extended mind, affordances and agency.

2. LIVE CODING AS IMPROVISATION BASED ON SCORES

2.1. Live coding as improvisation

Live coding performances tend to be characterised by an improvisational approach, in which performers embrace the 'from-scratch' challenge, structuring and developing their pieces entirely in real-time. To advance a definition of this approach, Magnusson

has used the lemma 'strong coding', as opposed to 'weak coding' (where the code is written in advance and simply executed or slightly modified during the performance) (Magnusson 2014a). In a strong coding performance, coders start with a blank page; additionally, the code is usually not saved at the end of the performance (Magnusson 2015). As such, live coding performances are focused on momentum and tend to have a scarce interest in reproducing previous performances. Indeed, Magnusson suggested that the live coding practice shares similarities with oral tradition as it tends to be primarily extemporaneous (Magnusson 2016). Parkinson and Bell (2015) discussed the lack of pre-packaged structures in live coding improvisation and proposed an analogy with *free improvisation* as discussed by Derek Bailey (1975) in the context of his musical practice. With this term, Bailey indicated the free exploration of the gamut of instrumental sonorities during an open improvisation. Such a form of exploration also tends to occur in live coding, when the performer explores the possibilities of a specific live coding system.

Overall, the practice of live coding is characterised by specific actions that define a high level of difficulty in this type of performance. First, musical ideas cannot be expressed immediately, as the performer needs to formalise them in the abstract form of code: *idea-to-code latency* (McLean and Wiggins 2009). Indeed, live coders make improvisational choices based on a need for the 'now', but also need to elaborate on them in the (near) future (ibid.). Second, live coding improvisations tend to be error-prone due to typos while writing code, and such risk increases with the complexity of the code (Blackwell and Collins 2005). Finally, two main interaction feedback loops (*manipulation feedback* and *performance feedback*) coexist during a live coding set, and therefore there is a probability of overloading the cognitive processes of the performer (Nash and Blackwell 2011). Some live coders have stated that they are *rarely able to develop new ideas while performing*, and they tend to recycle similar patterns and structures well assimilated into memory (McLean and Wiggins 2010). As a consequence, the structural organisation of the sonic material and the musical form has often been overlooked in live coding-related literature. We argue that this element is under-scrutinised due to live coding intrinsic lean towards improvisation.

Moreover, Sarath proposed that the main difference between composition and improvisation is the possibility to move back and forth in time (Sarath 1996). Composing with the support of traditional scores allows a musician to modify and refine musical material regardless of the order in which it appears during a piece. On the contrary, if we improvise, we cannot refine the introduction of a piece based on an

idea that we developed in the coda. With this perspective, we can observe how the two approaches imply different cognitive and musicking strategies in the relationship between musicians and music. The traditional Western dichotomy of composing–performing indeed tends to fail to represent contemporary electronic and digital music performances (see, for instance, Lansky 1990). However, we argue that this difference in the way composing and improvising deal with time is relevant in relation to musicking strategies.

2.2. Visual support in live coding practice

The relationship between computer music and live visuals has a long tradition. Far from aiming to provide our reader with an exhaustive account of such a relationship, we present here some main references that acted as a background against which we framed our two systems.

The use of visuals in digital music has been primarily used to compensate for the lack of feedback for the audience when a laptop is involved in the performance (Correia, Castro and Tanaka 2017). Starting from this original purpose, different visual solutions have been proposed to also support the understanding from the perspective of the performer (Joaquim-Fernandes and Barbosa, 2013) or of the musical piece (Hunt, Mitchell and Nash 2017). Live coding lies in an advantageous position compared with other laptop music practices, as the code is always exposed to the audience. Overall, in our approach we do not manage visuals from this traditional audience-centric perspective; rather, our visuals are a form of descriptive scores that provide the performer itself with different visual feedback. We do not exclude the possibility of using such systems as live visuals for the audience, simply, in this article, we just focus on the relationship between the performers and the systems.

According to Magnusson, the code can be considered as an archetype of visualisation of formal processes (2011), therefore as a form of score in itself (2014b). In the digital music domain, the term 'score' has been used for a variety of different purposes. An overview of the scores in digital musical instruments has been recently proposed in a systematic scrutiny of the NIME performance (Masu, Correia and Romao 2021), and five main uses of scores were identified: 1) scores as instructions or information (suggesting how to play an instrument; as in Hamano, Rutkowski, Terasawa, Okanoya and Furukawa 2013); 2) scores as an interface to play a DMI (score as a controller that can be tangible; e.g., Tomás and Kaltenbrunner 2014), virtual (e.g., Masu et al. 2020), or in form of code with a graphic visualisation (e.g., Magnusson 2014b);

3) score as synchronisation (the system uses a score to synchronise various events; e.g., Orio, Lemouton and Schwarz 2003); 4) score creation (tools that support the creation of instrumental scores; e.g., Garcia, Tsandilas, Agon and Mackay 2011); and 5) score recording (score as a recording of performative actions; e.g., Liang, Fazekas, McPherson and Sandler 2017).

In this taxonomy, live coding would fit in the category of 'scores as an interface', where the code (a form of notation) is the input of the digital system that is created or manipulated during a performance. However, as highlighted by Collins, since the code is generally overwritten or erased during the performance, it may not represent the piece as a whole, but only a cross-section of it at a given moment (Collins 2003). To cope with this issue and support the understanding of specific musical elements, various visualisation systems have been implemented. While some of them actually constitute new programming environments, mostly replacing the coding part (e.g., McLean, Griffiths, Collins and Wiggins 2010; McLean and Wiggins 2011), others rely on existing systems, complementing and supporting the coding part. A relevant example is Magnusson's Threnoscope, which consists of a series of concentric circles each representing a drone and its parameters (Magnusson 2014b). Abreu proposed another system, the Didactic Pattern Visualizer, which allows the visualisation of sound events sequenced from the TidalCycles library, arranged on a temporal grid (Abreu n.d.). Although these two systems propose different temporal representations, in both cases the entire form/score of the piece is not visible. These systems are hierarchically subordinate to the sound component, and their purpose is primarily to visualise some aspects of the musical creation that are strongly correlated to specific musical needs. To underline the support function of these two systems, both authors use terms such as 'helpful' and 'didactic'. Purcell reflected on these differences and proposes two macro-categories of visual techniques in live coding: *aesthetic* and *didactic* (Purcell, Gardner and Swift 2014).

