On Data, Media In-Formations and Aesthetic Formatting in PERMUTATION

Ricardo Rodrigues Erl
r.rodrigueserl@gmail.com
University of the Arts, Berlin, Germany

Maximilian Ardalan Bernhard
maximilian.bernhard@sasberg.com
FH Joanneum / University of Music and Performing Arts, Graz, Austria

Michele Samarotto[1]
michele.samarotto@hfm-karlsruhe.de
Institute for Music Informatics and Musicology, University of Music, Karlsruhe, Germany

PERMUTATION is an audiovisual live performance framework that explores the interchangeability of digital data between sound and image. Adopting media-theoretical, artistic and technological perspectives, we investigate how the project explores the premise that, at the computational level, all media reduce to numerical code. Implemented through live-coded sound synthesis in SuperCollider and a custom video engine built with openFrameworks, PERMUTATION establishes a strict data-level coupling between the auditory and visual domains. The paper discusses notions of data as representation and intermediary, outlines the technical pipeline of the performance framework, and reflects on limitations arising from the presented approach to audiovisual coupling, such as differing data densities, polarity mismatches, and frequential dependencies. Drawing on Niklas Luhmann's concept of the distinction between *medium* and *form*, our artistic practice can ultimately be understood as an act of formatting: an aesthetic actualization of the digital medium as perceptible audiovisual forms.

1 Introduction

"With numbers, everything goes." (F. A. Kittler 1999, 2)

PERMUTATION is an audiovisual live performance framework written by Ricardo Rodrigues Erl and Maximilian Ardalan Bernhard that connects the auditory and visual domains on the data level.[2] This computational bridge is largely based on the notion of data interchangeability as afforded by digital media. PERMUTATION explores the possibilities and limitations of this notion by situating itself between strictly formal computation and perceptible, aesthetic association.

The paper aims to investigate the performance framework starting from the idea of data interchangeability. The development of PERMUTATION initially proceeded from the assumption that a complete interchangeability of data within a digital system afforded a direct correlation between sound and image in a live performance setting. The paper reflects on this premise from both a technical and media-theoretical perspective in order to examine to what extent data interchangeability is practically realizable and how far the performance framework succeeds in attaining this objective in the context of an artistic practice.

Drawing a theoretical baseline, we begin by reconstructing the aforementioned notion through media-theoretical, artistic and technological perspectives in order to understand the decisions made during

Keywords: Audiovisual Performance; Data; Live Coding; Cross-Sensory Coupling; Formatting; Medium and Form.

DOI 10.34626/2026_xcoax_019

—
1. All authors have contributed equally to this paper.
2. Code can be found under: https://github.com/erlerlerl/PERMUTATION (last accessed: 26.04.2026).

the development of the framework. Next, a technical description of the framework is given, focusing on the limitations of the simultaneous cross-sensory representation of data and proposing artistic strategies for adapting to and constructively engaging with these limitations. The artistic practice is then reflected in the light of Niklas Luhmann's system-theoretical distinction between *medium* and *form*, pointing to the act of formatting as a central operation.

