THE MANO CONTROLLER: A VIDEO-BASED HAND-TRACKING SYSTEM

Jaime E Oliver LR

University of California, San Diego
Music Department
jaime.oliver@gmail.com

ABSTRACT

The MANO Controller is the second controller in the Silent Percussion Project. It consists of an illuminated rectangular black surface which is captured in video from above. When hands are present in the captured area, the image is analyzed to find them and extract multiple parameters. This paper explores the role of the hand in computer music, briefly exploring previous approaches to hand tracking. It then describes the interface and algorithm of the MANO controller as well as some of the guiding principles of the Silent Percussion Project.

1. THE HAND

The spanish word for hand is "mano". The word for "playing" an instrument or for "playing" music is the word "touch". One "touches" an instrument or a song.

The relevance of the hand in the development of humans and their cultural activity has not been studied to the depths it deserves. F. Wilson [11] suggests that there is a parallel growth of the brain and the shifting of the thumb into its current position. This shifting of the thumb is what allowed humans to use tools. The role of hand gestures and body language in communication is not auxiliary, but content itself.

Hands are indispensable for music. We use them to manipulate the environment to make sounds. In a way, all musical instruments (except perhaps the voice) consist of some sort of object manipulation with our hands. Acoustic instruments are shaped according to acoustic needs and to accommodate our bodies (our mouths, hands, etc.); that is, they reflect our anatomy. Computer music instruments don't need to accommodate the sound producing mechanisms; they need to focus on the interface and mappings; they only need to reflect our anatomy.

Tracking hands then, should be a priority in computer music instrument research.

2. DIRECT VS INDIRECT HAND-TRACKING

Almost every controller (keyboards, wind controllers, drum pads, etc.) track hands. However, when we use a joystick or mouse, we don't directly track the movements of a hand, but the effects that these movements have on the interface. In this model, we could talk of *indirect hand-tracking*, since the aim is to track the movements of a device as a result of hand manipulation.

Few interfaces aim to directly track the gestures of a hand. That is, by tracking gestures that are produced without manipulating a device, but with the intention of being tracked as a means to make sounds. In this model, we could talk of *direct hand-tracking*.

As it can be deduced, direct and indirect tracking are not fixed categorical distinctions, but ends in a continuum. While indirect tracking needs a device for manipulation, direct tracking needs an interface as transparent as possible, making the hand the device itself.

3. PRIOR WORK

The oldest instrument that directly tracks hand gestures is the theremin. It provided two variables: the distance of the closest part of the hand to the sensors. This allowed for occlusion as a performative opportunity. The theremin was culturally validated because of its ability to perform traditional classical repertoire of pitches and durations with expressive articulation and its stable, and therefore recognizable, sound mapping.

Other approaches include the use of gloves such as L. Sonami's Lady Glove. Although extremely rich in the number of variables they provide, gloves also obstruct the mobility of the hand itself and the visibility of the gesture, which would otherwise provide more information to the audience. A similar device is M. Waisvisz's legendary *Hands*.

Another approach consists of multitouch tracking tables such as the Reactable by S. Jordá [3]. In most cases these interfaces aim to track the points of contact of the hand with the table (generally finger tips). In the case of the Reactable, the hand is tracked indirectly though the manipulation of blocks with particular shapes called *fiducials*.

Another multitouch approach is the use of trackpads, of which the most advanced is the *Slabs* controller by D. Wessel [10]. The aim is to track contact points as well, although with two important advantages: the independent sensing of

pressure and high sampling rates, which reduce latency and jitter.

Finally, I should include the Silent Drum [5], the first controller of the Silent Percussion Project. In this case, the aim is to track hand shapes and trajectories, obtaining several variables by manipulating an elastic head.

Although there have been attempts at directly tracking hands with video, these have not always been designed for real-time computer music performance.

4. WHY VIDEO?

Digital Video has often been considered an unfit sensor for musical performance. Cameras were limited to low frame rates and computers were not fast enough to transmit and process high frame rates in real time. Faster processors and multi-core architectures along with cheap, high frame rate cameras now allow us to obtain reasonably low latency and jitter.

The question then is why to use video in the first place? In acoustic music, gestures are inseparable from the sound they produce, since it is the result of direct transfers of energy. Sound is invisible; gestures are the visual aspect of sound. Gestures have strong visual and spatial components that contain information about the sounds they produce or will produce; these are easily apprehended by the audience, who expects this information to be coherent with the sounds.