The two visualisation systems that we proposed in this article as probes belong to the category of 'scores as an interface'. However, they play a complementary function to the code, indeed the score acts as feedback providing a general view of the musical events. Additionally, while the code usually displays what is playing now and what will play in the near future, our visualisation systems focus on already played events. As such we try to 1) bring part of the improviser's focus into the past temporal dimension, and 2) support the musicking process of developing a musical piece.

3. TWO VISUALISATION APPROACHES

As mentioned in the introduction, to further investigate the use of visuals as support to the creative process of musicking for live coders we devised two systems (Time_X and Time_Z) that we used as probes in this study. The two systems were initially developed based on the need of one of the authors of this manuscript. The core idea was to devise visual counterparts for each sound event, so as to promote a different perception of the coding output. Our expectation was that a new layer in the cognitive process of the musician–code–music relation could beget novel musicking strategies. The description of this initial implementation and of its autobiographical evaluation are presented in a dedicated paper (Dal Rì and Masu 2022). We provide here an overview of the two systems.

Time_X (Figure 1) is based on the standard Western representation of time, as in classical score, musical events (notes) are arranged from left to right and top to bottom. The canvas is divided into horizontal areas and the events are temporally displaced following a moving pointer. The pointer's increment in time depends on the duration of the whole performance and by the number of staves, both determined by the user at the beginning of the performance. With this approach, the structure of the entire set will be visible in the resulting 'score'. As a drawback, squeezing the full piece in the space of the screen tends to result in blurred details on single graphical objects.

In Time_Z (Figure 2), the time is represented along an imaginary *z*-axis, and the graphic objects are arranged in different areas of the screen, overlapping each other. The idea of the passage of time is conveyed by superimposing a dark layer in transparency at a slow framerate; in this way, less recent shapes gradually become darker and darker until they eventually disappear in the background. By default, this layer overlay allows objects to remain visible for approximately 50 seconds. Consequently, with this system, it is not possible to visualise the entire performance. However, it ensures greater details on each graphical object/sound.

Both prototypes are based on the TidalCycles[1] live coding framework, and implemented using the atom.p5.js library,[2] which allows running p5.js[3] sketches directly in Atom.[4] Therefore, the window of the text editor became an actual canvas where the graphic objects and the code are displayed as overlapping layers (code in the foreground, visuals in the background). Further details on the original design,

[1]https://tidalcycles.org/.
[2]https://atom.io/packages/atom-p5js.
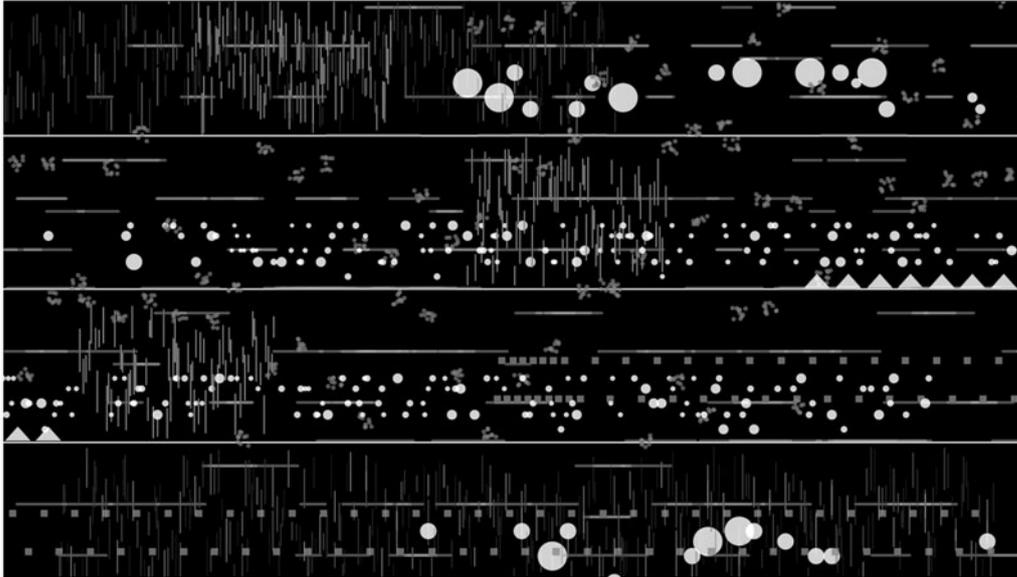[3]https://p5js.org/.
[4]https://atom.io/.

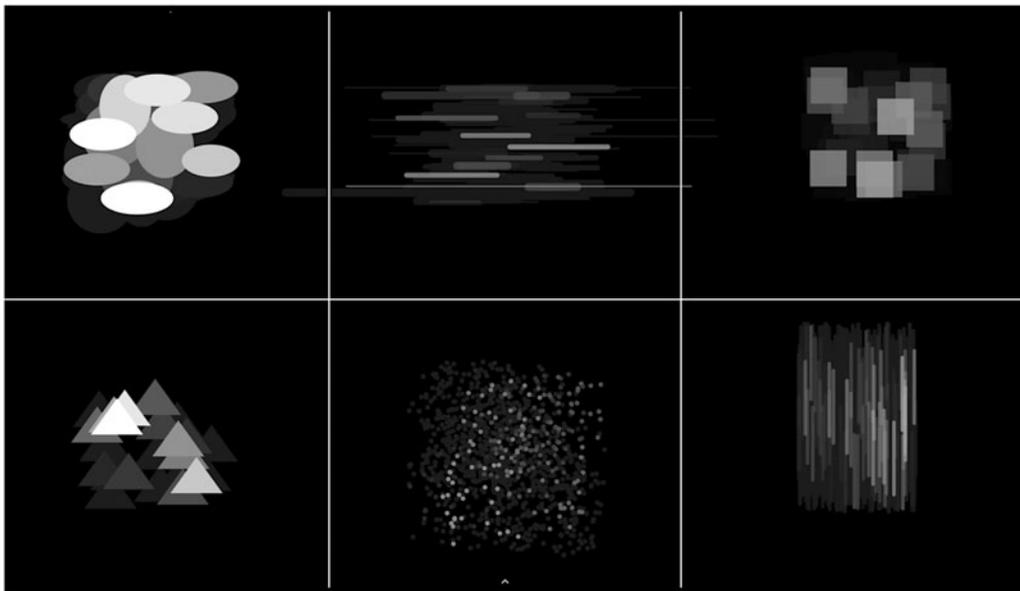**Figure 1.** An example of visualisation with Time_X.

