Elegant, natural motion of robots: lessons from an artist

Aleksandar Zivanovic[1]

Abstract. This paper examines the use of the control system used by the artist Edward Ihnatowicz (1926–1988) in his sculpture *The Senster* (1970). The limitation of the computer technology of the time led to the use of a digital-analogue hybrid system, where analogue circuits were used to modify the output of the computer to generate smooth motion. The artist used his aesthetic judgement to choose the particular characteristics of the response. This paper shows that the response resembles natural movement (e.g. the movement of the human arm). It goes on to present an algorithm developed by the author to achieve a similar outcome using micro-controllers, with a low computation and memory requirement. It is hoped that this would be of use in the development of robots to interact with humans, as this kind of movement appears to be more attractive than conventional motion control techniques used in robots.

1 A BRIEF BIOGRAPHY

Edward Ihnatowicz [1][2][3][4] was born in Poland in 1926, left at the outbreak of war in 1939 and eventually arrived in Britain in 1943. He studied sculpture at the Ruskin School of Art in Oxford from 1945 to 1949 but also had wide-ranging interests including photography, film-making and electronics. He worked as a photographer and a junior partner in a small furniture company until, in 1962, he left the business and his home to live in a garage and return to making art. During this period he developed "Sound Activated Mobile" (SAM) [5], which was exhibited at the Cybernetic Serendipity exhibition in 1968 and later toured the United States of America, ending at the Exploratorium in San Fransisco. He then started working on his greatest work, "The Senster" which was exhibited in 1970 at the "Evoluon," Philip's newly-opened exhibition centre in Eindhoven, the Netherlands. By that time, he had established a close relationship with a number of people in the Department of Mechanical Engineering at University College London (UCL) and was appointed to work as a research assistant there. He worked on a number of research projects and produced one further work of robotic sculpture, called "The Bandit." He eventually left UCL in 1986 to set up his own company mainly involved with computer graphics. He died in 1988.

Photographs, sketches and videos of his work, together with unpublished articles by Ihnatowicz are available on the Senster website [6]. His family retain an archive of his papers and SAM survives in their custody.

The remains of The Senster were acquired in 2017 by the AGH University of Science and Technology in Krakow, Poland and restored by the "Senster 2.0" project team, led by Anna Olszewska[7][8].

[1] Middlesex University, UK, email: a.zivanovic@mdx.ac.uk

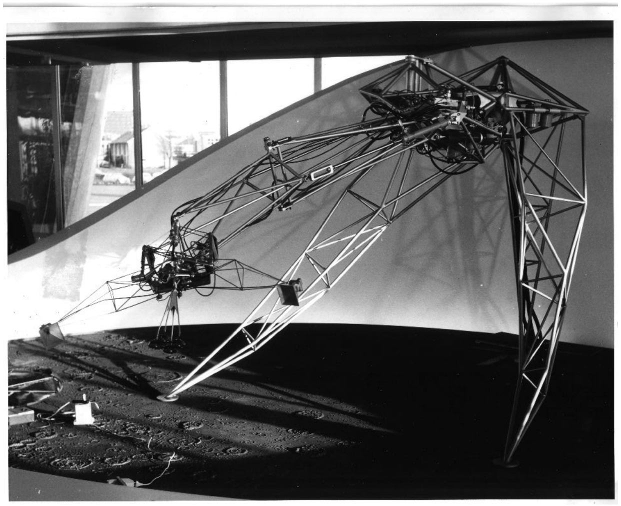

Figure 1. The Senster at the Evoluon in Eindhoven, the Netherlands in about 1970. Photograph by Edward Ihnatowicz.

2 THE SENSTER

The Senster (see Figure 1) was developed for Philips' technology showcase, the Evoluon, in Eindhoven, the Netherlands and went on display to the general public in 1970. Ihnatowicz was helped by engineers at University College London (UCL), Philips and Mullard.

The Senster was large: 15 feet (4.6m) long and 8 feet (2.4m) tall at the shoulder. It was made of welded steel tubes, with no attempt to disguise its mechanical features. There were six joints along the arm, actuated by powerful, quick and quiet hydraulic rams. Two more custom-made hydraulic actuators were mounted on the head to move the microphone array. The microphones were arranged in vertical and horizontal pairs and sound localisation was carried out in software by a process of cross-correlating the inputs on each pair of microphones. The actuators in the head moved the microphones very quickly in the calculated direction of the sound, in a movement reminiscent of an animal flicking its head. The rest of the body would then follow, making the whole structure appear to home-in on the sound if it persisted. Loud noises would cause the body to move upwards and sideways given the appearance of it shying away from the source of the noise. In addition, two Doppler radar units were mounted on the head of the robot, which could detect the motion of the visitors. Low level movements, such as waving or clapping hands, would cause the structure to move towards the source of the movements. Large, or violent movements made it move away, giving the impression that The Senster was frightened.