A long standing concern of electronic music is that an audience cannot to recognize the source of a sound. Sounds have two kinds of sources, the vibrating body and the action that sets it in motion. One of the contributions of controllers is to provide a source through gestures.

The camera attempts to fulfill the audience's expectations by "watching" and analyzing gestures and translating them onto sounds. Through the camera, the computer translates gestures (sources) into sounds.

Finally, there is a constant complaint about the inability of video to capture a 3 dimensional world. The captured image is a 2 dimension reduction of a 3 dimensional space. Sensing multiple parameters of this space provides us with a 3 dimensional behavior even when these are obtained from a 2 dimensional image.Video produces occlusion. Although it is usually seen as a limitation, it could also be seen as a characteristic of the sensor that becomes a performance opportunity, as in the theremin.

Although the most problematic of all features is having low frame rates, it has been my experience that both the audience and the performer can adapt through practice and performance.

5. THE INTERFACE

The interface is simple and robust. A diagram and picture of the first prototype can be seen in figure 1. It is a thick wooden rectangle wrapped in a black textile. The textile absorbs light, contrasting skin which reflects it.

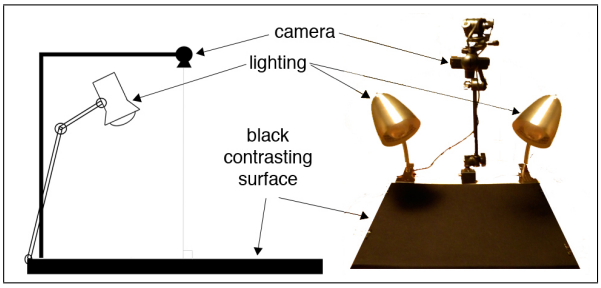

Figure 1. First prototype and Diagram of the MANO controller.

The surface is illuminated from above by two light sources. Other lighting strategies, with more sources and higher positions need testing. The camera is positioned on top of the surface at a right angle. Although it is possible, and for some gestures helpful, to use the surface as a resting point, gestures can be performed in the open air.

The interface is in way interface-less. Instead of an object for manipulation we have a space of action.

6. THE IMAGE ANALYSIS ALGORITHM

The algorithm is designed with real-time operation in mind, "rastering" the image once per frame and working in greyscale. The image analysis algorithm can be subdivided into image treatment and analysis processes. Further information can be extracted through learning techniques. The MANO controller has been programmed in the Pure Data [6] and GEM [1] environments with several custom made externals.

6.1. Image Treatment

The image is "rastered" once performing a contrast algorithm based on a threshold and an edge detection algorithm. In the contrast algorithm, if a pixel is more than the threshold, it becomes white and if less, black. The edge detection algorithm finds the edge of the white figure. The edge of each independent hand or finger is then white and the rest of the image black as seen in Fig. 2.

6.2. Analysis

After the image treatment process, the algorithm looks for the biggest entry section. That is, the longest white section in the image border, calculating its center and size. This center becomes the *entry point* parameter and the size, the *entry size* parameter as shown in Fig. 2. This means that only white sections that touch the image border are analyzed.

The algorithm then follows the trace of the edge of the hand so that it forms a single, closed line of continuous pixels, pruning any deviations (caused by hair and other noise

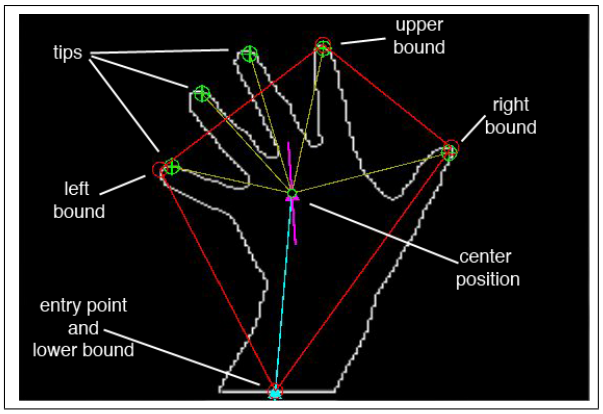

Figure 2. Results of the analysis window.

elements). The pixel coordinates of this line are stored in arrays. The number of pixels in these arrays is the *perimeter* parameter.