Figure 2. An example of visualisation with Time_Z.

the intrinsic choices and implementation can be found in Dal Rì and Masu (2022).

4. THREE IDEOGRAPHIC PROCESSES

We used the two systems as probes to investigate how different visualisation can stimulate different cognitive strategies in relation to managing the musicking processes in live coding. As already discussed, digital musical instruments can serve as cultural probes that are developed to explore how music can be thought of

and created (Tahıroğlu, Magnusson, Parkinson, Garrelfs and Tanaka 2020). Cultural probes were introduced by Gaver and colleagues in the field of interaction design as a set of tools designed to provoke inspirational responses (Gaver, Dunne and Pacenti 1999). In the context of this article, we look at our systems as probes that we hope can provoke responses in the way live coders explore musical creation. These visualisation systems are probes as they (hopefully) stimulate imaginations rather than define problems or strategies. To this end, we invited three experienced

live coders (MrReason, Etol, u-mano u-dito) to integrate our visualisation systems in their usual live coding environments over three weeks. Each of the three live coders has many years of experience as a performer, their ages range between 30 and 40 years and they are European and male. We acknowledge this as a limitation of this work: as the live coding community is spread worldwide, it is probable that, with a more diverse population, a more diverse variety of comments would emerge in terms of musical ideas related to their background. However, the musical approaches of the three performers involved in the study (described in a few paragraphs later) are quite different, thus covering a diverse gamut of musical ideas.

To facilitate the three live coders to use our systems in their usual environment, we underwent three short ideographic design processes, one per participant. Such a process is a form of research through design in which one artefact is tailored to one person's specific needs, and has been successfully used to investigate the design of creative technology (e.g., interfaces for audiovisual performance; Hook et al. 2013) However, to the best of our knowledge, this approach is still underused in live coding. Overall, our work is structured in three main phases (with each of the performers). After adapting the systems to each live coder, we asked them to use the systems for three weeks in their daily practice, and finally we interviewed them.

During the initial adaptation phase, we first performed an individual interview focused on how our participants normally use their systems and approach the structuring of the musical material to create a piece from scratch. Second, we asked them to cluster the sound objects they use (e.g., synths, samples) according to arbitrary musical functions, and assigned colours and shapes to each group. In order to find a clustering that suits their musical needs during a performance, we showed recordings of the systems, and briefly discussed the different categories and parameters. This activity produced a table for each participant in which the various sounds correspond to different colours and shapes. Based on their inputs and preferences, we modified the visualisation system for each participant.[5]

For each musician, we provide a brief description of his systems and approaches (derived from the initial interview). A summary table of the chosen sound clusterisation along with their respective mappings can be found in Tables 1, 2 and 3, respectively.

MrReason uses a variety of sounds to which he is already accustomed to attribute high-level functional

musical concepts, such as 'harmonic progressions' and 'melodic lines' (Figure 3). He comfortably uses traditional Western music terminology and has an instrumental background as an electric guitarist. During his from-scratch performances, he always tries to 'insert something new' and focuses mainly on rhythmic development and exploration/construction of harmonic–melodic relationships starting from simple musical cells.

Etol stated that he approaches the performance in an unstructured way, in which 'a series of sounds follow each other over time' (Figure 4). Generally, however, he tends to prepare the beginning and end of the set in advance. Influenced by his background as a percussionist, he thinks of his sound objects as 'a huge drumset', and the focus of his performances is consequently oriented towards the development of rhythmic textures. His way of thinking about sound categories is linked to the timbre of the sounds used.

u-mano u-dito bases his performances on one single sample, which he processes through a series of functions obtaining different sonorities (Figure 5). Therefore, the same starting sample gradually takes on different characteristics and different musical functions within the improvisation. Having very simple source material, he thinks of his sound material as divided into three essential categories: rhythmic structures, supporting sounds in the lower part of the frequency spectrum, and melodic lines in the foreground.

All the performers have spontaneously opted for a few parameters to be displayed – mainly related to intonation, intensity and duration – focusing their attention on what they use most. Furthermore, where possible, they have kept constant both the number/type of parameters and the relative mapping.

We asked each participant to use the tailored versions of the systems in their usual from-scratch practice for three weeks. They were also free to contact us for making further modifications in the system or operate modifications by themself. During this period, we asked them to keep a diary about what they performed and what changes they made to the system. As we were not prescriptive on how to develop the diary, the three live coders produced three different forms of logs: MrReason created a .doc with screenshots and comments each time he used the systems; Etol created an audio log, commenting at the end of each session (we transcribed it, for the scope of the analyst); and u-mano u-dito created a git project, adding the code he used for the rehearsals with some comments and notes about his feelings. Finally, we interviewed each of the participants using semi-structured interviews. Questions ranged from how the overall experience was, to specific questions about the issues

[5]The code can be found at: https://github.com/return-nihil/Live_Coding_Visuals.

Table 1. Summary of the clustering and mapping chosen by MrReason

Sounds	Shape	Colour	Parameters	Mapping
Pads	Rectangle	Yellow	Gain	Transparency
			Note	Saturation
			Filter	Brightness
Keys	Smaller rectangles	Orange	Gain	Transparency
			Note	Saturation
			Filter	Brightness
Leads	Horizontal line	Violet	Gain	Transparency
			Note	Saturation
			Filter	Brightness
Melodies	Horizontal line	Light violet	Gain	Transparency
			Note	Saturation
			Filter	Brightness
Basses	Circle	Light blue	Gain	Transparency
			Note	Saturation
			Filter	Brightness
FXs	Particles	White	Gain	Transparency
			Note	Saturation
			Filter	Brightness
Voices	Vertical line	Green	Gain	Transparency
			Filter	Brightness
Drums	Triangle	Blue	Gain	Transparency
			Type	Shape

Table 2. Summary of the clustering and mapping chosen by Etol

Sounds	Shape	Colour	Parameters	Mapping
Kicks	Circle	Blue	Gain	Transparency
			Speed	Y-axis
Basses	Rectangle	Green	Gain	Transparency
			Speed	Y-axis
Hats	Particles	Yellow	Gain	Transparency
			Speed	Y-axis
Highs	Triangle	Red	Gain	Transparency
			Speed	Y-axis
Long	Horizontal line	Green	Gain	Transparency
			Speed	Y-axis
Begins	Horizontal line	Purple	Gain	Transparency
			Speed	Y-axis