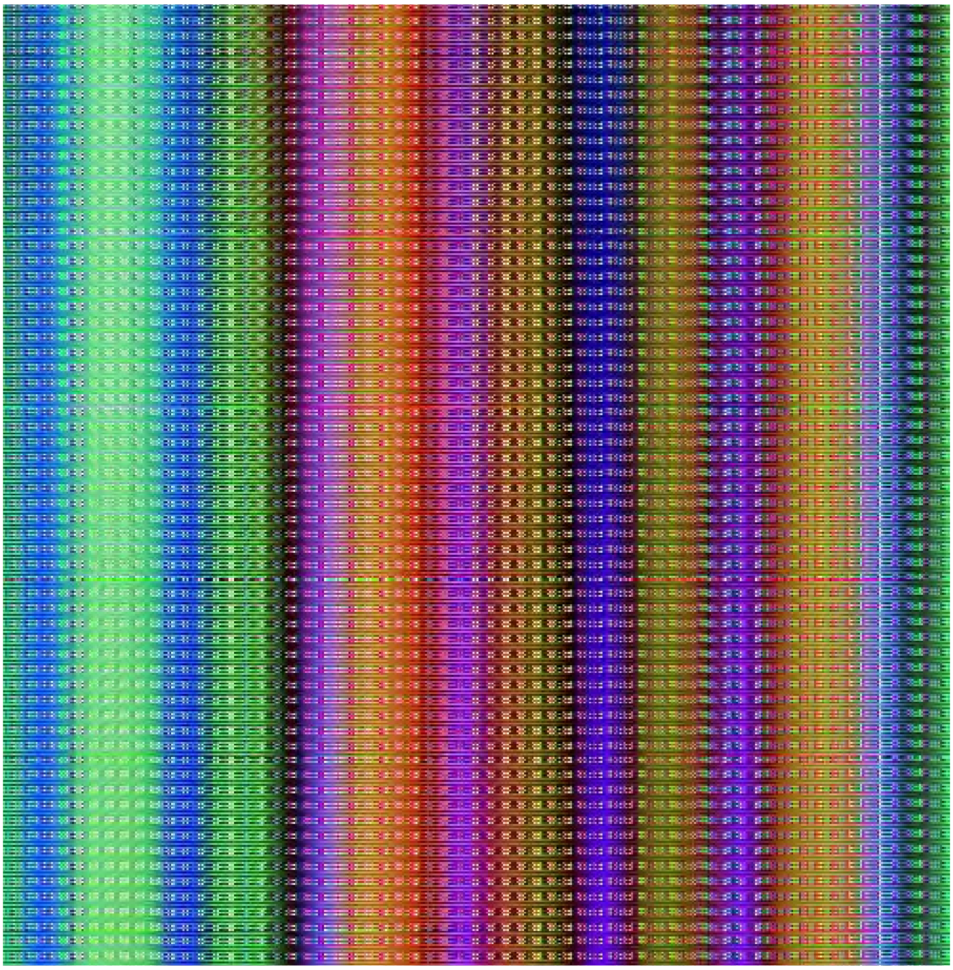


Fig. 1. Still from a live performance.

2 Theoretical Baseline

2.1 Data as Representation

What we mean by *data* is largely determined by the discipline in which the term is used. In most scientific fields data is understood as the result of representing some phenomenon in a certain way, often as a set of discrete and quantized numbers (Manovich 2015, 15). Measuring and sampling some phenomenon is required to perform computational calculations on it and thus examine it scientifically. It is important to note that the data is by no means equivalent to the underlying phenomenon, as the process of representation is a process of interpretation (Gitelman 2013, 3), limited by artificially defined boundaries (Manovich 2015, 15) and a great deal of filtering.

While these mandatory representational choices are widely discussed in data science, examining the resulting data requires a further transformation into a sensorially perceptible form, as “data itself is perceptually inaccessible” (Whitelaw 2004, 49).

2.2 Data as the Intermediary

Numerical code becomes the “new universal intermediary” (Manovich 2013a, 35), bridging the gap between the layer of sensorial perception and the layer of calculation. This allows for a situation where “[…] a digital representation of one type of media can also be translated into another media type that is meaningful to our senses” (Manovich 2013a, 36). A hint, perhaps, to the dissolution of the concept of separate media in the context of digital media, a mediatic conflation (see Section 2.3).

In practicality, however, this translational and interpretative flexibility is rarely given. The way we often operate with data is constrained by specific affordances of software, usually designed to fulfill specific tasks while ‘preventing’ other tasks (i.e. opening and displaying an image versus a sound file). Manovich somewhat boldly declares:

> There is no such thing as “digital media.” There is only software as applied to media (or “content”). Or, to put this differently: for users who only interact with media content through application software, the “properties” of digital media are defined by the particular software as opposed to solely being contained in the actual content (i.e., inside digital files). (Manovich 2013b, 152)

2.2.1 Glitch Art and a Rheology of Data

Accessing data in its raw numerical form seems to be an action not as easily accomplished. Glitch artist and theorist Rosa Menkman notes that:

> Computer programs only function when definite forms of formatted data are inserted: data for which the machine has codecs installed. (The value of) RAW, non-formatted or unprocessed data is easily dismissed because it is often hard to open or read. (Menkman 2020, 72)[3]

Practices related to *Glitch-Art* variously sought to overcome such restrictions in order “[…] to work beneath the previously impenetrable veil of digital media” (Cascone 2000, 12).

