



**DECO – Designing Comfort**  
Institute of Automation

| Faculty of Informatics

**INSO - Industrial Software**  
Vienna University of Technology

**Bachelor Thesis**

# Processing

The new way of visual programming

**Jürgen Leitner**

**Technische Universität Wien  
Vienna University of Technology**



## TABLE OF CONTENTS

<b>1. PROCESSING</b>	<b>3</b>
What is it? What is it for?	3
<b>2. EVALUATION</b>	<b>4</b>
Installation	4
The Processing IDE	4
The Processing API	6
Libraries	6
Community Developed Libraries	7
What is it used for?	11
Visualizations	11
Hardware	15
Video	16
Animation	17
Visual and Interactive Installations	18
Art and Artistic Presentations	19
Image Processing	20
Audio Processing	20
Games	20
<b>3. EXTENSIONS AND REWRITES</b>	<b>22</b>
Mobile Processing	22
Wiring and Arduino	22
Nodebox	23
<b>4. CONCLUSION AND OUTLOOK</b>	<b>23</b>
<b>REFERENCES</b>	<b>25</b>

### ABSTRACT

*This paper is about the programming language and IDE Processing, which evolved out of MIT in 2001. It is based on Java and this paper tries to evaluate what it can be used for. After a short introduction of the IDE the main libraries as well as a lot of community added libraries are listed and their functionality is described. A big list, categorized in groups like Visualization, Hardware, Video, Animation, Installations and so forth, of applications using Processing is given in chapter 2.4 ("What is it used for?"). This paper also tries to list some alternatives and offspring of Processing like Arduino/Wiring as well as similar IDEs for other programming languages (e.g. Python). In the last chapter a conclusion and an outlook is presented, which lists some ideas of what Processing can and hopefully will be used in the future.*

*"The purpose of computing is insight, not numbers."* Richard Hamming

*"Functional visualizations are more than innovative statistical analyses and computational algorithms. They must make sense to the user and require a visual language system that uses colour, shape, line, hierarchy and composition to communicate clearly and appropriately, much like the alphabetic and character-based languages used worldwide between humans."*

Matt Woolman  
Digital Information Graphics

# 1. Processing

## *What is it? What is it for?*

Processing is an open source programming language, toolkit and API based on Java™ and is intended to make programming of visualizations easier. It was initiated by Ben Fry, who is currently teaching at Carnegie Mellon University in Pittsburgh[1], and Casey Reas, currently teaching at UC LA[2]. Processing first emerged in 2001 from the “Aesthetics + Computation Group” at MIT’s Media Laboratory[3], where the two were working together.

The official Processing website [www.processing.org](http://www.processing.org) describes the idea behind it in the following way:

*“Processing is [...] for people who want to program images, animation, and sound. It is used by students, artists, designers, architects, researchers, and hobbyists for learning, prototyping, and production. It is created to teach fundamentals of computer programming within a visual context and to serve as a software sketchbook and professional production tool. Processing is developed by artists and designers as an alternative to proprietary software tools in the same domain.”*

In 2005 Processing 1.0 (BETA) was released under a combination of GPL and LGPL licenses and can therefore be downloaded, distributed and used free of charge. Processing is also open source and is actively developed firstly by the two creators and a core group of programmers who basically maintain and develop the core functions and libraries. Since Processing is not a corporation they are not paid nor employed[4] for these activities. Processing has a focus on having a very minimal core but also an easy way to build extensions into it. Therefore the website also lists lots of libraries and extensions[5] created by a growing community around the Processing language, which consists of not only programmers and computer technicians but also designers, architects and people in marketing[6].

Processing has profited from the research into “Design by Numbers” (DBN) also at the MIT[7], which was an endeavor by Professor John Maeda to show and bring closer the “idea” of computation and programming to designers and artists. DBN is also both a programming language and a programming environment but it has a narrower focus than Processing. It mainly tries to familiarize people with the basic concepts of computational media.