Table 3. Summary of the clustering and mapping chosen by u-mano u-dito

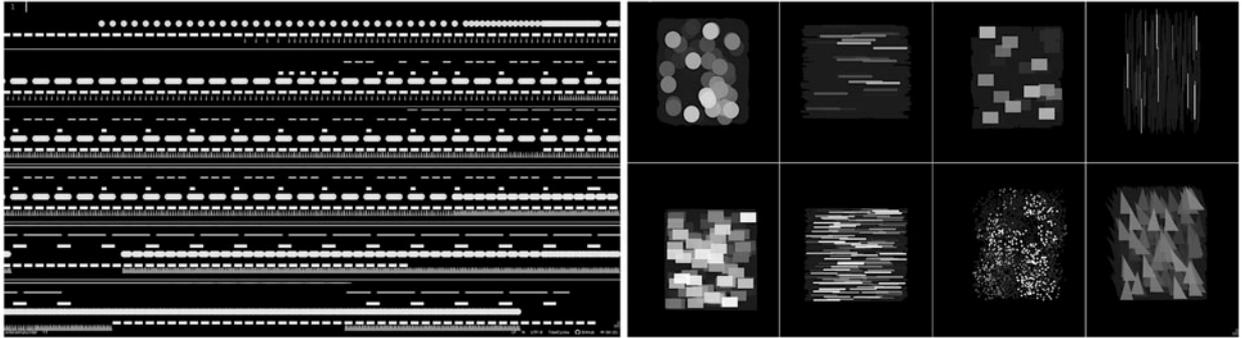| Sounds	Shape	Colour	Parameters	Mapping
Rhythms	Triangle	Blue	Gain	Transparency
			Speed	Y-axis + dimensions
			Begin/end	Shape
Basses	Triangle	Yellow	Gain	Transparency
			Speed	Y-axis + dimensions
			Begin/end	Shape
Melodies	Triangle	Red	Gain	Transparency
			Speed	Y-axis + dimensions
			Begin/end	Shape

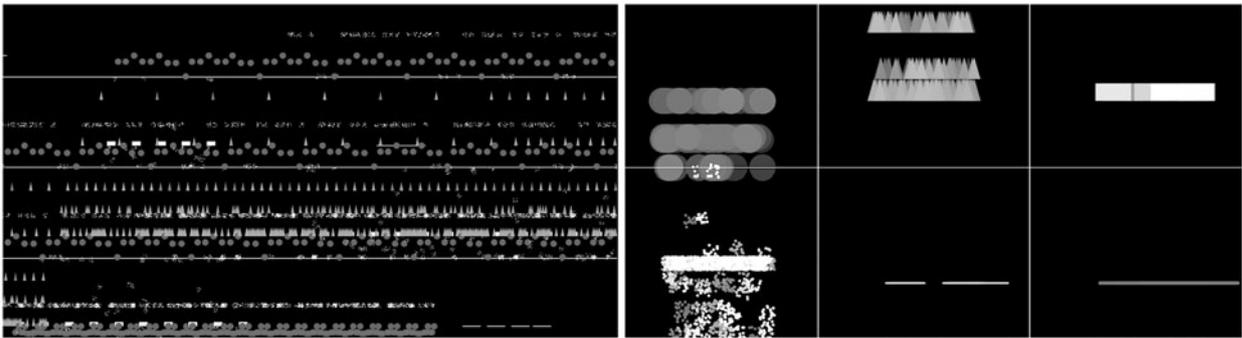**Figure 3.** Examples of Time_X (left) and Time_Z (right) adapted for MrReason.

Figure 4. Examples of Time_X (left) and Time_Z (right) adapted for Etol.
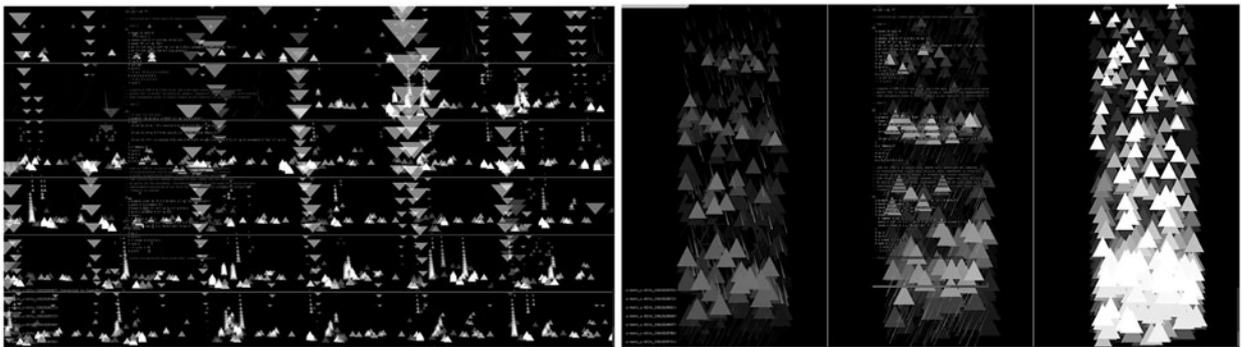

Figure 5. Examples of Time_X (left) and Time_Z (right) adapted for u-mano u-dito.

encountered or if the system suggested a new way of structuring their pieces.

5. GRAPHIC REPRESENTATION AND LIVE CODING: MUSICKING APPROACHES

Data collected from personal explorations (diaries) and in the transcript of the interviews were analysed using thematic analysis (Braun and Clarke 2006). Following this technique, we coded the text and progressively and recursively harmonised the code combining them into themes, which served as the basis for the reflection developed in the remainder of this article. We decided to include some direct quotes in our reflection – one interview was conducted in English, the other two in Italian and the quotes were translated by the authors.

5.1. New technology suggests new ideas

A musical instrument – and per extension, our visualisation tools – embeds a musical vision.

Magnusson suggested that a musical tool 'has such a high degree of symbolic pertinence that it becomes a system of knowledge and thinking in its own terms' (Magnusson 2009: 168). The author further reflects that the technologies used as part of practices of making and thinking music incorporate the musical ideas of its author(s): 'Writing digital musical interfaces therefore necessarily entails the encapsulation of a specific musical outlook' (ibid.: 173). Using terminology from Latour (1987), the musical ideas are black-boxed in the interface itself.