The *center position* is the average of all pixels in the *perimeter* and is shown in Fig. 2 . An estimate of the general direction of the hand is also calculated based on the position of the central third of points in the perimeter.

The perimeter arrays are then sampled every *n* pixels to obtain partial pixel coordinates, shown in Fig. 3 as squares on the perimeter. Each successive pair of sampled pixels forms a vector with a direction. As shown in Fig. 3 the change in direction of successive vectors determines an angle that represents the curvature of the perimeter (angle array). We deduce that fingertips will create positive peaks in the angle array and finger valleys will create negative peaks. These positive and negative peaks become the *tip* and *valley* parameters. Tips are shown in Fig. 2 as pointers at the ends of fingers.

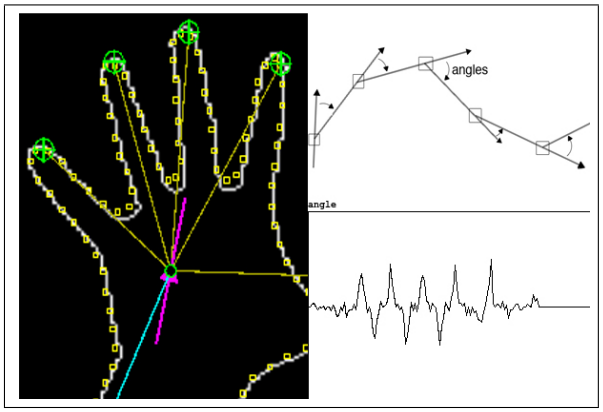

Figure 3. Sampling the perimeter, calculating an angle profile and from it, finger tips and valleys.

Tips are presented both as vectors from the center position to the tip or as an independent location. The first method is dependent on the center position while the latter is independent form it.

6.3. Further Learning

The image analysis section is iterative and so it can find several hands, as shown in the first frame of Fig. 4 and Fig. 5. In Fig. 4, it is clear that we are seeing four fingers and not one hand. It is necessary to distinguish fingers from fingertips. The "learning" portion of the algorithm is concerned with classifying a white mass that enters the frame into fingers and hands.

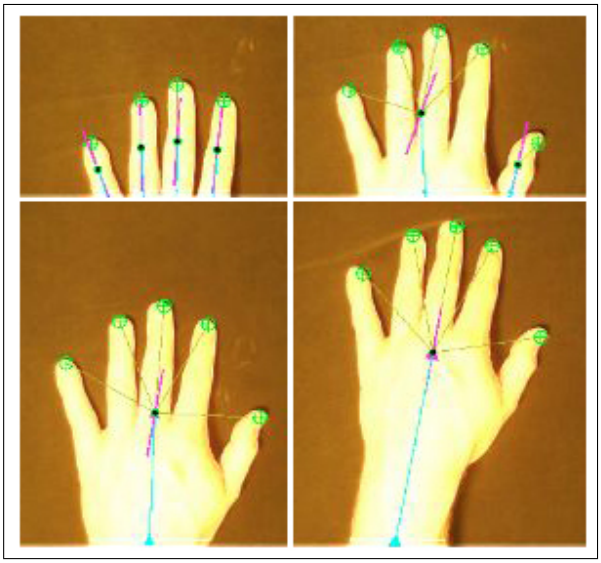

Figure 4. The algorithm can find several hands or fingers.

As we can see in the next frame of Fig. 4, as the hand continues to enter the frame, the four fingers that were detected independently now form part of the same shape. This bigger shape is now classified as a hand and the thumb, as a separate shape, is classified as a finger.

Other tasks of the "learning" section are to find the best tracks for parameters so that multiple hands/fingers and fingertips are treated independently and continuously.

Several discrete features can be extracted from continuous data. When a mass appears or disappears we obtain triggers (onsets and releases). When dealing with continuous parameters changes in direction can also be detected.

6.4. Other considerations

As shown in Fig. 5, hands and fingers can enter from each of the four sides of the frame. This allows the performer to access different mappings depending on how they enter the frame.