Such possibilities may be forced through disruptive interventions or unorthodox uses of certain systems (i.e. “hacks”) or provided by the affordances of specific or possibly custom-programmed pieces of software, such that “any selection of algorithms can be interfaced to pass data back and forth, mapping effortlessly from one dimension into another” (Cascone 2000, 17).

—
3. Page XVII of the essay titled *Refuse to Let the Syntaxes of (a) History Direct Our Futures*.

Glitch art practices often constitute an exciting entry point to creatively engage with digital technologies, emphasizing concepts such as spontaneity and randomness, but may also lead to more systematic investigations of the underlying algorithmic principles at hand, as is the case with the here-presented project. Carleigh Morgan remarks:

> [...] it is imperative to keep in mind how much the randomness, unpredictability, or messiness which glitch studies invokes around the glitch is in danger of overlooking the ways that the material architectures and algorithmic protocols structure the digital glitch by organising, constraining, and given form to its appearance. (Morgan 2019, 215)

Borrowing a term from physics, Rosa Menkman describes the concept of a *rheology of data*, which strongly resonates with the suggested ideas of freely accessing and molding data:

> By rheology of data I [...] mean a study of how data can be read and displayed in completely different forms, depending on the context, software or interface. A rheology of data facilitates a deformation and flow of the matter of data; it demonstrates the possibility of pushing data into different applications and the presentation of data in different forms. (Menkman 2020, 71)[4]

It is interesting to notice that such an approach to computation bears similarities to central concepts in the design of unixoid operating systems. Ken Thompson developed the first version of the *Unix* operating system at Bell Laboratories in 1969, drawing several ideas from an earlier operating system called *MULTICS* (Multiplexed Information and Computing Service) "including [...] the notion of files as unstructured streams of bytes" (Kerrisk 2010, 2). Based thereon, different models of *IPC* (Inter-Process Communication) originated (e.g., pipes, FIFOs, sockets).

2.3 Data as the Interchangeable

Once the surface cracks open, we gain access to the underlying rheological data currents. Data 'matter' can now become building blocks for new mediatic couplings, an image file might be interpreted as an audio waveform, an executable rendered as image, etc.; all courtesy of immaterial code, mathematical descriptions.[5]

Friedrich Kittler writes:

> The general digitization of channels and information erases the differences among individual media. Sound and image, voice and text are reduced to surface effects, known to consumers as interface. [...] Inside the computers themselves everything becomes a number: quantity without image, sound or voice. And once optical fiber networks turn formerly distinct data flows

—

4. Respectively page XVI, *ibid*.
5. Here a discussion about the different concepts of materiality as applied to the digital may be expedient.

> into a standardized series of digitized numbers, any medium can be translated into another. With numbers, everything goes. (Kittler 1999, 1f.)

The metaphorization of the computer as a *universal medium*[6] can further be contextualized from a practical computer science perspective:

> All information in a system—including disk files, programs stored in memory, user data stored in memory, and data transferred across a network—is represented as a bunch of bits. The only thing that distinguishes different data objects is the context in which we view them. For example, in different contexts, the same sequence of bytes might represent an integer, floating-point number, character string, or machine instruction. (Bryant and O'Hallaron 2011, 3)

While again emphasizing the universality of bits as the language of the computer, there are further ideas worth noting here. The differentiation of the term *user data* and the *data object* broadens the data term formulated in Section 2.1. The inclusion of computer programs or network traffic clarifies that not all data originates in real-world phenomena. From this point of view, everything that is represented by a sequence of bytes constitutes a *data object*.

The second important aspect is the receptive context that distinguishes data objects. As the notion of interchangeability enables the same sequence of bytes to represent different data objects, we as users need additional information to accurately perceive the intended data object encoded in the sequence.