3 TRAJECTORY GENERATION USED IN THE SENSTER

Fortunately, Ihnatowicz's family have kept an archive of his papers with technical specifications and schematics of the control system. The following description was derived from studying this material.

The computer used to control The Senster was a Philips P9201 with 8k core memory, which used punched paper tape to load the program. It was very similar to the more common Honeywell 16 series. An assembly code program listing exists. Several racks of custom electronics interfaced the computer to The Senster and it is fortunate that most of the circuit diagrams survive.

There were eight hydraulic actuators in total (including the two in the head) and they were controlled in pairs, so, essentially, there was one standard output circuit repeated four times. The following description is for one such circuit.

The output from the computer was latched as 16 data bits. The 16 bits were split into two sets of 5 bits, which represented the next required position for an actuator, thus each joint had 32 possible discrete positions. Each set of five bits was passed to a digital to analogue converter and thence to a circuit Ihnatowicz called *the predictor*. The remaining 6 bits were used by *the acceleration splitter* circuit described below.

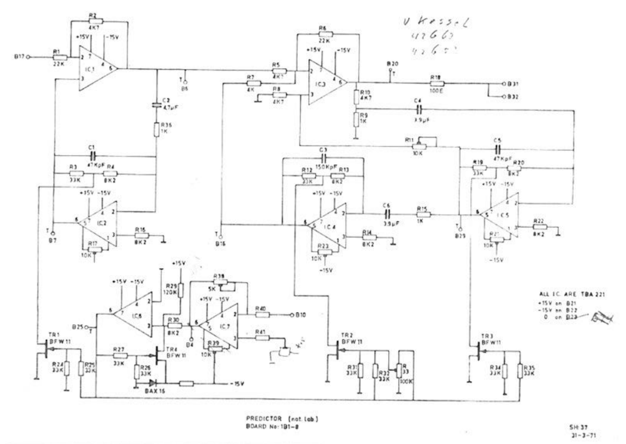

Figure 2. The Predictor circuit. From papers kept by Ihnatowicz's family.

The predictor (see Figure 2) was a second-order low-pass filter, with an adjustable roll-off frequency set by a circuit called *the acceleration splitter*, fed by three spare bits from the latch, via another digital to analogue converter. This circuit distributed an analogue voltage, with a resolution of 8 values, to the predictor circuits, which altered their roll-off frequencies. It effectively set the time by which all the joints had to reach the next set positions, so that they all arrived at the same time. There were two separate acceleration splitters: one for the hydraulics which moved the microphones and another for the joints in the rest of the structure, thus the microphones could flick quickly, while the main structure moved at a more sedate pace.

The predictor filtered the analogue voltage output so that it followed a smooth curve. The computer was not fast or powerful enough to do this in real-time, hence the use of analogue circuits. The output from the predictor circuit was fed to a closed-loop hydraulic servo system, so that the actuators followed the analogue voltage in a proportional way.

Fortunately, the circuit diagram for the predictor survives and was simulated using SPICE, a standard circuit simulation software package. Figure 3 shows the effect of the circuit. At time = 1s, the output from the computer (via a digital to analogue converter) undergoes a step change from 0 to 10V. The predictor filters out the high frequency components, so that the robot starts and stops smoothly. The different curves illustrate the effect of changing the value output by the acceleration splitter.

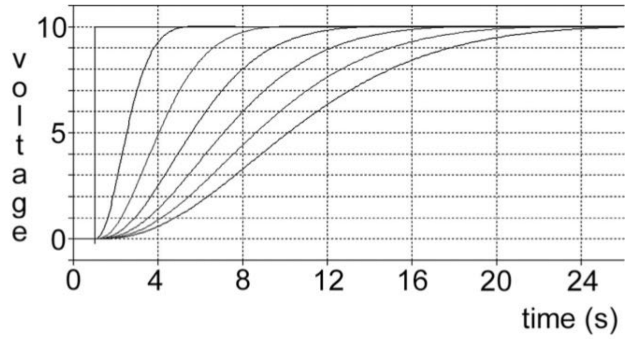

Figure 3. Predictor output for different values output by the Accelerator (position is proportional to voltage)

The derivative of one of these curves is shown in Figure 4a. Figure 4b is a graph of normalized velocity against normalized time of a tracked human arm[9]. It can be seen that shape of the velocity profiles match quite well, so The Senster moved with similar characteristics as biological motion (specifically, a human arm).

