# BeatSketch: Combining digital and physical user interfaces

VLAD HAȚEGAN, Department of Computer Science, Eindhoven University of Technology, The Netherlands


Fig. 1. BeatSketch with instrument blocks

This paper presents an innovative approach of musical note manipulation in a DAW (digital audio workstation) using a combined physical-digital interface that makes use of computer vision, allowing users to compose short "bars" in an interactive, intuitive manner. The device is composed of a screen on which cubes featuring CV markers can be placed, their position on the X and Y axes controlling the pitch and timing of a sound, the rotation controlling the duration of it and the specific ID of a cube (marked by its color) representing the kind of sound it produces.

Additional Key Words and Phrases: computer vision, user interface, music, intuition

## 1 INTRODUCTION

*Instrument for experts.* Currently available music-making interfaces make use of complex systems of predefined knobs, keys and buttons. While those standardized interfaces [12] make the process of transforming thoughts into music seamless for professionals, there is a certain creative aspect that is cut off when the comfort level gets too high and the window of possibilities is shifted. BeatSketch represents a fresh creative departure from the well-defined, into the new and exciting, creating a new type of musical canvas that artists can use.

*Beginner opportunity.* On the other side, beginners and artistic outsiders lack the intrinsic connection that experts have built with the standard interfaces, such as the piano layout. For such a user, an intuitive interface might represent an opportunity to express the inner creativity that is otherwise held back by the lack of capability of expression.

*Reinvention.* BeatSketch reinvents music-making interfaces, by exchanging the predefined controls for a more fluid digital-physical interface, based on intuitive interactions and accessible affordances[6], that do not require the depth of the cultural meta-context that surrounds currently available products.

Concepts such as piano keys, pitch knobs and drum pads are replaced with "instrument blocks" that can be placed in ways that construct a simple beat.

## 2 RELATED WORK

*Design inspiration.* The core concepts that BeatSketch is built on are inspired by devices such as the Reactable[8], while keeping the interface more in-line with pre-existing DAWs [12]. Further inspiration comes from music trackers, such as Nitrotracker[3] and music-creation interactive experiences, such as Electroplankton[9].

*Technical workings.* On a technical level, the project exists thanks to the work done towards the Python programming language, OpenCV[10] and ArUco markers[5].

## 3 DESIGN AND IMPLEMENTATION

*Constraints breed creativity.* Similarly with how the device is meant to be used: a creative constraint [11], the design process was centered around maximizing the usability of a seemingly random set of devices from a personal parts bin.

*Design concepts.* The project heavily relies on the concept of affordances and intuitive interfaces[6]. BeatSketch creates a canvas on which note "blocks" (physical) of different colors-flavors can be placed, with graphics being shown underneath the blocks (digital). The classical western intuition of sounds being "high" or "low" [2]and of time flowing from the left to the right [4] is harnessed here. When the user places a cube on the screen canvas, its corresponding sound is played within a "bar" of music, with the starting timestamp of the sound being represented by its X axis position and the pitch by its Y axis position. While this takes a slight amount of exploration, rotating the cubes affects their pitch. This is visible via the changing width of the projected graphic under each cube.

*Making sure the experience is seamless.* Since sound playback, interface drawing and CV-marker detection are all real-time tasks, each of them runs in a separate thread. Compared to a single-threaded implementation, which capped the maximum beats per minute to around 90, this implementation allows for any speed to be played in real-time, irrespective of the speed of image acquisition. A frustrating interface does not invite the user to explore [7] and delays in the feedback might make the user think that their approach is wrong, hindering exploration. We invite the reader to inspect the code, available on GitHub.

## 4 RESULTS AND DISCUSSION

The rough prototype presented during the Midterm presentation fully implements the basic functionality described above on a baseline level, as a proof-of-concept.

The user uses color-coded cubes (Fig. 2), each color representing an instrument, here implemented as square, sine and triangle waves.

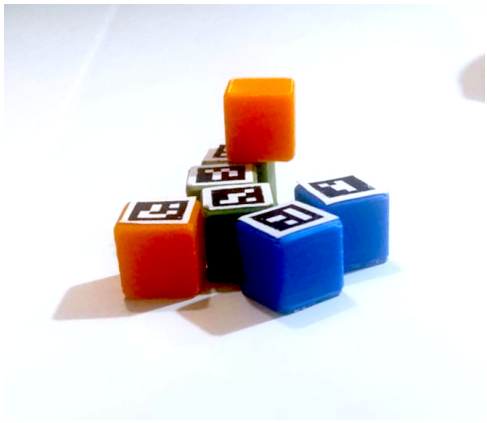

Fig. 2. Instrument cubes, of different colors

In Fig. 3, the basic UI elements are visible: the BPM display, the individual notes, with their color, position and duration, and the sweeping line, that represents the timestamp that is being played at a certain moment.

