

# Der Gestus des Fließens

## The Gestures of Flowing

Prof. Herbert Traub  
Tim Blechmann  
Thomas Rühr

September 23, 2004

### Abstract

The project “Der Gestus des Fließens” is a space installation by Professor Herbert Traub. Sensor information emerging in real-time from both the audience in the exhibition place and ambient data is used control animated sculptures, video and audio. The sensor inputs are being processed by a Pure Data patch to specify the state and movements of the installation.

## 1 Introduction

### 1.1 The Project

“Der Gestus des Fließens” is a project led by Professor Herbert Traub the Director of the “Institute of Fine Arts and Media 2” (IDG2), a division of the faculty of architecture at the University of Stuttgart. The “Institute for Image Understanding”, headed by Professor Paul Levy, a division of the “Institute of Parallel and Distributed High Performance Systems (IPVR)” at the faculty of informatics at the University of Stuttgart, is supporting the project in the aspects of image recognition in terms of expertise and development resources. Dr. Andreas Mahling, lecturer at the Studio for Electronic Music at the Music Conservatory, had been pointing to use Pure Data<sup>1</sup> as control software for the installation and designed and implemented the first control patches and software architecture<sup>2</sup>.

Thomas Rühr is the scientific assistant of Professor Levy at the IPVR and working on the implementation of the image recognition software.

Tim Blechmann is currently employed at the IDG2 as scientific assistant to redesign the software architecture and implement the control software capable of controlling the space installation.

### 1.2 The Installation

The ambition of the project “Der Gestus des Fließens” is to create an interactive space installation with the interdisciplinary approach to combine both fine arts and electronic audio-visual media. The main aspect of it will be to create an interaction between the audience and the machine so the audience will be both observer and creator of a specific moment of the installation. Representing water as shape-shifting medium is planned for all actions and audio-visual media. The software is supposed to use a model developed in artificial intelligence systems to achieve interaction.

---

<sup>1</sup>Pure Data was written by Miller Puckette and others:  
<http://crca.ucsd.edu/~msp/software.html>  
<http://www.puredata.info>

<sup>2</sup>He presented his work at the ICMC 2003 in Singapore under the title: The Gestures Of Flowing, Using PureData as a Backbone for Interactive Sculpture Animation, Video and Sound

## 2 Hardware

The hardware of the installation “Gestus des Fließens” consists of 3 frames, one center frame and two side frames. The main frame is equipped with 4 movable surfaces, 1 transparent water tank, 4 lights and 2 speakers, the side frames are each equipped with 2 movable surfaces, 1 transparent water tank, 3 lights and 1 speaker. Between the main and the side frames, 2 towers are placed, each containing 2 monitors for video projections. Figure 1 shows a 3d model of the installation.

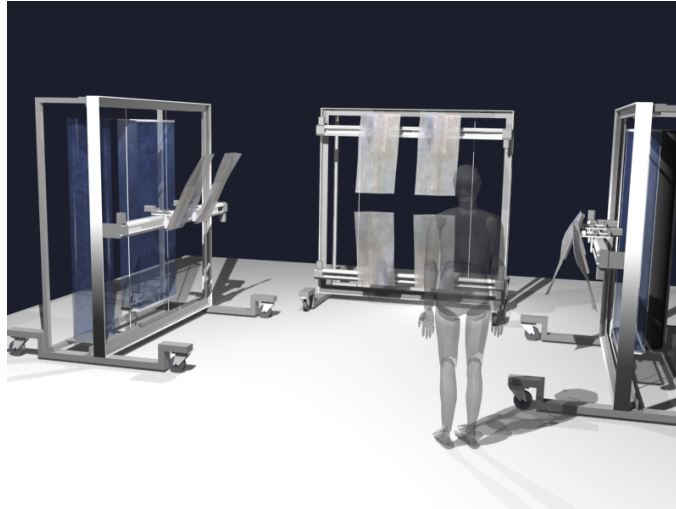


Figure 1: 3d model of the installation

### 2.1 Water & Air

Each frame has one visible water tank (glass) in the center and another hidden one at the back. It is possible to pump water from the visible and the hidden tank to each other and thus raise or lower the surface of the water. On the main frame it’s possible to fill in the water from one valve at the top and from a circular sprinkler in the middle of the tank. The tanks on the secondary frames are only equipped with one valve on the top and one on the bottom. So it’s possible to fill all visible tanks from 2 different valves.