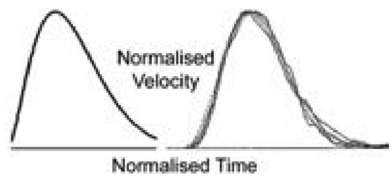

Figure 4. a: Velocity profile from Predictor circuit; b: Velocity profile of human movement (from [9])

4 DIGITAL FILTER IMPLEMENTATION

4.1 Selection of digital filter

The implementation of smoothing Ihnatowicz chose for The Senster was a clever approach to overcome the weaknesses of the computer technology of the time. However, now it is much easier to implement digital filters rather than use analogue circuitry in that way. There are a wide range of standard filters described in the field of Digital Signal Processing (DSP). See, for instance [10]. Exponential smoothing was chosen as is it easy to implement and uses very little computational resources.

4.2 Exponential Smoothing

Exponential smoothing is a technique for smoothing time series data using the exponential window function. The output is the weighted average of the current input value T and the previous smoothed value

$S^{(1)}_{k-1}$. If the time series starts at $k = 0$, the simplest form of exponential smoothing is given by:

$$S^{(1)}_0 = T_0 \quad (1)$$
$$S^{(1)}_k = \alpha T_k + (1-\alpha)S^{(1)}_{k-1} \quad (2)$$

where α is the *smoothing factor* and $0 < \alpha < 1$. In this application, where the technique is used to smooth the movement of a robot joint, values of α close to one will ensure that the joint will reach its target angle quicker than when low values of α are used.

Note that the (1) superscript notation is used to indicate single exponential smoothing - the technique of double and triple exponential smoothing is introduced below.

Exponential smoothing has many advantages over other techniques. It produces an output as soon as two data points are available (c.f moving average filters). It will not overshoot. It takes the same length of time for the joint to reach its target, no matter the size of the movement, so that if multiple motors (e.g. in a robot arm) are being controlled all the joints will arrive at their commanded position at the same time. The *time constant* is the amount of time for the smoothed output to reach $1 - 1/e \approx 63.2\,\%$ of the original signal. The relationship between this time constant, τ, and the smoothing factor, α, is given by:

$$\alpha = 1 - e^{\frac{-\Delta T}{\tau}} \quad (3)$$

where ΔT is the sampling time interval.

A key advantage of exponential smoothing is that it is very simple to implement, with a very low processor and memory requirement. It can easily run on micro-controllers (e.g. Arduino) and imposes a low overhead.

A straightforward implementation in C of an exponential filter is:

```
S1 = (a * T) + (b * S1p);
S1p = S1;
motor_position = S1;
```

Where `T` is the target value of the particular joint. `a` is the smoothing factor α, `b` is $(1-\alpha)$, `S1` is the smoothed value and `S1p` is the smoothed output from the previous iteration of the code.

This code is run in a loop, or using an interrupt handler, so that it is regularly updated at an appropriate frequency (e.g. 50Hz) so that the step changes in position are not noticeable. Each iteration requires two multiplication and one addition operation, and only the previous value needs to be stored in a variable. The initial value for S1p is something of an issue. It makes sense for the robot system to read the real value of the joint and use this value for S1p, when the robot is first switched on.

Exponential smoothing is equivalent to applying a first-order Infinite Impulse Response (IIR) filter as used in digital signal processing (DSP). The advantage with using the smoothing approach is that it is so simple. There is no need for expertise in DSP. The smoothing factor α and the sampling frequency are all that needs to be set and they can be determined in an empirical way.