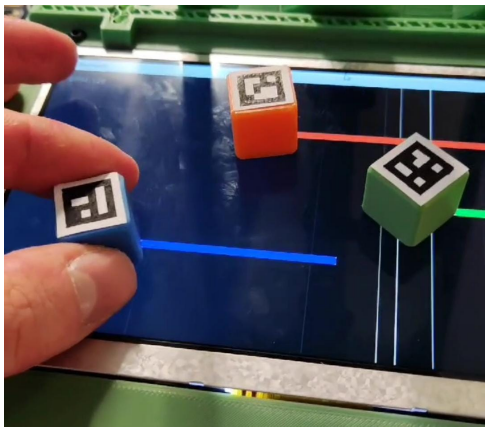

Fig. 3. Interface showcase

The prototype also features four different buttons: two for modifying the BPM, one for "holding" the current notes and one for staring and pausing the playback. The hold function works by freezing the current position of the notes, showing them in a darker color, and playing them irrespective of what the current representation is on the screen. When the user depresses the button, the new configuration becomes the active one. This allows the user to continue playback of a current configuration, while allowing them to compose a new melody.

The musical distance between the bottom and top of the screen is equal to one octave, and the resulting frequency follows a logarithmic function based on the Y position of each block. [1]

The working product can be checked here. The code repository is available here.

## 5 CONCLUSION

While testing is limited, experiments with friends and colleagues deemed the interface easy to use and intuitive. Refinement is needed, and features such as snapping the cubes to notes and a larger canvas have to be included in a future iteration.

## 6 FUTURE WORKS

Future improvements to the current prototype are welcome. While satisfaction with the concept is high, developing it from a designer perspective can only be done through a multitude of iterations and tests with target users, something that the limited time did not allow.

Some potential improvements would be using a larger screen, implementing notes that snap to a grid, adding more instrument types and small overall tweaks.

## 7 A NOTE ON AI USAGE

AI was not used in any way in the making of this report. Copilot and, to a lesser extent, ChatGPT were used during programming to speed up repetitive tasks and to research ways of technically implementing certain features. All design considerations and interaction ideas were the product of a human mind.

## ACKNOWLEDGMENTS

Many thanks to my housemate Vlad Muntean for taking some of the photos of the product while I was away, and to Alexanda Voinea for helping me flash the STM32 microcontroller, used for the control keys, with the correct firmware.

## REFERENCES

[1] 2022. https://en.wikipedia.org/wiki/Cent_(music)

[2] 2024. https://www.psychologicalscience.org/publications/observer/obsonline/metaphors-for-musical-pitch-vary-across-languages-but-the-basic-principles-are-the-same.html

[3] 0xtob. 2015. nitrotracker. https://github.com/0xtob/nitrotracker

[4] Lera Boroditsky. 2000. Metaphoric structuring: understanding time through spatial metaphors. *Cognition* 75, 1 (Apr 2000), 1–28. https://doi.org/10.1016/s0010-0277(99)00073-6

[5] S. Garrido-Jurado, R. Muñoz-Salinas, F.J. Madrid-Cuevas, and M.J. Marín-Jiménez. 2014. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognition* 47, 6 (Jun 2014), 2280–2292. https://doi.org/10.1016/j.patcog.2014.01.005

[6] James J. Gibson. 1967. The Senses Considered as Perceptual Systems. (1967).

[7] Morten Hertzum. 2010. Frustration: A common user experience. https://www.researchgate.net/publication/228975627_Frustration_A_common_user_experience

[8] Sergi Jordà, Günter Geiger, Marcos Alonso, and Martin Kaltenbrunner. 2007. The reacTable. *Proceedings of the 1st international conference on Tangible and embedded interaction - TEI '07* (2007). https://doi.org/10.1145/1226969.1226998

[9] Ltd. Nintendo Co. and Toshio Iwai. 2005. ELECTROPLANKTON. http://electroplankton.com/ via Archive.org.

[10] opencv.org. 2021. opencv/opencv-python. https://github.com/opencv/opencv-python

[11] Kate Torgovnick May. 2013. Can limitations make you more creative? A QA with artist Phil Hansen. https://blog.ted.com/can-limitations-make-you-more-creative-a-qa-with-artist-phil-hansen/

[12] Yutong Yang. 2024. Analysis Of Different Types of Digital Audio Workstations. *Highlights in Science, Engineering and Technology* 85 (Mar 2024), 563–569. https://doi.org/10.54097/6vvy8z41