Although Processing first emerged in 2001 no widely known papers have been published and not much scientific work has been done about Processing as a programming language but Processing has appeared in papers for projects that actively use it for visualizations. Processing was never the main focus in these and therefore the evaluation of it fell a little short. In the last years the interest in Processing and also the community around it grew bigger and bigger. In 2005

Ben Fry and Casey Reas were given the prestigious “Golden Nica” award at the “Prix Ars Electronica” in the category “Net vision” for the creation of Processing.[8]

## 2. Evaluation

### *Installation*

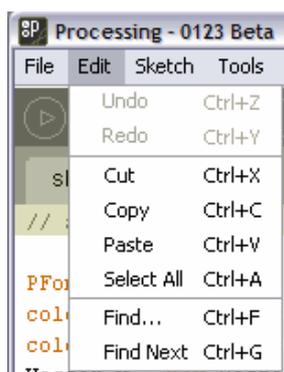
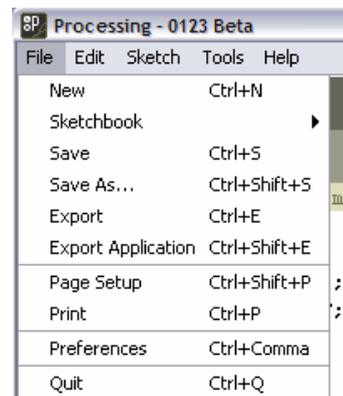
The installation of Processing is pretty easy. For most systems it is enough to just download the respective archive from the Processing download site[9] and unzip it. After this the IDE should be able to start. In the rare case that it does not work, lots of information can be found in the discussion forum[10] of the Processing website. Usually a look through the forum archives brings up all the information needed, if there is no information new question are usually answered really fast.

### *The Processing IDE*

The downloaded and installed package of Processing includes a toolkit for developing Processing applications. This rather minimal IDE is a good starting point to develop ideas and start with a sketch therefore the environment is also called sketchbook in the Processing community.

As you can see in the following screenshot the IDE does include various other functions, like the easy starting and stopping of your application for testing purposes. The toolbar also has buttons for “New Sketch”, “Open Sketch”, “Save Sketch” and “Export Sketch”, which are pretty self explanatory.

The IDE has a menu bar which includes the standard “File” menu with the usual file handling items, as well as a “Sketchbook” submenu and an entry for the properties dialogue called “Preferences”.



The “Sketchbook” submenu includes an item to “Open” a sketch as well as a list of predefined examples that are located in the subfolder “examples” in Processing’s main folder.