As the name suggests, exponential smoothing produces an exponential output from a step change input. At the start of a step change in the input, the smoothed output changes quite suddenly. This sudden change can, itself, be smoothed by running the smoothing algorithm on the already smoothed output. This is called double exponential smoothing. The process can be repeated again, to get triple exponential smoothing:

$$S^{(1)}_0 = S^{(2)}_0 = S^{(3)}_0 = T_0 \quad (4)$$
$$S^{(1)}_k = \alpha T_k + (1-\alpha)S^{(1)}_{k-1} \quad (5)$$
$$S^{(2)}_k = \alpha S^{(1)}_k + (1-\alpha)S^{(2)}_{k-1} \quad (6)$$
$$S^{(3)}_k = \alpha S^{(2)}_k + (1-\alpha)S^{(3)}_{k-1} \quad (7)$$

A simple implementation in C is:

```
S1 = (a * T) + (b * S1p);
S2 = (a * S1) + (b * S2p);
S3 = (a * S2) + (b * S3p);
S1p = S1;
S2p = S2;
S3p = S3;
motor_position = S3;
```

Table 1 shows the output of this algorithm for the first 29 steps time steps, for $\alpha = 0.3$ and a step change in T from 0 to 100 at $k = 1$, and Figure 5 plots the results.

Table 1. First 29 steps of the algorithm, for $\alpha = 0.3$ and a step change in T from 0 to 100 at $k = 1$

k	T	S1	S2	S3
0	0	0.00	0.00	0.00
1	100	30.00	9.00	2.70
2	100	51.00	21.60	8.37
3	100	65.70	34.83	16.31
4	100	75.99	47.18	25.57
5	100	83.19	57.98	35.29
6	100	88.24	67.06	44.82
7	100	91.76	74.47	53.72
8	100	94.24	80.40	61.72
9	100	95.96	85.07	68.73
10	100	97.18	88.70	74.72
11	100	98.02	91.50	79.75
12	100	98.62	93.63	83.92
13	100	99.03	95.25	87.32
14	100	99.32	96.47	90.06
15	100	99.53	97.39	92.26
16	100	99.67	98.07	94.00
17	100	99.77	98.58	95.38
18	100	99.84	98.96	96.45
19	100	99.89	99.24	97.29
20	100	99.92	99.44	97.93
21	100	99.94	99.59	98.43
22	100	99.96	99.70	98.81
23	100	99.97	99.78	99.10
24	100	99.98	99.84	99.33
25	100	99.99	99.89	99.49
26	100	99.99	99.92	99.62
27	100	99.99	99.94	99.72
28	100	100.00	99.96	99.79

Figure 6 shows that it takes the same amount of time to reach the target destination, no matter how big the step change in the input. This means that if multiple joints of a robot are being controlled, if the same α if used, they will arrive at their destinations at the same time.

To examine the velocity profile, the difference between each output value and the previous value is plotted in Figure 7. Only S3 is shown because it is of the same order as the analogue filter used in The Senster. Indeed, the graph very closely matches the simulation of the circuit. If the algorithm is repeatedly applied to get S4, S5, etc. the effect on the graph is to keep the same general shape, but to add more of a curve to the initial rise.

The velocity profile exhibits many of the characteristics of natural motion: smoothness and asymmetry, and that it compares well with

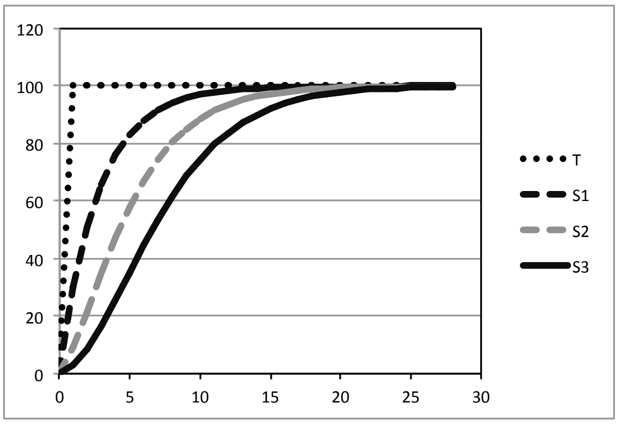

Figure 5. A plot of the input T, and S1, S2 and the output, S3, for $\alpha = 0.3$ and a step change of T from 0 to 100 at $k = 1$. X axis are in units of time steps, Y axis are appropriate position units