3 PERMUTATION

3.1 Audiovisual Context

As we've already discussed in Section 2.2.1, the interchangeability of data inside the computer offers a rich potential for artistic exploration. Especially in the field of audiovisual arts, the intermediary role of data provides unique possibilities for bridging the auditory and visual domains. PERMUTATION seeks to explore this exact process of audiovisual bridging at the data level and constitutes a live performance framework based on live-coded sound in SuperCollider[7] and a custom-written video engine in openFrameworks.[8]

While audiovisual art is often based on 'deliberate' mappings between sonic and visual parameters, the act of bridging is subjected to media-specific characteristics. The spectrogram as a basic example of an audiovisual mapping displays the frequency spectrum over time of a given (audio) signal as a two-dimensional graph, where the position

—
6. See also: Schröter 2002, url: https://www.theorie-der-medien.de/text_detail.php?nr=46 (last accessed 26.04.2026).
7. https://supercollider.github.io (26.04.2026).
8. https://openframeworks.cc (26.04.2026).

on the x-axis represents a specific point in time, and the position on the y-axis a specific frequency. Mainly used as a tool for analysis, it offers great insight into the nature of the incoming signal. However, this audiovisual mapping makes media-specific assumptions as it is targeted towards representing the media-specific concept of a frequency and is therefore already the product of a specific receptive context.

3.1.1 Classification of Audiovisual Couplings

To systematically investigate how audiovisual couplings can be organized and compared, we refer to Michael Haverkamp's taxonomy for a general classification of cross-sensory couplings. Michael Haverkamp distinguishes five different types of couplings: *genuine synaesthesia, cross-sensory analogies, iconic coupling/association, symbol/metaphor* and *mathematical/physical correlation* (Haverkamp 2010). These strategies are differentiated by the extent to which they are consciously perceived or intuitively felt. While genuine synaesthesia originates as an intuitive impression, a mathematical correlation calls for a conscious decision to establish a formal connection.

Considering our aim of exploring the bridging between the auditory and visual domain on data level (i.e. algorithmically), the cross-sensory coupling in PERMUTATION can be located within mathematical/physical correlations. Haverkamp notes additionally that "[...] algorithmic or physical approaches of cross-sensory coupling need to additionally present 'intuitive' connections on the basis of analogies as well as of iconic and symbolic features [...]". (Haverkamp 2010) In PERMUTATION this "intuitive" connection lies in the time synchronicity. Further explained in the following sections, the core idea of the framework is built around synchronizing the audio data to a video stream, a process that is by no means trivial.

3.2 Conversion Concept

To understand our aim to work as close to a purely mathematical correlation as possible, the following paragraph examines the computational layer of an audio as well as a video signal. Digital audio streams are most often computed as a time- and value-discrete signal in the range −1 to 1, with a predefined floating-point precision, wherein a sustained audio stream of samples with the same value is perceived as silence (DC-offset). In a typical example, an audio stream is represented by 44,100 discrete values per second, as specified by the sample rate, which defines the temporal resolution of the audio signal. Accordingly, we can consider an audio signal as a one-dimensional function $f(t)$, defining a value for each given point in time t.

An 8-bit uncompressed black and white image is computed as an array of numbers, where each number, ranging from 0 (fully black) to 255 (fully white), represents the brightness value of one pixel of the image. We can thus define an image as a two-dimensional function $f(x,y)$, describing brightness values at given pixel coordinates in the frame.

Consequently, a video signal, representing a continuous stream of images, can be defined similarly, expanding the image function to include a time variable $f(x,y,t)$, so as to compute the brightness value for a specific pixel at the coordinates x and y at a given point in time t.

As the above definitions of audio and video signals define the smallest digital unit of the respective surface effect, to use Kittler's term (see Section 2.3), we decided to bridge the auditory and visual domains at this fundamental layer, mapping the value of each audio sample to a pixel on the screen.

3.3 Framework Duality

With SuperCollider as a real-time data creation and data processing tool and openFrameworks as the video engine in the background responsible for the data conversion, the framework is divided into two components with fundamentally different functionality. This division is not only found in the components' targeting of different sensory domains but is also emphasized in the technological characteristics of the used programming frameworks themselves.