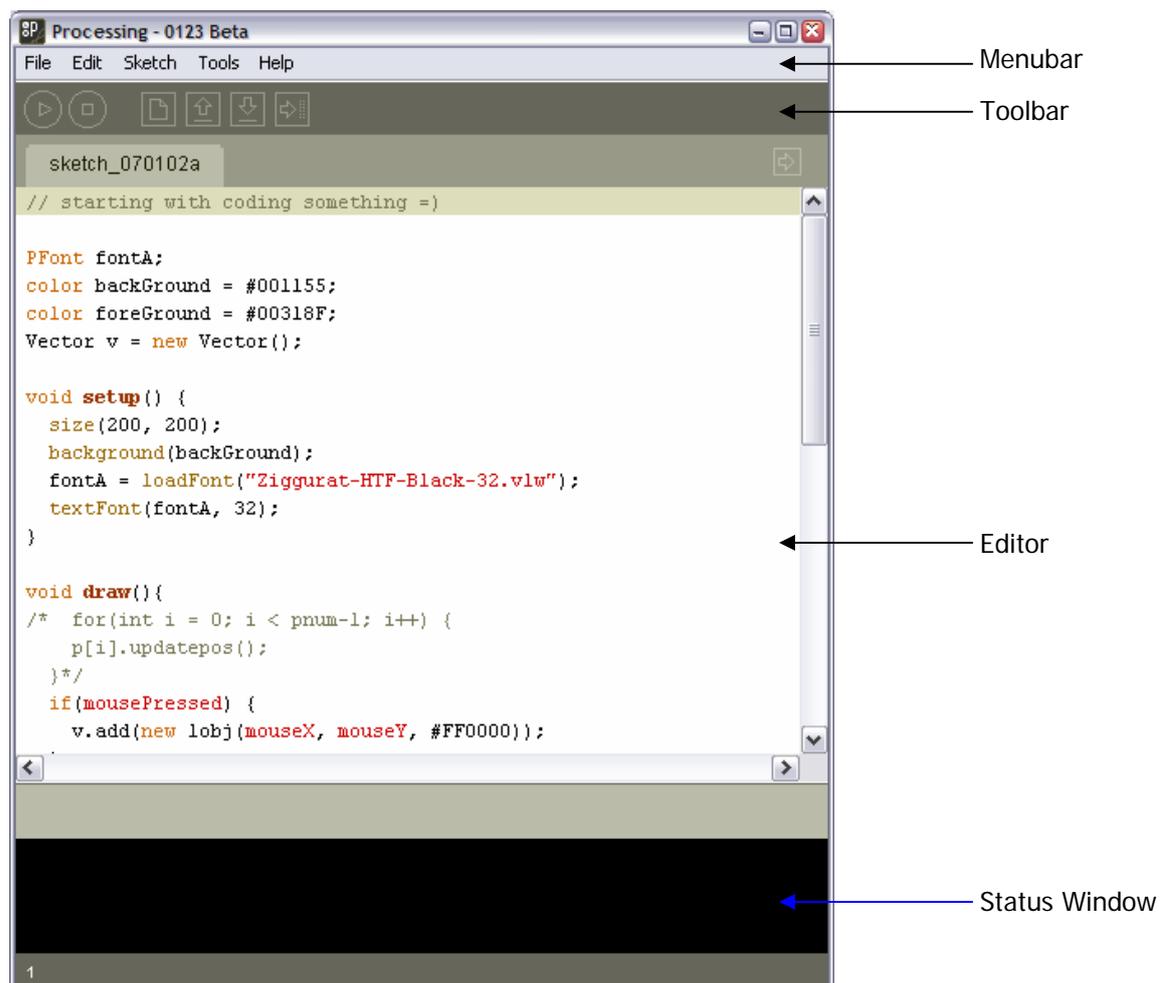
The Export buttons allow a sketch to be exported to either a Java™ Applet or a “standalone” application.

The IDE also includes the standard “Edit” menu which has such features as “Undo”, “Redo”, “Cut”, “Copy”, “Paste”, “Select All” and the calls to the search functionality.

The "Sketch" menu allows to "Run", "Present" and "Stop" the currently opened sketch. It also includes an "Import Library" item, which allows easy integration of libraries into the current sketch to extend its functionality.

Also in the Tool menu included are the following features:

- Auto Format (Ctrl-T): *Auto Format checks the amount of parenthesis and their correct use but its main feature is to intend blocks correctly and by that create more-readable code.*
- Create Font: *Allows the programmer to create bitmap fonts out of the standard system fonts to make them usable with Applets and some libraries. These are then saved in vlw files.*
- Copy for Discourse: *Copies the selected source code to the clipboard and includes bb-style markup for easy readability when copied to the Processing forum.*
- Archive Sketch: *Creates a zip archive with all the items belonging to the current sketch.*
- Color Picker: *This nice feature is does exactly what its name suggests, namely let you pick a color by clicking on a rainbow scale, which makes it really easy to find the right color even for people not used to HTML's #RRGGBB nomenclature.*



**Image 1: The Processing IDE also called "Sketchbook" with one loaded sketch. The main parts of the IDE are marked and labeled on the right.**

The screenshot above shows the IDE with an opened sketch. The editor has syntax highlighting and automatic tab indent. Although it works for starters and for fast experimenting with ideas and code, it does not replace a full text editor or programming IDE.

The sketches are saved in files with the extension *pde* and will be processed and translated into Java™ files when the export function is called.