All the visible water tanks are connected to a compressor, that allows it to blow compressed air of about 2 bar through a valve at the bottom of each tank.

### 2.2 Movable Surfaces

Each frame is equipped with movable surfaces, that are made of epoxy resin. The center frame has 2 on the top of the water tank and 2 on the bottom, the side frames each have 2 surfaces in front of the tank. The surfaces can be moved and tilted along the x-axis with controllable step motors.

### 2.3 Light

The movable surfaces on the main frame can be lit from behind with different lights. The upper left with red, the upper right with blue and the bottom ones with yellow light. The intensity of the light can be smoothly adjusted.

On the side frames the lighting system is not as elaborated as on the center frame, since the transformers can’t be controlled by continuous values, but only switched on or of.

## 2.4 Visuals & Sound

The 2 towers each have 2 video screens. With these screens, and if the room permits it, with an additional video beamer, it is possible to show short movie clips. And with the 4 speakers, installed on the frames, it's possible to play back sound clips from a sample library.

## 2.5 Hardware Interface

To control the lights, the valves, the pumps and the motors for the movable surfaces memory programmable control units (SPS) have been used. The control computer is equipped with two interface cards that provide 16 bidirectional channels for controlling the motors, and several analog and digital outputs for valves, pumps and lights. A library installed on the control computer to access the SPS can be directly accessed from Pure Data using a wrapper dll.

## 2.6 “Eyes & Ears”

On the two towers video cameras have been installed to observe the audience. These video cameras, basic Logitech web-cams since we didn't have the possibility to install high quality firewire cameras, have been connected to the computers inside the towers.

Additionally there are several microphones installed on both the towers and the frames that are connected to a Midiman Delta 1010 audio interface. Additional microphones can be put up, wherever the exhibition place permits.

Also it is possible to put panels with contact microphones on the floor to get position triggers on a few areas such as the entrance door.

## 2.7 Computer System

At the moment, the computer system controlling the installation is based on 4 computers, one for the audio IO, equipped with the Midiman Delta 1010, two for the video IO, each connected to 2 video cameras and two video screens (if a video beamer is supposed to be used for an exhibition, it may be connected to one of these computers, too) and one computer controlling the SPS. Except for the control computer that's running Windows 2000, every machine is running Windows XP.

# 3 Software

The whole software side of the installation is realized with Pure Data by Miller Puckette. The control software consists of a number of patches distributed among the 4 computers. Figure 2 shows a design overview on the single patches. The fat connections represent pd internal message connections, the lines represent tcp/ip network connections, the dashed lines are connections that are planned to be implemented in the near future.

## 3.1 Low-Level Control Layer

### 3.1.1 Controlling Hardware

The main aspect of the low-level control patch is controlling the hardware. This is done using the external for Pure Data, we had to write for accessing the SPS<sup>3</sup>. For the step motors it is possible to control the position, the speed, and the acceleration by directly setting the hardware registers of the SPS. The SPS offers more advanced control features like setting a relative position or the time granularity for acceleration / deceleration, that could be set from this layer, too (for the installation we probably won't use these features). The problem of the direct access to the hardware is a very cryptic interface. Therefore the main function of the low-level control layer is to

<sup>3</sup>The external had been implemented by the former scientific assistant Konrad Eisele.