With the introduction of JITLib,[9] SuperCollider has emerged as a live-codable audio programming framework and therefore bears the potential to be used in a live performance situation. In contrast to this, openFrameworks is based on C++, a compiled programming language. This severely restricts any real-time capabilities, as the code needs to be compiled into binary format before it can be run by the computer. Any performative flexibility thus needs to be defined (i.e. programmed) beforehand and cannot be modified during the performance.

As the openFrameworks engine is responsible for the data conversion of the audio stream to video data, the actual conversion process stays fixed during the performances and is only changed between performances to allow for different artistic explorations.

4 Limitations

In relating our ideal vision of the pipeline to the actual realization of the technical setup, we encountered limitations that had a considerable influence on the development of PERMUTATION.

4.1 Data Density / FIFO Principle

One of the main problems that arises when file sizes of audio versus video are considered, is different data densities. Using the following formulas we can highlight the problem. We define b as the data rate in $\frac{bits}{s}$, f_{sr} as the audio signal's sample rate, k_{audio} as the audio signal's bit depth, and c as the number of audio channels.

—
9. See: Wilson et al. 2025, Chapter 7.

$$b_{audio} = f_{sr} \cdot k_{audio} \cdot c$$

For the visual domain we define f_{fr} as a whole frame's update rate, k as the bit depth per color channel of the pixels, and N as the side length of a square-shaped frame.

$$b_{video} = f_{fr} \cdot k \cdot 3 \cdot N^2$$

When we compare values for a standard audio format versus a standard video format, we can easily demonstrate the underlying problem. A 3 minute long stereo audio file in CD quality contains $169.344\ Megabits = 44100 \frac{1}{s} \cdot 16bit \cdot (3 \cdot 60)s$, while a Full HD video of the same length contains $2.239.488\ Gigabits = 3 \cdot 8bit \cdot 1920 \cdot 1080 \cdot 25\frac{1}{s} \cdot (3*60)s$ of raw pixel data.[10] This conclusively shows the 4 orders of magnitude in difference between data densities. Following this, we decided on implementing a scrolling image buffer, based on the FIFO principle, wherein a new row of pixels is added to the bottom of the image buffer, while the oldest is erased from the buffer. This way we avoid having to resort to slow framerates in order to keep time synchronicity.

As the engine is reading floating-point sample values instead of the bit sequence itself, we can discard any bit depth considerations. In this regard, the only effect the bit depth has is the resolution of our floating-point values. This also simplifies the mapping of audio to pixel data, such that each sample corresponds to one pixel in the image.

As we decided to work with color, without taking any perceptual considerations into account, rather focusing on its mathematical description, we chose to use three audio channels, each corresponding to one of the three color components of an RGB-encoded pixel. Adjusting our previous formula to our new FIFO scrolling image buffer, the buffer size (l) of the audio signal determines the rate at which new rows appear (f_r), as well as the size of our frame, as each sample in the buffer provides the data for one pixel,[11] and the whole buffer maps to exactly one row.

$$N = l$$

$$f_r = \frac{f_{sr}}{l}$$

This means that if we wanted to synthesize audio at $f_{sr} = 48\ k\,Hz$ with our buffer size set to $l = 512\ samples$ we'd have a row update frequency of $f_r = 93.75\ Hz$. To calculate the time it takes until a frame has been fully filled with data, we multiply $\frac{1}{f_r}$ with h resulting in $5.5613\ s$.

—

10. These examples assume uncompressed data. As PERMUTATION maps individual samples to individual pixels the opportunity to deploy a visual compression to reduce required data size is very restricted and therefore not considered here.

11. More precisely, *for one color channel of one pixel*.

Fig. 2. Half a cycle of a bipolar sine wave, with $f = 0{,}5 \cdot f_h$, played only on the red (positive part) and cyan (negative part) channel.

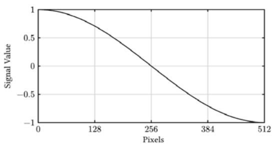


Fig. 3. Plot of the signal producing Fig. 2.