## ***The Processing API***

Processing includes a set of core libraries which were developed by the creators of Processing to add basic functionality. These libraries include basic support for network transactions, OpenGL® support, video and graphics interfaces as well as PDF support.

### **Libraries**

This section tries to explain Processing's core libraries in more detail.

#### **XML Import**

This library provides simple and compact means to read from and write to XML data streams (e.g. files) with Processing. This is rather useful in Web 2.0 context but also in regular use. This library is strongly based on the NanoXML Lite project by Marc De Schneemaecker[11]; a drawback for this library is that it is intentionally kept compact, even to the fact that it is not a full-featured XML handling library. If extended functionality is needed community added libraries (e.g. the ones listed in the next section) can and should be used.

#### **Candy SVG Import**

This minimal import library extends Processing's functionality to be able to import SVG (Scalable Vector Graphics) files. This library was specially designed to work with Adobe Illustrator but should work with all SVG files. The advantages of SVG are that it allows infinite resolution and rather small file sizes.

#### **OpenGL®**

This library adds OpenGL® functionality to Processing to enable 3D accelerated graphics output. It facilitates the JOGL (Java™ OpenGL®) libraries[12] from within Processing. This library does not add any additional methods, fields or interfaces. It just uses the Processing language, but exports the graphics differently to be OpenGL® conform (and accelerated). Therefore programs must only import the library by stating "import processing.opengl.\*" at the beginning of their code and specify the OpenGL® renderer must be specified as the third pa-

parameter to the `size()` function and then the sketch is already OpenGL® accelerated, no other line of code should have to be changed.

## Video

This library allows Processing to play QuickTime videos files and also playback of video data from a camera. It supports USB and Firewire cameras as well as video cards with composite or S-video in. To work properly this library needs QuickTime for Java™ installed on the system where the application is running. It also needs information about the codec used in the video file, which are generally stored in specific video driver software. Currently this library only supports Mac and Windows. Also the quality of the video is not great but there are improvements done continuously, other possibilities for video processing are listed below in the community developed libraries section.

## Network

As its name suggests this library adds network functionality to the Processing language. In detail it adds Server and Client classes which ease the problem of reading and writing data across machines over a network (e.g. the Internet). Servers have a list of clients they are connected to, to read and write data from. A Client is connected to only one Server, from which it can read or write data.

## Serial

This library enables interaction with all kinds of serial devices (e.g. custom-made micro controller). This library uses only the standard 9-pin serial port of the system but USB can be mapped using tools such as the Keyspan USB Serial Adaptor.

## Other

The other core libraries are:

- PDF Export
- DXF Export
- Netscape.JavaScript

Apart from the core libraries Processing also has a large community developing libraries. These can be downloaded from the Processing webpage and then easily be integrated into a project. The finished applet will include the library code, so no extra software needs to be installed on the client side.

## Community Developed Libraries

The following is an (incomplete) list of libraries available for Processing:

- **Sound Manipulation:** libraries which allow easier interaction with audio streams e.g. analyze, play, stream or manipulate them.
  - Ess by Krister Olsson: *Sound library that allows sound files (e.g. sample data) to be loaded or streamed, generated in real-time, manipulated, saved, analyzed or simply played back.*
  - ID3 by Jorge Cardoso: *Allows Processing to read the well known ID3v1.1 tags from MP3 files, which contain meta information about the sound file (e.g. artist, title ...).*
  - jm-Etude by Daniel Dihadja: *Provides functions to communicate with jMusic, an extension for Java™ to ease sound handling[13], for easier music composition programming.*
  - oscP5 by Andreas Schlegel: *An Open Sound Control (OSC) implementation for Processing. OSC is a protocol for communication with sound synthesizers and other multimedia devices, which was developed at UC Berkeley[14].*
  - proMidi by Christian Riekoff: *A simple toolkit to send and receive midi information within Processing.*
  