As a multi-hand/finger controller, parameters between multiple elements can be calculated. For example, a particular mapping might be assigned to a set of parameter differ-

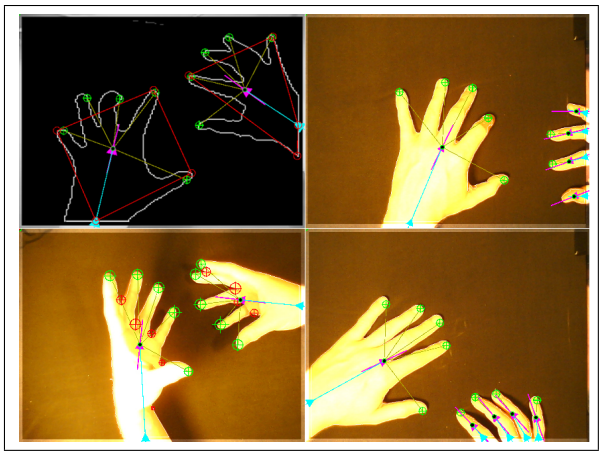

Figure 5. Analysis windows showing hands and fingers entering from several sides of the frame

ences between a hand coming in from below and one from the side.

7. TRACKING PRINCIPLES

Throughout the work in the Silent Percussion Project, I have developed a series of principles that guide the design of video-based controllers. Some of these principles are related to theories of embodiment as developed in cognitive science by Gibson [2], Varela [8], Noe [4] and Rowlands [7].

Silent Interface. Most controllers, particularly percussion pads and keyboards, make audible noises when they are played. These noises interfere with the sound they control. For this reason controllers need to be as silent as possible.

Signal vs. Trigger: Discrete from Continuous. Our movements in the world are continuous. Multiple parameters are extracted from the analysis of one video signal, obtaining streams of continuous data and avoiding the trigger philosophy imposed by the keyboard paradigm in the same spirit as [9]. Instead, discrete variables are features that are extracted from the continuous streams of data obtained in the image analysis. These discrete variables can be used to control transitions through a score, mapping changes or triggering.

Bounded Space. A physical object with clear boundaries allows the performer to start and stop interacting; to detach from it and stop producing data.

Parameter Hierarchy and Interdependence. Parameter design follows an effective hierarchical logic: there are no fingers without a hand, no hand without an arm, no arm without a body. A central aspect in the design is the interdependence of variables. Changes in one variable generally imply changes in all other variables.

This last point is particularly important. Instead of searching for complex mappings, the MANO controller offers complex inputs. As opposed to a set of independent sliders or knobs, we obtain interdependent variables. From this derives the need for the performer to learn how these variables behave, through a kind of *babbling* [9] and practice towards expert performance.

8. CONCLUSIONS

The MANO controller offers a new approach to hand-tracking with video. The performer has a high degree of continuous control over sounds, and is able to change mappings with simple fast gestures. The controller could serve as a complement to other controllers such as D. Wessel's Slabs controller. It could also be complemented with pedals. As computers improve in speed and parallel processing it should allow for higher frame rates reducing latency and jitter.

9. REFERENCES

[1] M. Danks, "Real-time image and video processing in GEM," in *Proceedings of the International Computer Music Conference*, 1997.

[2] J. Gibson, *The senses considered as perceptual systems*. Houghton Mifflin, 1966.

[3] S. Jordá, M. Kaltenbrunner, G. Geiger, and R. Bencina, "The reactable," in *Proceedings of the International Computer Music Conference*, 2005.

[4] A. Noë, *Action in perception*. The MIT Press, 2005.

[5] J. Oliver and M. Jenkins, "The Silent Drum Controller," in *Proceedings of the International Computer Music Conference*, 2008.

[6] M. Puckette, "Pure Data: another integrated computer music environment," *Proceedings of the Second Intercollege Computer Music Concerts*, 1996.

[7] M. Rowlands, *Body language: representation in action*. The MIT Press, 2006.

[8] F. Varela, E. Thompson, and E. Rosch, *The embodied mind*. The MIT Press, 1992.

[9] D. Wessel, "An enactive approach to computer music performance," *Le Feedback dans la Creation Musicale*, 2006.

[10] D. Wessel, R. Avizienis, A. Freed, and M. Wright, "A force sensitive multi-touch array supporting multiple 2-d musical control structures," in *Proceedings of the International conference on New Interfaces for Musical Expression*, 2007.

[11] F. Wilson, *The hand: How its use shapes the brain, language, and human culture*. Vintage, 1999.