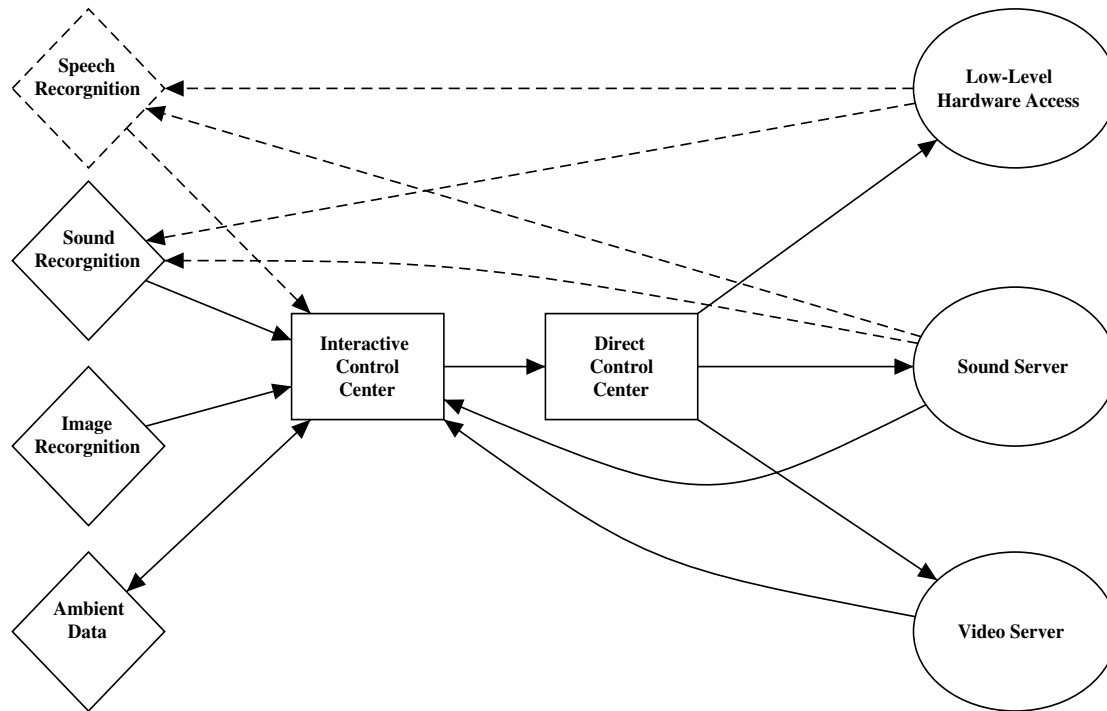


Figure 2: Structure of the control software

provide a level of abstraction between the hardware and the user interface by providing symbolic addresses for hardware registers, normalized value ranges, and hardware initialization.

### 3.1.2 Query state of Hardware

The secondary task of the low-level control layer is to provide access to the state of the hardware. It is possible to read several registers from the SPS, like the actual position or the actual speed. To access the actual position is important, since the collision control hasn't been implemented very sophisticated so it's necessary to check the actual position with the target position and possibly resend the command to change the position .

Another application would be to send the audio recognition layers information about the movements. We can turn of the audio recognition for the time of fast movements, what makes sense, since the sound of the motors would return signals, we don't want to analyze and since it is possible to read the actual frequency of the motors, it might be possible to implement a noise reduction algorithm to filter the signals before the analysis to reduce the interferences we get.

## 3.2 “Direct Control” Layer

The “Direct Control” provides user-access to the controllable parameters of the machine and represents another level of abstraction between the user / the interactive control layer and single parameters of the machine.

### 3.2.1 User Interface

As described above, the access to the SPS is based on setting and reading hardware registers. But it's nearly impossible to create complex scenarios on by directly accessing the hardware registers. The “Direct Control” layer provides a graphical user interface that allows the user to define specific

scenes or directly control the installation.