Musicians *offload part of their musicking thinking process* into the tools and instruments they used. Magnusson proposed the idea of an epistemic tool as 'a system of knowledge and thinking in its own terms' (Magnusson 2009: 168). The idea of epistemic tools incorporates elements from the extended mind theory by Clark and Chalmers (1998), as well as elements from enactment by Varela, Thompson and Rosch (2017). Enactment refers to the idea that cognition is the enactment of a mind and world on the basis of the variety of actions performed (ibid.). Learning to play is, therefore, an enactive activity. Clark's concept of an extended mind proposes the idea that humans tend to extend the cognitive process outside the head, offloading part of it on tools and instruments. In a live coding performance, the creative process is partially offloaded in the code. When we added the visual components, we inserted a layer that provides different feedback on what the code does, and therefore, it can change the musicking process. A new layer requires *specific focus and attention*. This redundancy of stimuli, in which each layer actually represents the same thing in different ways, was overall appreciated, but also required a different way to manage focus and attention while performing:

> Having three types of representation is a bit more difficult, since one can't pay attention to everything. For instance, at the beginning I dedicated more attention to the code and to what I was hearing. Then, once I had built some small musical cells … I saw what was happening visually, shifting the focus from the audio-coding part to the visual part. (u-mano u-dito)

Due to the need to switch attention, introducing graphic visuals could increase the *idea-to-code latency* (McLean and Wiggins 2009), since the code is not only translated into sound (code-to-sound) but also into visuals, creating a parallel code-to-visual process. Such a process could in some cases initiate a visual-to-idea-to-code loop, increasing the overall idea-to-sound latency during the performance. However, loops can also reduce this latency, as they provide feedback that helps understand the musical patterns in a specific moment:

> The possibility of seeing the musical events can help to swiftly understand what is happening, without the need to spend time reading the code. (Etol)

Additionally, *by observing the visuals some specific new ideas emerged*. As discussed by Magnusson (2009), playing music is an enactive activity. In this context, enactment refers to the idea that cognition is the enactment of a mind and world based on the variety of actions performed (Varela et al. 2017). Further developing the relationship between cognition and the external world, Rowlands developed this idea by asserting 'things going on in the environment partially constitute a cognitive process' (Rowlands 2010: 21).

As musicking is a cognitive process, it comes naturally that the visualisations – an add-on to the live coding environment – have an impact on the performative strategies and contribute to the germination of novel musical ideas. Live coding is also an enactive activity, in our case mediated by the visual. This extension of the cognitive music process in the visuals is exemplified by the fact that visuals suggested new musical patterns. For instance, Time_Z suggested to Etol to fill the graphic space until the patterns became 'something else, completely different' (Etol). This idea emerged by watching the visual that reminded him of 'some kind of cloud' (Etol). In some cases, MrReason and Etol developed strategies by purposefully changing the graphic behaviour of the visuals (not just passively observing them).

Overall, the introduction of visuals modified the musician–code–music relationship, providing new interactive patterns that opened up different possibilities. Historian of technology Melvin Kranzberg wrote: 'Technology is neither good nor bad; nor is it neutral' (Kranzberg 1986: 545). Additionally, music 'programming languages may be as culturally loaded as the communities of practice that produce and use them' (McPherson and Lepri 2020). It is therefore quite obvious that adding visualisations affects the performance. These modifications in the performance produced new ideas, as suggested by Etol:

> Both systems, in their own way, can be a breeding ground for new ideas.

In this sense, the observation by Etol echoes what was stated by Evan Parker in Phil Hopkin's film *Amplified Gesture*:

> You couple yourself to that instrument and it teaches you as much as you tell it what to do. (Hopkins 2009, cited in Melbye 2021: 20)

Furthermore, through our interactions with the three live coders, we observed how the visualisations have agency. As any artefact has agency, as pointed

out by Latour in his Actor Network Theory: 'Any thing that does modify a state of affairs by making a difference is an actor' (Latour 2005: 71). In the musical domain, in particular, technology always expresses a specific musical vision, practice or theory:

> Instruments are actors: they teach, adapt, explain, direct, suggest, entice. Instruments are impregnated with knowledge expressed as music theory. (Magnusson 2018: 79)

5.2. Different visualisations for different musical approaches

In the previous section, we have seen that instruments have agency, and as such, they introduce a new level of complexity in the musician–code–music relationship that promotes new musicking strategies. We will now discuss how the agency of the visualisation does not have an absolute inscribed and immutable value that suggests the same musical ideas to any live coder. To this end, we rely now on the concept of affordance as conceptualised by Gibson (2014). Indeed, on the one hand, the fact that artefacts have agency is a truism, on the other hand, the affordances are determined by the coupling relation between an object – in our case the visualisation – and a person – in our case the live coder (Gibson 2014).

The musical ideas that a certain type of visualisation affords are not intrinsically determined in the system but determined in the ecological relationship between the live coder and the system. Etol and u-mano u-dito reflected upon the fact that each of the two systems better fit different approaches and that different live coders would naturally prefer one or the other. The two of them expressed a preference towards Time_Z, as it gives more immediacy and is more congruent with their usual way of performing live coding. In alignment with Etol and u-mano u-dito, MrReason argued that different systems better accommodate different composition approaches. However, he preferred Time_X, and added that this system helped him to reflect upon a problem musically related to the formal development he was already considering, because 'I can directly see the development of the performance. I think that's positive' (MrReason).

MrReason also reflects on the scarce variety that his live coding performances have in terms of *structure*. We can speculate that the lack of a form of notation helping to visualise the piece in its entirety caused to a certain extent a scarce variety of different musical structures. Sarath supported that the main distinction between composing and improvising is the possibility of going back in time (Sarath 1996). Despite Time_X not allowing the performer to go back in time (as it is still generated in real time), it allows visualising the events at the beginning of the performance, as such, it allows for the possibility to keep considering events occurring in a longer period. As a result, MrReason de-intensified parts of the piece by introducing *rests and silence*, supporting variety in the overall form. However, we also observed that this feature was not particularly valued by two out of three of our participants.