- **Interfaces:** The following libraries supply a set of User Interface elements like e.g. Buttons, Sliders and such. The libraries vary in design, class structure and visualization.
  - controlP5 by Andreas Schlegel: *This library with custom built GUI elements allows having dynamic controls which are not conform to any Java™ standard.*
  - Interfascia by Brendan Berg: *This library also provides a set of non-Java™-standard interface widgets like text fields, buttons, checkboxes, sliders, etc.*
  - proCONTROLL by Christian Riekoff: *Allows Processing to communicate and interact with other (no mouse) control devices like joysticks and joypads.*
  - SpringGUI by Philipp Seifried: *This GUI library wraps around the standard java.awt (Spring) components and makes accessing them from Processing easier.*
  
- **Computer Vision:** the following is a list of libraries that try to make life easier for implementing CV problems with Processing by providing interfaces, algorithms and vision engines.
  - BlobDetection by v3ga: *Provides techniques to find "blobs" in an image.*
  - JMyron (WebCamXtra) by Josh Nimoy et al.: *Enables programmers to use a camera with motion detection, color tracking, glob distinction and pixel addressing, while not needing QuickTime or WinVDIG.*
  - LibCV by toxi: *Allows grabbing of video frames from streams (e.g. a camera) with the Java™ Media Framework (JMF)[15] instead of QuickTime.*
  - reactIVision TUIO by Martin Kaltenbrunner: *Object tracking with the reactIVision vision engine, which is a framework for tracking*

*fiducial markers attached to objects[16], is enabled with this library.*

- **3D:** To enable various 3D controls and helpers the following libraries can be used.
  - Fog by JohnG: *Just does what its name suggests. It provides simple fog effects for Processing using P3D and OpenGL® renderers.*
  - Gestalt by d3 and senor pako: *Gestalt is an open structured programming environment. The environment can be used to prototype and develop OpenGL®- and Java™-based sketches more easily.*
  - OCD by Kristian Linn Damkjer: *The Obsessive Camera Direction (OCD) library allows the creation of Processing view port cameras, and allows intuitive control of them. It also allows easy switching between the view ports and also moving the cameras with standard movement commands.*
  
- **Import and Export of Data:** Enables Processing to import various different data formats from the file system or the web and also enables output of information to those formats. The following also add support for data transmission with and without protocols and program and process interaction over networks.
  - Adobe Illustrator Export by Allen William Martin & Mark Hill: *Does exactly what its name suggests namely export the Processing generated animation to Adobe Illustrator for post-processing.*
  - Apple SMS by Daniel Shiffman: *This library allows getting hardware information from the Apple Sudden Motion Sensor (SMS) which is built-in into Powerbooks (and Macbooks) since 2005 to allow emergency shutdown of hard drives.*
  - Carnivore by RSG: *This is an interface to easily “sniff” TCP/UDP packets from the local network and use them in Processing.*
  - EnvironmentXML by Usman Haque: *This library was created to allow the programmer to easier use information posted at EnvironmentXML[17] and use them in the Processing application. EnvironmentXML was generated to have a repository of (hopefully) live data about the environment of a given location all around the world, e.g. temperature, wind speed, humidity in London.*
  - Google API by Tatsuya SAITO: *Is a library to access Google Search, which allows sending queries and retrieving the results, from within Processing.*
  - MovieMaker by Daniel Shiffman: *Generates QuickTime movies out of the Processing application for easier viewing and distribution.*
  - MySQL by Florian Jenett: *Allows easy communication with MySQL databases from within Processing applications.*
  - OBJ Loader by Tatsuya Saito: *This plug-in allows Processing to import and export information into the .obj file format, which is used for object data including geometry information and texture data.*
  - proHTM and proXMLby Christian Riekoff: *Allows Processing to take an XML or HTML document and change it from a linear string to a*