Figure 3 shows the current state of the GUI. Position and tilt of the surfaces can be adjusted with the sliders and knobs. This interface also provides the user with subroutines to access the pumps and valves. Since the valves have to be switched in certain ways (access to a source and a destination tank) to assure the proper function of the pumps, the modes “off”, “pump in”, “pump out” and “circular pumping” can be selected via a pop-up menu. Also the valves for the compressed air can only be accessed through subroutines. After a defined time the valves will close automatically. The connection between the “Direct Control” layer and the “Low-Level Control”

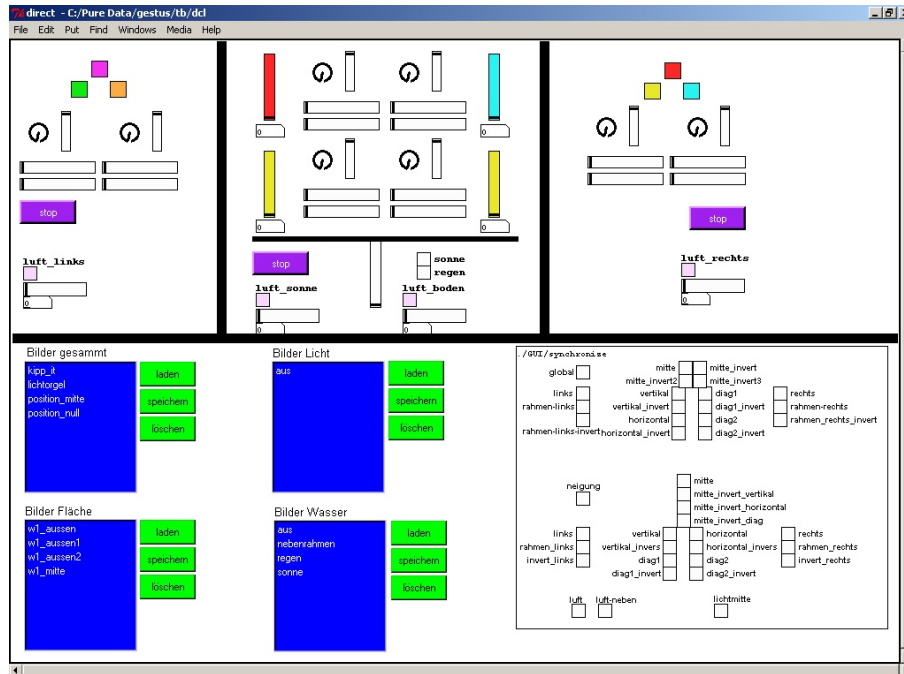


Figure 3: Direct Control Center

layer is realized using pd messages.

### 3.2.2 Synchronization

It is possible to synchronize several parameters. The specific part can be found on the bottom right of Figure 3. This is useful for generating presets, but especially necessary to generate symmetric patterns by mapping real-time parameters to the positions or tilts of certain movable surfaces.

### 3.2.3 Preset system

Every control parameter is directly mapped to a value<sup>4</sup>. The preset system queries the values of all control parameters and saves them into a pool<sup>5</sup> object. These stored scenes can be reloaded either by the symbolic name or an internal ID. There are four instances of the preset system for water, lights, surfaces and for the whole setup of the machine.

### 3.2.4 Markov-Chain based Patterns

Based on the 4 different kinds of scene presets, a support for Markov-Chains has been implemented. In a pool object the transition probabilities and durations for certain edges have been defined.

<sup>4</sup>A value is a symbolic name for a global floating point variable in Pure Data

<sup>5</sup>Pool is a hierarchical storage object for pd and max/msp by Thomas Grill: <http://grrrr.org>

Using an index<sup>6</sup> object it is possible to give symbolic names an integer ID. That way symbolic scene names are converted to IDs that can be interpreted by a prob<sup>7</sup> object. After a new scene has been selected by prob, a lookup to the definition of the edge in the pool is done to determine the duration between the current and the future scene.