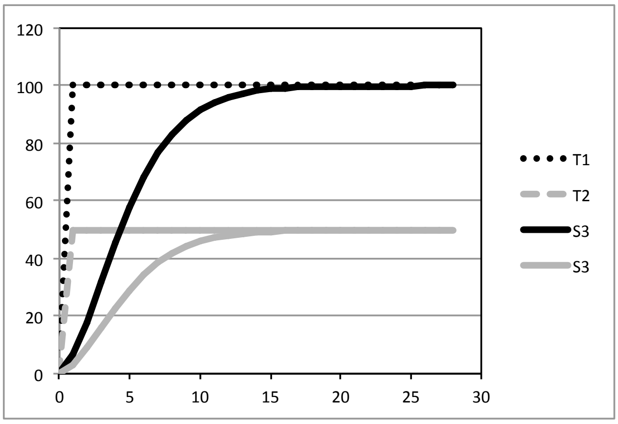

Figure 6. A plot of the input T, and S3, for $\alpha = 0.3$ and a step change of T from 0 to 100 at $k = 1$ and a step change of T from 0 to 50 at $k = 1$. X axis are in units of time steps, Y axis are appropriate position units

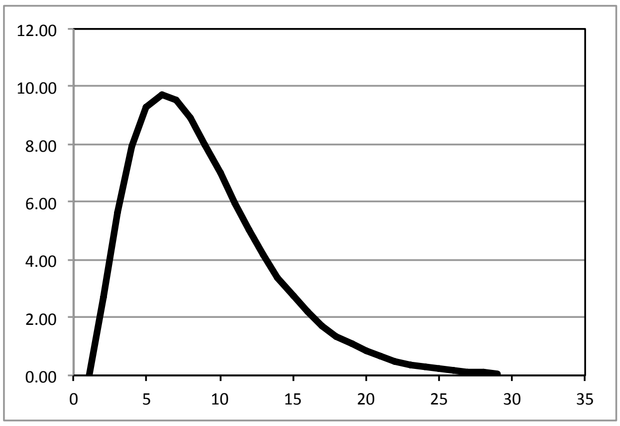

Figure 7. Velocity profile of S3. X axis are in units of time steps, Y axis are the change in position per time step

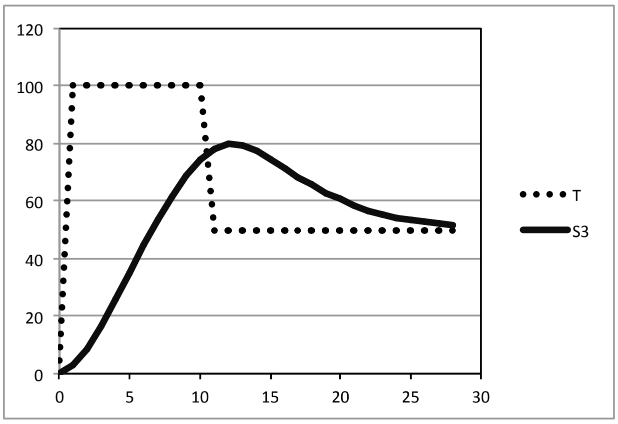

Figure 8. S3, for $\alpha = 0.3$ and a step change of T from 0 to 100 at $k = 1$ followed by a step down to 50 at $k = 11$. X axis are in units of time steps, Y axis are appropriate position units

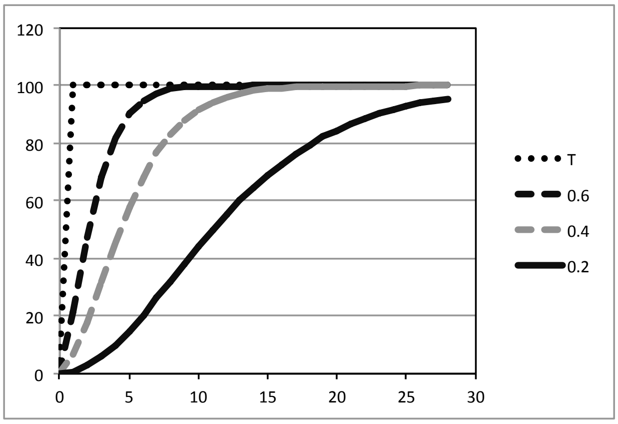

Figure 9. A plot of S3, for $\alpha = 0.6, \alpha = 0.4, \alpha = 0.2$ and a step change of T from 0 to 100 at $k = 1$. X axis are in units of time steps, Y axis are appropriate position units