- tree structure therefore making parsing of the file and accessing the wanted information easier.*
- SimplePostScript by Marius Watz: *Allows easier interoperability with PostScript files and data.*
  - Social Networks Library by Todd Holloway: *This library provides general network library components but emphasizes on tools for selecting subsets and traversing the network, which is quite useful in connection with the currently growing number of Net2.0-social-networking-websites and data extracted from there.*
  - Switchboard by Jeff Crouse: *Like the Google API but more advanced and it allows to access information from a variety of web services, such as Google, Yahoo, Amazon, Del.icio.us, Flickr, and many more.*
  - xmlrpc-lib by Burak Arikan: *An XML-RPC library for Processing, which allows making use of the Remote Procedure Calling protocol over the Internet with XML data structures.*
- **Simulation and Math libraries:** The following libraries provide useful extensions for the field of simulation and mathematics as well as physics and similar scientific notions.
    - AI Libraries by Aaron Steed: *These libraries should assist the programmers in generating artificial intelligence tasks such as genetic algorithms and the A\* algorithm[18].*
    - Cell Noise by Carl-Johan Rosén: *For an easy-to-use pattern generation this library adds a cell noise (also known as Worley noise) generator, which is quite useful for animations.*
    - Mersenne Twister by Marius Watz: *A random number generator at a very high and sophisticated level.*
    - Oscillator by taka: *A library that helps with the generating waves or series thereof.*
    - Physics by Jeffrey Traer Bernstein: *This is a very prominent and widely used library that adds a simple particle system physics engine. It does not include collisions but particles, springs, gravity & drag. This library is also widely known as "Traer library".*

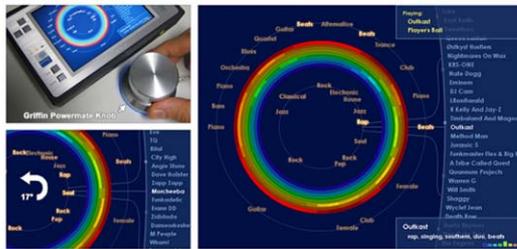
Apart from using Processing libraries, which is the preferred style to reuse code, Java™ libraries can usually be used too. To make a regular Java™ library usable from within a sketch the library file(s), either DLL or JAR, have to be added to the code folder of this sketch. This can easily be done by using the entry "Add File" in the "Sketch" menu. After adding the libraries, they can be used and imported like any other Processing library.

As of January 2007 Processing does not include any Bluetooth libraries, which would be quite useful to develop nice applications for e.g. the Wii™ Remote[19] controller. Mobile Processing does include a library for handling Bluetooth on e.g. mobile phones, more about that further down.

## What is it used for?

In this section a list of projects where Processing was used will be shown, this list is split up into different categories like visualization, hardware and so on. The projects are described with a few sentences and an image of the application is presented. Those images are taken from the Processing website or the respective project website which is also linked in the description.

### Visualizations



#### Music Rainbow

by Elias Pampalk & Masataka Goto[20]

Category: Visualization & Interaction

This visualization is based on a collection of artists. The artists are projected onto a circle, where artists whose music is similar are placed close

to each other.

By computer analysis of the audio contents of their songs the proximity and similarity of artists is calculated. To map the artists onto the circle a "traveling salesperson" algorithm[21] is used. The different colors encode different musical styles, whereas words describe the different regions of the rainbow, these words are extracted from websites mentioning the artist's name. On the right side a magnification of the currently selected part of the circle is visible to ease the user control. The selected artist is highlighted white. On the lower right side a summary of the keywords and colors for this artist is shown.

The user input is controlled by turning a knob to select an artist and pushing it to listen to the selected artist. In this installation a Griffin PowerMate knob was used.

Link: <http://staff.aist.go.jp/elias.pampalk/MusicRainbow/>

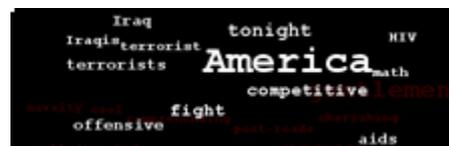
#### State of the Union by Brad Borevitz

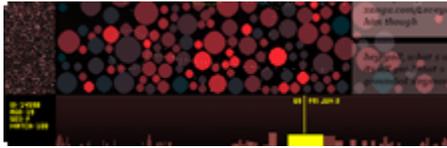
Category: Visualization

This project by Brad Borevitz is using Processing to display information visually. The

information is gathered from the State of the Union addresses of the US President from 1790 to 2007. Borevitz wants the users to get information about the connection between the state of the union and the use of words. The users can interactively scroll through the timeline and see what words were used at most and how they were connected with the rest of the text. They also have the chance to retrieve additional historical information that is linked from the applet.