### 3.2.5 Media Player

From the “Direct Control” layer, it is also possible to access the visual and sonic media. The media are divided into “Sound”, “Music” and “Video”. Figure 4 shows the user interface to the media

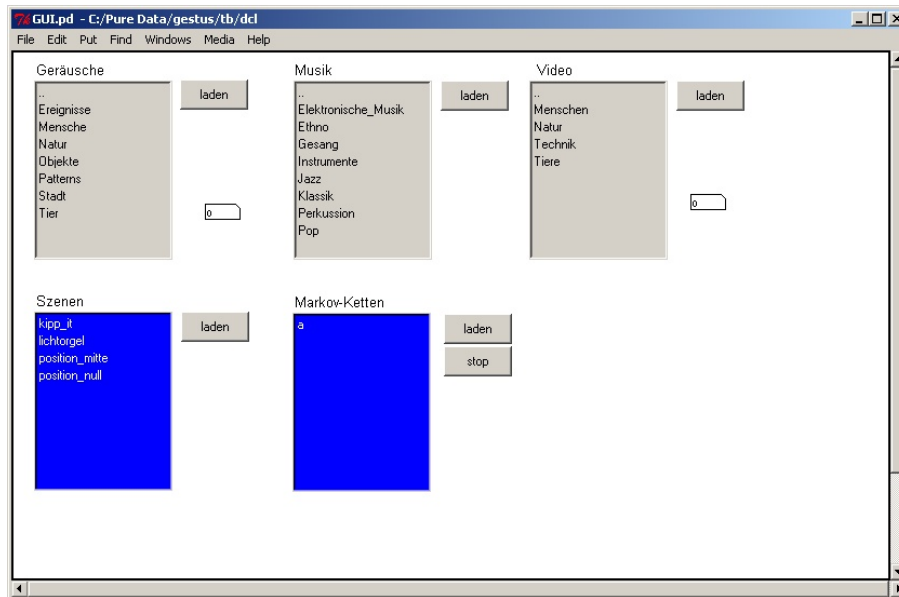


Figure 4: Media file and preset loading interface

files. The media files are being sorted in a hierarchy of categories. If one hierarchy is being loaded, a random media file from this category will be selected and a command will be sent to the video or sound server to load this specific file.

On the Pure Data side the category tree is imported to a pool object via xml. Using an different program called Filesort<sup>8</sup>, it is possible to generate a file hierarchy and export it to an xml file format that pool can load.

### 3.3 Video Server

The video server is a simple PD/GEM<sup>9</sup> patch, running on the two video processing computers. It is working as 3 channel video mixer with alpha-blending. In future real-time video processing will be added. The video server is gets commands from the “Direct Control Layer” which video to load and sends status reports to the interactive control layer.

### 3.4 Audio Server

The audio server running on the audio procession computer, equipped with the Delta 1010. At the current implementation, the audio server is a Pure Data patch with three parallel sound file

<sup>6</sup>Index is part of zexy by Iohannes Zmoelnig: <ftp://iem.kug.ac.at/pd/Externals/ZEXY/>

<sup>7</sup>Prob is a part of cyclone by Krzysztof Czaja: <http://suita.chopin.edu.pl/~czaja/miXed/externs/cyclone.html>

<sup>8</sup>Filesort was written for “Der Gestus des Fließens” by Konrad Eisele

<sup>9</sup>GEM is an external for Pure Data that provides real-time OpenGL-based graphics, including image procession. GEM is currently maintained by Iohannes Zmólnig: <http://gem.iem.at>

players. Each of the three players is accessing all 4 speakers installed on the main and side frames. Future developments will extend the simple sound file player with audio processing capabilities and spatial distribution of the sound to the 4 speakers. At the moment, the sound file player is only able to play back two-channel sound files. The connection to the rest of the installation is similar to the video server.

## 3.5 Interactive Control Layer

### 3.5.1 Collecting Data

Via tcp/ip connections, the data collected by the sensors is sent to the “Interactive Control Layer”. The data of video or sound including a possible speech recognition, is sent directly when events occur, the ambient data will be requested on demand. This data is being stored for further use. In addition to that every change one of these data will be written to a text file, qlist<sup>10</sup> can read. This is done to examine the behavior of the installation after data has been collected during an exhibition.