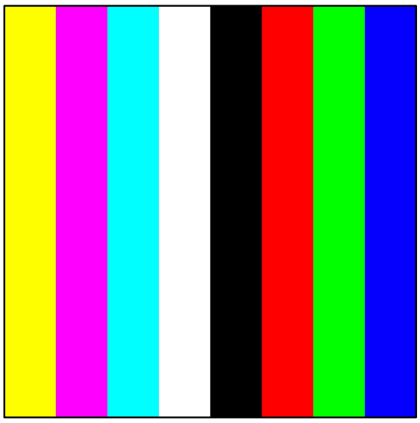


Fig. 4. Demonstration of color channels mixing, to create different colors, with $f = f_h \cdot 8$.

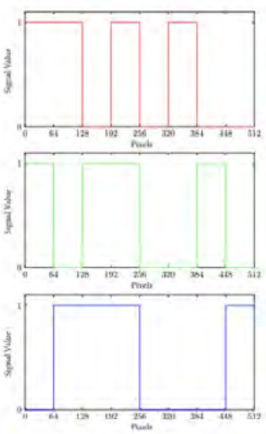


Fig. 5. Channel data of Fig. 4.

4.2 Polar Gap / The Notion of Silence

Based on the idea of a one-to-one relation between the audio and video data, in our case a bipolar audio signal and an RGB-encoded pixel frame, we encountered the problem of not only having different ranges but also polarities of input and output streams. A series of questions arose: Do we just discard data in the range of −1 to 0? If yes, what does discarding imply in the context of a data stream? Do we 'stretch' the rest of the data to fit the range of 0 to 255? Another fundamental question pertains to the notion of silence: How do we correlate auditory silence, as in a lack of air pressure change, to visual 'silence', as in the absence of light?

As auditory silence isn't bound to any specific value, but only defined through the lack of change,[12] it was easier to change our sound generation approach than it was to translate perceived auditory silence into its visual correlate, in the context of a real-time performance. Furthermore, this reflected our purist approach to data representation in the most direct possible way. Nonetheless, a different approach crystallized in later development stages, mapping positive values to an additive RGB color space and negative values to a subtractive CMY color space.[13] Using two color spaces simultaneously seems counter-intuitive at first, especially concerning the technical realization of this idea. As a matter of fact, the CMY color space only exists theoretically, as it can be simulated by inverting the values of the RGB space. For example, cyan CMY: 255/0/0 can be represented as RGB: 0/255/255. Consequently, every negative value of a specific audio channel is not applied to the corresponding color channel but to the other two.

This technological solution, however, bears its own problems, as negative audio values now impact the visual domain by twice as much, applying the value to two color channels instead of one, thereby increasing the overall brightness of the pixel (determined by the average value across all color channels) by twice as much. For this reason, we decided to only apply half the negative value to each color channel to preserve an equal 'numerical impact'.[14]

4.3 Frequential Dependencies

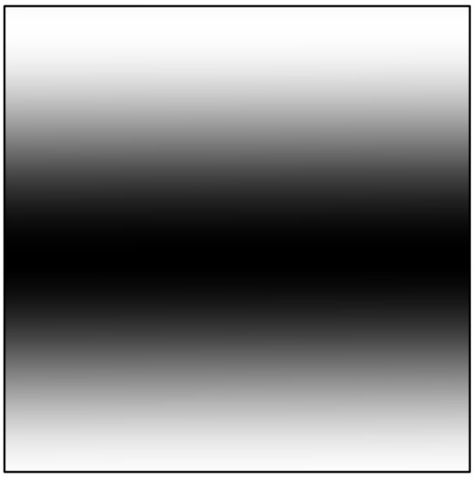
Fixing the parameters f_{sr} and N makes it possible to calculate frequential dependencies that link audio and video, and thus our underlying data together even more tightly. Considering the following two frequencies f_h, where one period of a waveform fills one row, and f_y, where one period matches the height of the whole image buffer, we as performers have new opportunities to 'draw' with our data.