Indeed, Etol perceived the lack of zoom details in Time_X as a limitation, and was forced to explore the 'speed' function more than he usually does to create a more sparse performance 'in order to create a more easily searchable graphic result'. He discovered that this strategy integrates well with the rest of his live coding ones. This observation on constrained creativity is in line with what psychologist Margaret Boden stated about human creativity. The author proposed that limitations and constraints are far from being the antithesis of creativity. On the contrary, they map out a territory of structural possibilities that can then be creativity explored (Boden 2004). Therefore, a limitation in a musical system can be a resource to develop new music strategies (Gurevich, Stapleton and Marquez-Borbon 2010). It is interesting to notice how both Etol and MrReason have created a less dense musical structure despite having reached it from different paths: purposely exploring the form (MrReason) and overcoming a limitation (Etol).

As we have seen, the systems afforded different music ideas to the various musicians we were working with. While discussing affordances in digital interactive systems, Gaver advanced the proposal of hidden affordances that emerge in the use of a specific piece of technology of which the designer was not aware (Gaver 1991). Such an interactive system has a level of ambiguity that emerged by coupling systems with people. 'Things themselves are not inherently ambiguous', rather ambiguity is determined through an 'interpretative relationship between people and artefacts' (Gaver, Beaver and Benford 2003: 235). Reflecting on ambiguity in music systems and how it begets agency, Stapleton and Davis proposed that:

> As such, ambiguous encounters impel 'users' … to assess the situation for themselves, to construct a personal understanding and connection to objects, and to question the function of these objects within their contexts of use. (Stapleton and Davis 2021: 60)

The musicians we collaborated with engaged with our visualisation system from different experiences in contexts co-determined by their usual connections, samples, functions and synths. The specificities of their new strategies are idiosyncratic and cannot be generalised as characteristic of our visualisation systems, nor should they be. That being said, it is interesting to observe how in general different cognitive processes led to ideas related to musical structures and manipulation of density.

5.3. Visualisations as scores

While developing their personal relationships with our systems, our participants related themselves to the visuals as different forms of musical scores. When communicating with our participants, to avoid biases, we purposefully avoided using the term 'score'. However, the concept of score was so clearly embedded in our study that the term emerged both in the diaries and in the interviews. We trace here how it assumed different meanings in relation to the different considerations that emerged.

Time_X has been perceived as a score in the sense of a recording or a 'trace of the sound' – using an expression proposed by Etol in his diary – of the performance. This would place Time_X as a form of 'score as a recording' according to the recently proposed taxonomy of score based on the analysis of the NIME proceedings (Masu et al. 2021). Indeed, Etol stated that, by looking at the score, he was able 'to identify macro-spots in the structure.' He also supported that these scores could be useful to revise performances afterwards, and claimed that he could probably retrace what he did.

Based on this comment, we could position Time_X as belonging to a trend that has a long history in visuals *score* for electronic music, which is **visualising existing pieces** (Adhitya and Kuuskankare 2012). Already in 1954, German composer Karlheinz Stockhausen created a visual score for his electronic piece *Studie II*. In that case, the score would grant the possibility to actually recreate the piece. Another historical example is Rainer Wehinger's visual listening score to accompany Gyorgy Ligeti's *Artikulation*. Magnusson has recently called for a musicology of code, arguing that the ability to read code should be a natural extension of the musicologist's skill set in the modern age (Magnusson 2019). We agree with this statement, however, a graphic score that visualises the overall structure of performance could complement the information provided by the code, for instance by providing an overview of the structure. This last point was also highlighted by MrReason:

> I think that it is not useful to recall specific patterns/parameters/rhythms, but the overall structures. It's more like a reflection on what kind of flow you have. You can just use the visuals in order to see if a certain idea has fit what you were trying to achieve. (MrReason)

In the Western tradition, scores progressively assumed the role of the incarnation of the ideal artwork. The idea of *Werktreue* (a German word that could be translated into 'true work') represents such an idealisation (Goehr 2007). The cases that we are discussing here, especially in light of MrReason's comment, propose a notion of score that at least partially does not align with the notion of score as *Werktreue*. Indeed, despite that the visuals created with Time_X can be used a posteriori to recall, study, and even partially recreate a performance, it does not represent the piece in its entirety, nor provide all the information necessary to recreate it (it would be necessary to know the mapping, and have the actual samples and synths controlled via a code-score).

Despite being less similar to a traditional score, Etol reported in the diary that Time_Z, which does not allow for later recovery of the piece at all, represents a better score for live coding. Elaborating more on this during the interview, however, he reflects on the fact that Time_Z is not actually a score, as 'it gives you a visual mapping to the things you are doing at the moment. I liked very much seeing the segmentation of my material' (Etol). Similarly, u-mano u-dito also preferred Time_Z, as it 'paints' a visual counterpart to what he was doing musically, 'slowly vanishing without leaving a trace'. He added that he prefers 'the classic interactive aspect of the performance, as what I did before may not interest me.' (u-mano u-dito).

Overall, we can reflect on how shaded the concept of score can be. The idea of a score as a representation of an entire piece is still quite strong – 'It shows what I am doing in real time, it's not a score' (u-mano u-dito), 'Time_Z . . . is not really a score' (Etol). However, in its initial statement, Etol argued that Time_Z is a better score for live coding. By looking at these statements, our interpretation is that, on the one hand, the term score is still impregnated with the idea of *Werktreue*, on the other hand, the new forms of digital performance are indeed pushing towards a new conception of score. Alvin Lucier, while commenting on Gordon Mumma's *Hornpipe* (1967) proposed that 'the scores were inherent to the circuits' (Lucier 1998). Therefore, we can argue that new conceptions of scores have been proposed at least for 50 years. So why are we still so bound by a traditional conception of score? On the one hand, we have to remember that our participants come from European countries, and can therefore have a natural bias towards the latter conception of score. On the other hand, if that is the case, we could assume this bias can be encountered in many other European live coders. Is that because in the 'common language' a score is a piece of paper with staves and notes whose graphemes were formalised more than 500 years ago? Replying to these questions is beyond the scope of this work. However, we wish to highlight that to work towards a visualisation of live coding as a form of musical reflection, the intrinsic bias in the term 'score' needs to be considered. Overall, a 'traditional' form of score, where the entirety of the musical symbols is arranged along the *x*-axis, might

not be preferred by some live coders, but it can give new ideas to others.