### 3.5.2 Future: Interactive Machine Model

The ambition of the interactive machine model is to create behavior that mimics biologic rhythms by establishing a system of needs and fulfillment of needs. The machine is able to take several states. In every state the needs of the machine will change with the time.

For example if there are 3 different states, eating, sleeping and playing, the model is that in every single state the machine will both consume and gain certain attributes for example hunger, fatigue or lust. Sensor information such as movement towards a frame, temperature, color etc. will change the gradients of the attributes in the specific states. The attributes themselves will be related to the transition probabilities between the states.

In the example, the machine would stay in the ‘play’-state until the ‘hunger’ exceeds a limit or the lust to play is gone. It would then go to the rest state or the eating state in order to regain lust and energy to ‘play’. The interaction with people would for example partially inhibit the consumption of the ‘lust’-energy, so that the machine stays longer in the ‘play’ state than it would if there were no people. We consider to create different behaviours of the machine for each state it is in, for example the movements, colors and sound the machine produced in the ‘sleep’ state would be gentle and smooth. In the ‘play’ state, the machine would show direct interaction with the people surrounding it.

By using such a system, it is possible to create a nondeterministic interaction that is both reproducible and useful for both statical and dynamic movements.

### 3.5.3 Sending Commands to the “Direct Control” abstraction layer

The data sent to the “Direct Control Layer” will either be commands for controlling the media players or controlling the hardware. The hardware control is based on the loading of presets, or direct access to single parameters including synchronization of specific parameters. The loading of movement patters will be implemented, soon.

## 3.6 Image Recognition<sup>11</sup>

At the moment, the vision is the most sophisticated source of sensor information about the environment of the installation.

In order to gain semantic knowledge, it is not sufficient to process only low level data, but this so called iconic data must be raised to a higher level of understanding by extraction of sub symbolic and symbolic cues. Iconic data in the field of vision are in our manner-of-speaking simple color values of pixels. The subsymbolic level is reached by combining the low level information to clusters

<sup>10</sup>qlist is the pd internal message sequencing object

<sup>11</sup>This part has been written by Thomas Rühr, who is responsible for the Image Recognition

that might be connected with a real object. The semantic level would be in our case information about the entering or leaving of a person into the room of exhibition or the understanding of his or her gestures etc.

The vision is completely implemented as library for Pure Data using OpenCV<sup>12</sup>. The OpenCV implementation for Pure Data includes several objects that are communicating with the message interface, exchanging IDs for the images.

### 3.6.1 Trigger Fields

The first vision sensor that was implemented was a simple trigger. Areas in the video can be defined and the movement in these areas will then be analyzed. If in one of these areas a motion is detected that exceeds a previously defined strength, the triggering action of that field is activated. Although this sensor lives on a very low level, it has its advantages over other possible sensors through its robustness. The activation in our implementation is independent of the overall light situation, light changes do not affect the operation as long as they don't transcend a certain speed like flashlights do. In contrary to other vision features, trigger fields also work in a very dark environment<sup>13</sup>.

Although the trigger fields provide only low level cues, the combination of more of them enables us to establish a higher level of understanding of the scene.

The trigger filter is implemented as a patch including the capture filter and the motion classification filter. The single triggers are implemented as subpatches, each describing a single trigger field for one camera. The trigger signals generated are analyzed in combination to find correlations in time and space. If for example a smooth series of trigger signals occurs, that may imply that there was a certain movement of an observer. The simultaneous observation of a movement from different cameras gives a hint to the position of the movement in space.

Apart from combining trigger signals in time and finding correlations, it is possible to search for overlappings in space. Multiple cameras observe the scene from different perspectives. A single camera can only detect 2d motion, an area in the frame represents a cone in space. We therefore combine the trigger signals to detect spacial correlations between them. With three cameras observing the scene from all 3 directions of space, we are able to get a hint of the true spacial origin of an observed motion.

### 3.6.2 Color Calibration, Classification & Segmentation

We use a color classification filter. The filter is trained online with sample images. For each color class, e.g. skin color, a set of sample colors is learned. Thereafter, the learned color class can be used to segment the image into parts of same color class. We then have a list with connected regions of for example skin color. They can then be used directly to give a hint where people are. As a second step, we consider to use the regions of skin color and run a more reliable face detector filter there.