—

12. Audio, unlike video, is perceived as changes in air pressure, which can be understood as the derivative of the source signal. While this observation warrants further consideration, it is left aside here.
13. This approach still only depicts silence in both media if the data stream is made up of zeros, but it also opens up new possibilities in displaying the difference of negative to positive values.
14. It is important to note here, that this solution is solely based on numerical consistency and is not intended to reflect perceived brightness.

$$f_h = \frac{sampleRate}{frameWidth}$$

$$f_v = \frac{sampleRate}{frameWidth * frameHeight}$$



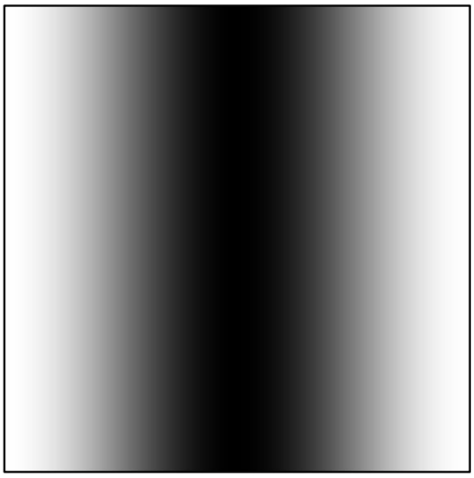


Fig. 6. One cycle of a unipolar cosine wave, appearing vertical, with $f = f_v$

Fig. 7. One cycle of a unipolar cosine wave, appearing horizontal, with $f = f_h$

5 Reflections

Reflecting on the design choices explained in Section 4, it seems that the notion of 'freely' interchangeable data, as delineated in Section 2 has weakened. This realization is especially relevant since at the beginning of the project we set out to explore this very interchangeability. With reference to Haverkamp's classification of cross-sensory couplings, the technical solutions to the limitations we faced steer away from a strictly *mathematical correlation*. Especially the decision to translate perceived auditory silence into a visual one (see Section 4.2) would probably be considered as a *cross-sensory analogy* (ignoring *genuine synaesthesia*).[15]

This begs the question whether the way we developed the PERMUTATION framework still leaves traces of the original idea of exploring strictly mathematical correlations between cross-sensory couplings as a way of exploring the interchangeability of data as formulated in Section 2. To examine this question, the following paragraph focuses on Niklas Luhmann's system-theoretical distinction between *medium* and *form*.

5.1 Form (Luhmann)

While audio and video manifest as surface effects of the digital machinery, targeted to sensorial perception, it is their mediatic substrate that opens up countless possibilities for creative intervention. Niklas Luhmann's system-theoretical concepts of *medium* and *form* seem to be a very useful conceptual frame to be applied to a data-level approach to

—

15. The classification of *genuine synesthesia* is ignored here, as Haverkamp himself states, that this category is still under research (Haverkamp 2010).

audiovisual coupling. Luhmann's distinction between medium and form is thought of in terms of loose and tight couplings of elements.

> The term element does not refer to natural constants [...] Rather, it always points to units constructed (distinguished) by an observing system – to units for counting money, for example, or to tones in music. (Luhmann 2000, 103)

As the notion of a medium refers to a loose coupling of elements, it "[...] indicates an open-ended multiplicity of possible connections that are still compatible with the unity of an element – such as the number of meaningful sentences that can be built from a single semantically identical word" (Luhmann 2000, 104).

A medium opens up a space of combinatorial possibilities. Forms, on the other hand, as tight couplings of elements, are "selective actualizations of a medium"[16] (Krämer 1998, 566). What results as a tightly coupled form in one medium can itself become a loosely coupled medium for other forms.

5.2 Formatting / In-Forming

A medium as such then cannot be observed directly except through forms that tightly couple its elements, does it become perceivable, just as digital code remains invisible until it seeps to the surface to reach our senses. Audio and video actualize the digital medium as performative and contingent forms. To in-form a medium becomes a compositional act, an aesthetic choice. The notion of a format itself can be thought of as a demarcation between medium and form. With reference to Luhmann,[17] Michael Niehaus states:

> "It then turns the medium into a medium by arranging the elements in such a way that it becomes 'receptive'. In this sense, formatting is 'pre-formatting': the elements are formed in such a way that the nature of their coupling proves to be loose for another form." (freely translated from Niehaus 2018, 43)[18]

6 Conclusion

Niklas Luhmann's distinction between medium and form lets us reformulate the question brought up at the beginning of Section 5. Understanding the act of formatting as a demarcation between medium and form, thereby actualizing the distinction, recontextualizes the idea of data interchangeability. As the process of converting (audio) data into a

—

16.Original: selektive Aktualisierung eines Mediums.