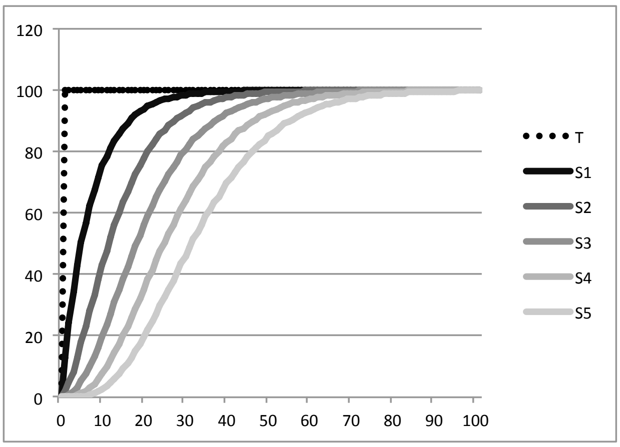

Figure 10. Repeated exponential smoothing

both The Senster's and the human arm velocity profile shown in Figure 4. It should be noted, however, that this algorithm does not aim to simulate biological movement, but to simulate the movement of The Senster. Ihnatowicz was not deliberately trying to simulate animal motion: he states in his papers that he was aiming to achieve a *pleasing* movement.

Figure 8 shows the response of the algorithm to a change in the input before the output has a chance to settle. In this case the input value changes from 100 to 50 at time step 11. It can be seen that the algorithm tracks the change smoothly.

Figure 9 shows the response of the algorithm to a variety of values of α. It demonstrates that the choice of the value of α sets the speed the joint moves to its destination.

Figure 10 shows the result of repeated exponential smoothing, up to S5. It is clear that the output becomes smoother, especially at the beginning of the curve. However, significant lag is introduced. Simulations have shown that there is not much to be gained by applying smoothing more than three times.

5 CONCLUSION

Some initial work has been carried out to explore the subjective impression this style of movement has on observers[11]. In this study, a robot arm was programmed to carry out three gestures: a simple point-to-point motion, a waving action and a bowing action. The robot was controlled using an algorithm very similar to the one described in this paper and the acceleration was varied from low to high. Observers were asked to rate the emotional content of the movement using Russells, the Tellegen-Watson-Clark and the PAD models for measuring emotions. The results showed that people were prepared to ascribe emotions to the movements, with most ascribing sadness, unhappiness or tiredness to low acceleration; happiness, pleasure or calmness to medium acceleration and excitement, alertness, arousal or surprise to high acceleration movements. Observers commented that the movement seemed "natural" and not "robotic".

Research which started as an investigation into the details of how Edward Ihnatowicz's Senster worked has led to the development of a simple method of generating smooth, natural movement for multi-joint robots.

REFERENCES

[1] R. Ihnatowicz, *Forty is a Dangerous Age: A Memoir of Edward Ihnatowicz*, 111–118, White Heat Cold Logic: British Computer Art 1960-1980, MIT Press, Cambridge, Massachusetts, 2008.

[2] C. Mason, *A Computer in the Art Room*, JJG Publishing, Hindrigham, Norfolk, UK, 2008.

[3] A. Zivanovic, *The Technologies of Edward Ihnatowicz*, 95–110, White Heat Cold Logic: British Computer Art 1960-1980, MIT Press, Cambridge, Massachusetts, 2008.

[4] J. Walewska, *Relationship of art and technology: Edward Ihnatowiczs philosophical investigation on the problem of perception*, 172–177, Re:live Media Art Histories Conference 2009

[5] A. Zivanovic, E. Ihnatowicz *Sound Activated Mobile (SAM) at Cybernetic Serendipity*, Symposium on Cybernetic Serendipity Reimagined, In conjunction with the 2018 Convention of the Society for the Study of Artificial Intelligence and Simulation of Behaviour (AISB 2018), Liverpool, UK, 6th April 2018

[6] A. Zivanovic, *www.senster.com*

[7] A. Olszewska *Senster 2.0: cultural context of media art curatorship*, 64–69, Proceedings of the Conference on Electronic Visualisation and the Arts, London, United Kingdom, 2018

[8] A. Olszewska *senster.agh.edu.pl*

[9] C. Atkeson, J. Hollerbach *Kinematic features of unrestrained vertical arm movements*, 5(9), 2318–30, Journal of Neuroscience, Sept 1985

[10] S. Smith *The Scientist and Engineer's Guide to Digital Signal Processing*, California Technical Pub, 1997. Available online at *http://www.dspguide.com*

[11] S. Sial, A. Zivanovic *Communicating simulated emotional states of robots by expressive movements* International Conference on Social Robotics, Workshop 2: Embodied Communication of Goals and Intentions, 27-29 Oct 2013, Bristol, UK.