### 3.6.3 Future: Dense Stereo Vision

We want to employ 3d techniques, in order to understand the movement of the people observing the object. The problem of this approach is the ambiguity in dense scenes where more than one object moves.

Real time Dense Stereo Vision is a relatively new approach in computer vision, since the computing power did not suffice in the past. From two cameras mounted side by side with a certain distance between, two frames of the same scene are captured in the same instant. The algorithms used mimic of the human perception of space. Via a correlation of image elements in the two frames, a disparity is detected, which then serves as the hint to the depth respectively the distance of the

<sup>12</sup>The library has been implemented by Konrad Eisele and Thomas Rühr. Open Computer Vision Library is available at: <http://www.intel.com/research/mrl/research/opencv/>

<sup>13</sup>Although the cameras are very noisy in dark environments, the noise is pretty constant over time.



object seen. As a result, a dense grey level frame is generated which displays the distance for each pixel.

## 3.7 Sound / Speech Recognition

### 3.7.1 Trigger Panels

The most basic use of sound recognition is the use of panels with contact microphones on the floor. If a person steps on a panel, the sound the contact microphone records can easily be analyzed by an envelope follower and send a message trigger to the “Interactive Control Layer”. After one of these triggers have occurred, it is very probably that a person is located near the trigger panel.

### 3.7.2 Sounds of the Exhibition Place, Speech Recognition

A more difficult approach is the analysis of the sounds the audience produces. Since these are more or less sounds of people speaking, it would probably only make sense to search for certain specific sounds like clapping of hands or something similar.

More sense would be implementing a speech recognition library to pd. Open Source libraries for such a task are available<sup>14</sup>. Since the machine itself produces sounds (speakers, pumps, motors, compressed air) this might produce a few problems. But since we know the state of the machine any time, it might be possible to reduce the interference. The basic approach would be to turn the speech recognition off during actions, but it would also be possible to create a sonar fingerprint of the machine’s own sounds and try to remove this sound before analysis using noise reduction algorithms.

## 3.8 Ambient Data

In the time, no visitor is present in the exhibition place, there won’t be any usable information to set the state of the machine. One approach would have been to do nothing at all. On the one hand, that would make sense, since the installation won’t be looked at by anyone. On the other hand, it would be possible to implement a course of a day for the null position.

There are several homepages, providing meteorologic data that are being updated once once or twice per hour. Using the python interface<sup>15</sup> for pd it is easily possible to download an html page and parse the html code to gain the necessary data<sup>16</sup>. Currently the used data are temperature, wind velocity, average rainfall, barometric pressure and humidity. The interactive control center can poll data from this patch, that’s running in a different instance of pd (possibly on another machine).

## 4 Future

In future, it is planned to create a bigger space installation on a public place in Stuttgart. It will be based on the same means and insights gained from current installation consisting of the 3 frames, but in another, bigger shape. The plan is to create a walkable tunnel with walls of transparent water tanks. On the inner surface of the tanks movable panels similar to the movable surfaces of the current installation are supposed to be mounted.

## References

- [1] Professor Herbert J. Traub, Director of the Institute of Fine Arts and Media 2, University of Stuttgart: Der Gestus des Fließens

<sup>14</sup>At the moment, we consider to use CMU Sphinx: <http://www.speech.cs.cmu.edu/sphinx/>

<sup>15</sup>py/pyext was written by Thomas Grill: <http://grrrr.org>

<sup>16</sup>At the moment only one homepage is used to download the meteorologic data. For an exhibition this would probably be extended to be independent from server downtimes.

- [2] Dr. Andreas Mahling, University of Music and Performing Arts Stuttgart: The Gestures of Flowing: Using PureData as a Backbone for Interactive Sculpture Animation, Video and Sound; In Proceedings of the International Computer Music Conference 2003, Singapore