5.4. Reflections on graphic visualisations

We finally collected a number of practical suggestions and feedback on issues or improvements related to the visual components. We report them here, hoping that they can be of use to design other systems in the future. MrReason and Etol state that in some moments of their performances the visuals (Time_X in particular) distracted them, this can happen because 'they are very present in the foreground' (MrReason).

With Time_Z, we have gathered mixed opinions on whether the object clusters are too many or too few to deal with during a performance. For instance, MrReason reported that

> Eight squares are too many, especially because you will surely want to play only four/six at a time in order not to have to consider too many elements during live coding. … So I further grouped the elements which I rarely use and concentrated on the essential. (MrReason)

On the other hand, Etol 'would have added other shapes and colours, further subdividing the macro-categories we had previously clustered in order to have more references.

However, this remains a completely personal factor and depends on 'the type of use that each performer makes of his/her sound material and on what and how much he/she wishes to visualise' (u-mano u-dito). Finally, in the Time_X approach it emerged that there is a risk for 'too much rhythmic density, therefore [with Time_X] the visualisation became a kind of a big stain' (Etol).

6. CONCLUSION

In this article, we investigated the use of interactive visuals as a support to musicking in live coding performances. Our discussion is grounded on a study involving three live coders in which we devised two visualisation systems (Time_X and Time_Z) as a probe.

The external visualisation creates an additional level in the musician–code–music relationship. In the traditional one, the live coding environment in its entirety has agency. As we have seen, the two systems afforded different music ideas, which are idiosyncratic to the characteristics of the live coders. We suggest that our systems can propose a gamut of possibilities. On the one hand, seeing the past elements suggest formal change, on the other, the visuals themselves can become a focus with the aim of creating visual patterns. Based on personal background and preferences of the live coders, different coupled affordances emerged within the space defined by these many

possibilities during the poiesis of the musicking activity, which becomes a distributed activity among an ecology of actors (human and non-humans). First, we observe how important it is to find a good approach on how to manage the visualisation system in alignment with the practice of a performer. Overall, we observed how a more interactive approach (Time_Z) was preferred by two of our participants; however, a more static and traditional approach (Time_X) generated a variety of ideas in relation to the musical form and density. By seeing the musical actions in the visual forms, a few new musical ideas emerged, in particular working towards reducing the musical density and adding rest as a way of creating variety or even changing the structure of the piece. The visuals add a layer in the system, therefore they constitute an additional tool where the cognitive process of musicking is offloaded, suggesting new possibilities. However, this element also increases the complexity, requires specific focus and can introduce latency in the idea-to-code-to-sound process. Finally, we proposed some reflection on the legacy of the traditional conception of score, how the traditional score as representation of a piece is largely non-applicable, and probably not so useful, to a live coding performance.

We propose that using visuals as a form of score to complement code can be useful to stimulate new ideas during a performance. However, the most fruitful application is probably for preparation or learning, as a tool to stimulate new ideas. This possibility can be particularly relevant as some live coders have stated that they are rarely able to develop new ideas while performing (McLean and Wiggins 2010).

We hope that this article offers meaningful insight on how to design or use live visuals as a creative support for live coders to reflect on musical ideas and to develop greater awareness of their creative choices with respect to the musical performances as a whole.

REFERENCES

Abreu. n.d. Didactic Pattern Visualizer. https://github.com/ivan-abreu/didacticpatternvisualizer (accessed 4 January 2023).

Adhitya, S. and Kuuskankare, M. 2012. Composing Graphic Scores and Sonifying Visual Music with the

SUM Tool. *Proceedings of the 9th Sound and Music Computing Conference (SMC 2012)*, Copenhagen, Denmark, 171–6.

Bailey, D. 1975. *Improvisation*. Rochester, NY: Ampersand.

Blackwell, A. F. and Collins, N. 2005. *The Programming Language as a Musical Instrument*. PPIG 11.

Boden, M. A. 2004. *The Creative Mind: Myths and Mechanisms*. New York: Routledge.

Braun, V. and Clarke, V. 2006. Using Thematic Analysis in Psychology. *Qualitative Research in Psychology* **3**(2): 77–101.

Clark, A. and Chalmers, D. 1998. The Extended Mind. *analysis* **58**(1): 7–19.

Collins, N. 2003. Generative Music and Laptop Performance. *Contemporary Music Review* **22**(4): 67–79.

Correia, N. N., Castro, D. and Tanaka, A. 2017. The Role of Live Visuals in Audience Understanding of Electronic Music Performances. *Proceedings of the 12th International Audio Mostly Conference on Augmented and Participatory Sound and Music Experiences*. New York: ACM, 1–8.

Dal Rì, F. A. and Masu, R. 2022. Exploring Musical Form: Digital Scores to Support Live Coding Practice. *Proceedings of the International Conference on New Interfaces for Musical Expression 2022*, Aotearoa, New Zealand.

Garcia, J., Tsandilas, T., Agon, C. and Mackay, W. 2011. InkSplorer: Exploring Musical Ideas on Paper and Computer. *Proceedings of the International Conference on New Interfaces for Musical Expression*, Oslo, Norway.

Gaver, B., Dunne, T. and Pacenti, E. 1999. Design: Cultural Probes. *interactions* **6**(1): 21–9.

Gaver, W. W. 1991, March. Technology Affordances. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, New Orleans, LA, 79–84. https://doi.org/10.1145/108844.108856

Gaver, W. W., Beaver, J. and Benford, S. 2003. Ambiguity as a Resource for Design. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. New York: ACM, 233–40. https://doi.org/10.1145/642651.642653.

Gibson, J. J. 2014. *The Ecological Approach to Visual Perception: Classic Edition*. London: Psychology Press.

Goehr, L. 2007. *The Imaginary Museum of Musical Works: An Essay in the Philosophy of Music*, rev. edn. Oxford: Oxford University Press.

Gurevich, M., Stapleton, P. and Marquez-Borbon, A. 2010. Style and Constraint in Electronic Musical Instruments. *Proceedings of the International Conference on New Interfaces for Musical Expression*, Sydney, Australia, 106–11.

Hamano, T., Rutkowski, T. M., Terasawa, H., Okanoya, K. and Furukawa, K. 2013. Generating an Integrated Musical Expression with a Brain-Computer Interface. *Proceedings of the International Conference on New Interfaces for Musical Expression*, Daejeon, Korea.

Hook, J., McCarthy, J., Wright, P. and Olivier, P. 2013. Waves: Exploring Idiographic Design for Live Performance. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Paris, France, 2969–78.