17. "Media are generated from elements that are always already formed. Otherwise, we couldn't speak of their loose or tight coupling" (Luhmann 2000, 106).

18. Original: Es macht dann das Medium zum Medium, indem es die Elemente so anordnet, dass es ›aufnahmefähig‹ wird. Formatierung ist in diesem Sinne ein ›Vorformatieren‹: Die Elemente werden so formiert, dass sich die Art ihrer Kopplung als lose für ein [sic] andere Form erweist.

video stream ultimately represents a process of formatting, it is possible to argue that the data takes on the role of a medium in a Luhmannian sense. The auditory and visual domains can thereby be seen as actualizations of this medium. Searching for traces of data interchangeability in the performance framework consequently resembles a search for the medium in one of its actualized forms. In Luhmann's theoretical complex, this question is surprisingly easy to answer. As the relation between medium and form is codependent, the medium and therefore the notion of data interchangeability is a prerequisite for any actualized form. The developed solutions merely indicate the significance of understanding media and their actualized forms as products of a differentiating operation.

PERMUTATION ultimately reflects the notion of data interchangeability by treating the act of formatting as an aesthetic intervention. Surface effects then are the product of in-forming the digital medium, made possible by the interchangeability of data as loosely coupled elements.

Acknowledgements. We would like to thank Christoph Seibert and Patrick Borgeat for providing valuable input and critical reflection.

References

Bryant, Randal E., und David R. O'Hallaron.
2011. *Computer Systems: A Programmer's Perspective*. Prentice-Hall.

Cascone, Kim.
2000. "The aesthetics of failure: 'Post-digital' tendencies in contemporary computer music". *Computer Music Journal* 24 (4): 12–18.

Gitelman, Lisa.
2013. *"Raw Data" Is an Oxymoron*. Infrastructures Series. MIT press.

Haverkamp, Michael.
2010. "Synesthetic approach for evaluation of the cross-sensory quality of multi-media applications". *2010 Second International Workshop on Quality of Multimedia Experience (QoMEX)*, June, 136–141.

Kerrisk, Michael.
2010. *The Linux Programming Interface: A Linux and UNIX System Programming Handbook*. No Starch Press.

Kittler, Friedrich A.
1999. *Gramophone, Film, Typewriter*. Stanford University Press.

Krämer, Sybille.
1998. "Form als Vollzug oder: Was gewinnen wir mit Niklas Luhmanns Unterscheidung von Medium und Form?" *Rechtshistorisches Journal* 17: 558-573.

Luhmann, Niklas.
2000. *Art as a Social System*. Stanford University Press.

Manovich, Lev.
2013a. "Media After Software." *Journal of Visual Culture* 12 (1): 30–37.

Manovich, Lev.
2013b. *Software Takes Command*. International Texts in Critical Media Aesthetics. Bloomsbury Academic.

Manovich, Lev.
2015. "Data Science and Digital Art History." *International Journal for Digital Art History 1*.

Menkman, Rosa.
2020. *Beyond Resolution*. Website. https://beyondresolution.info/beyond-resolution (last accessed 26.04.2026)

Morgan, Carleigh.
2019. "Calculated error." *A Peer-Reviewed Journal About* 8 (1): 204–217.

Niehaus, Michael.
2018. Was ist ein Format? Wehrhahn Verlag.

Schröter, Jens.
2002. “Intermedialität, Medienspezifik und die universelle Maschine.” https://www.theorie-der-medien.de/text_detail.php?nr=46 (last accessed 26.04.2026)

Whitelaw, Mitchell.
2004. “Hearing pure data: Aesthetics and ideas of data-sound.” In *Unsorted: An A to Z for Sonic Acts X*, edited by Arie Altena, 45–53. Sonic Acts/De Balie.

Wilson, Scott et al.
2025. *The SuperCollider Book*. MIT Press.