Hunt, S., Mitchell, T. and Nash, C. 2017. How Can Music Visualisation Techniques Reveal Different Perspectives on Musical Structure. *International Conference on Technologies for Music Notation and Representation*, A Coruna, Spain.

Joaquim-Fernandes, V. and Barbosa, Á. 2013. Are Luminous Devices Helping Musicians to Produce Better Aural Results, or Just Helping Audiences Not to Get Bored. *Conference on Computation, Communication, Aesthetics and X*. Bergamo, Italy: xCoAx.

Kranzberg, M. 1986. Technology and History: 'Kranzberg's Laws'. *Technology and Culture* **27**(3): 544–60.

Lansky, P. 1990. A View from the Bus: When Machines Make Music. *Perspectives of New Music* **28**(2): 102–10.

Latour, B. 1987. *Science in Action: How to Follow Scientists and Engineers through Society*. Cambridge, MA: Harvard University Press.

Latour, B. 2005. *Reassembling the Social: An Introduction to Actor-Network-Theory*. Oxford: Oxford University Press.

Liang, B., Fazekas, G., McPherson, A. and Sandler, M. 2017. Piano Pedaller: A Measurement System for Classification and Visualisation of Piano Pedalling Techniques. *Proceedings of the International Conference on New Interfaces for Musical Expression*, Copenhagen, Denmark.

Lucier, A. 1998. Origins of a Form: Acoustical Exploration, Science and Incessancy. *Leonardo Music Journal* **8**(1): 5–11.

Magnusson, T. 2009. Of Epistemic Tools: Musical Instruments as Cognitive Extensions. *Organised Sound* **14**(2): 168–76.

Magnusson, T. 2011. Algorithms as Scores: Coding Live Music. *Leonardo Music Journal* **21**: 19–23

Magnusson, T. 2014a. Herding Cats: Observing Live Coding in the Wild. *Computer Music Journal* **38**(1): 8–16.

Magnusson, T. 2014b. Improvising with the Threnoscope: Integrating Code, Hardware, GUI, Network and Graphic Scores. *Proceedings of the International Conference on New Interfaces for Musical Expression*, London.

Magnusson, T. 2015. Code Scores in Live Coding Practice. *Proceedings of the International Conference for Technologies for Music Notation and Representation*, Paris, 5.

Magnusson, T. 2016. Presentation at New Notation Symposium, IRCAM, Paris, September.

Magnusson, T. 2018. Ergomimesis: Towards a Language Describing Instrumental Transductions. *Proceedings of the 4th ICLI*, Porto, Portugal.

Magnusson, T. 2019. *Sonic Writing: Technologies of Material, Symbolic and Signal Inscriptions*. London: Bloomsbury Academic.

Masu, R., Bala, P., Ahmad, M., Correia, N. N., Nisi, V., Nunes, N. and Romão, T. 2020. VR Open Scores: Scores as Inspiration for VR Scenarios. In R. Michon and F. Schroeder (eds.) *Proceedings of the International Conference on New Interfaces for Musical Expression*, Birmingham, UK.

Masu, R., Correia, N. N. and Romao, T. 2021. NIME Scores: a Systematic Review of How Scores Have Shaped Performance Ecologies in NIME. *Proceedings of the International Conference on New Interfaces for Musical*, Shanghai, China.

McLean, A., Griffiths, D., Collins, N. and Wiggins, G. 2010. Visualisation of Live Code. *Electronic Visualisation and the Arts (EVA 2010)*, 26–30.

McLean, A. and Wiggins, G. 2009. Patterns of movement in live languages.

McLean, A. and Wiggins, G. 2011. Texture: Visual notation for live coding of pattern. *Proceedings of the 2011 ICMC*, Huddersfield, UK.

McPherson, A. and Lepri, G. 2020. Beholden to our Tools: Negotiating with Technology While Sketching Digital Instruments *Proceedings of the International Conference on New Interfaces for Musical Expression*, Birmingham, UK.

Melbye, A. P. 2021. Resistance, Mastery, Agency: Improvising with the Feedback-Actuated Augmented Bass. *Organised Sound* **26**(1): 19–30.

Nash, C. and Blackwell, A. F. 2011. Tracking Virtuosity and Flow in Computer Music. *Proceedings of the 2011 ICMC*, Huddersfield, UK.

Orio, N., Lemouton, S. and Schwarz, D. 2003. Score Following: State of the Art and New Developments. *Proceedings of the International Conference on New Interfaces for Musical Expression*, Oslo, Norway.

Parkinson, A. and Bell, R. 2015. Deadmau5, Derek Bailey and the Laptop Instrument–Improvisation, Composition and Liveness in Live Coding. https://research.gold.ac.uk/id/eprint/12838/1/101_Deadmau5_Derek_Bailey_and_the_Laptop.pdf (accessed 14 July 2023).

Purcell, A., Gardner, H. and Swift, B. 2014. Visualising a Live Coding Arts Process. *Proceedings of the 26th Australian Computer-Human Interaction Conference on Designing Futures: The Future of Design*. New York: ACM, 141–4.

Rowlands, M. J. 2010. *The New Science of the Mind: From Extended Mind to Embodied Phenomenology*. Cambridge, MA: MIT Press.

Sarath, E. 1996. A New Look at Improvisation. *Journal of Music Theory* **40**(1): 1–38.

Stapleton, P. and Davis, T. 2021. Ambiguous Devices: Improvisation, Agency, Touch and Feedthrough in Distributed Music Performance. *Organised Sound* **26**(1): 52–64.

Tahıroğlu, K., Magnusson, T., Parkinson, A., Garrelfs, I. and Tanaka, A. 2020. Digital Musical Instruments as Probes: How Computation Changes the Mode-of-Being of Musical Instruments. *Organised Sound* **25**(1): 64–74.

Tomás, E. and Kaltenbrunner, M. 2014. Tangible Scores: Shaping the Inherent Instrument Score. *Proceedings of the International Conference on New Interfaces for Musical Expression*, London.

Varela, F. J., Thompson, E. and Rosch, E. 2017. *The Embodied Mind: Cognitive Science and Human Experience*, rev. edn. Cambridge, MA: MIT Press.

VIDEOGRAPHY

Hopkins, P. 2009. Amplified Gesture. *YouTube*. www.youtube.com/watch?v=0e60eKflPfo (accessed 4 January 2023).