Link: <http://stateoftheunion.onetwothree.net/>





## The Dumpster by Golan Levin et al

Category: Visualization

This is a project that tries to use a big data source from blogs on the World Wide Web to generate "a portrait of romantic breakups". The data is gathered from blogs that use the terms "break up" or "broke up" in any sentence. The whole posting is saved and can be access from the Processing applet, which shows the whole year period of 2005. The postings are represented as bubbles which do vary in size, color and speed.

Link: <http://artport.whitney.org/commissions/thedumpster/>

## Rewiring the Spy

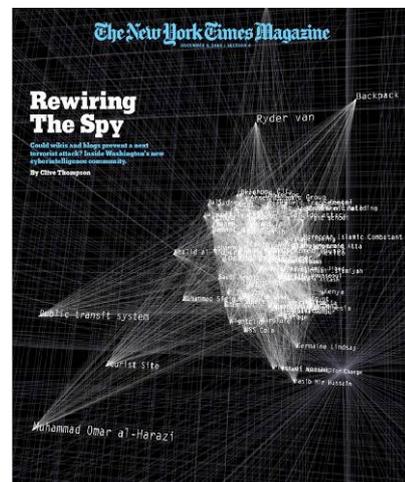
by Lisa Strausfeld and James N. Sears

Category: Visualization



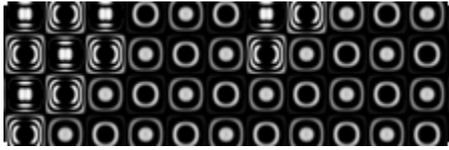
Lisa Strausfeld and James Nick Sears, both designers at various design companies and advisors, designed the illustrations for the cover story of the December 3<sup>rd</sup> issue of The New York Times Magazine. Their illustrations, which are named "Rewiring the Spy", are based on the data retrieved from "Web 2.0" applications like blogs and wikis. The cover story is trying to find out if and how this data can be used by such agencies like the C.I.A. and F.B.I. to combat terrorism. The visualizations are created in a 3D space in which, through the use of the physics library (mentioned above), the physical relationships between actors, weapons and targets are drawn. The number of connections should suggest their level of connection in an attack.

From the article: *"A number of times each year, we find that our cover article is better suited to illustration than to photography [...] (because) the substance of the reporting doesn't ultimately exist in the corporeal world [...] (like with) Clive Thompson's article on the intelligence community's quest to better utilize the tools of digital technology. Thompson, a contributing writer, reports regularly on high-tech matters. [...] Drawing on information maintained on a large counterterrorism database, the team created a pictorial equation in which terms that showed up most frequently in word searches interacted with one another. The resulting patterns are quite revealing, and quite beautiful."* [22]



**Image 2: The cover of The New York Times Magazine on December 3<sup>rd</sup> 2006**

Links: <http://jamesnsears.com/applets/spies/> also [23]



### Polyptic by George Legrady

*Category: Visualization*

This is a visualization that has a two-state matrix, which is represented by two different images. The cells can switch from one to the other state, but how often depends on their neighboring states and a temperature function. The hotter it gets the more random the swaps are, if the temperature normalizes again the cells try to switch to the same state as their neighbors. This function is calculated with a sine wave.

Link: <http://www.mat.ucsb.edu/~g.legrady/glWeb/Projects/algo2/i/>

### Letter-pairs Analysis

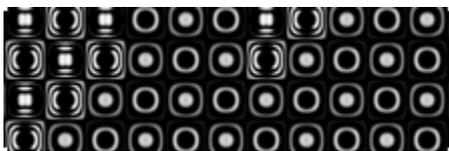
by Martin Ignacio Bereciartua

*Category: Visualization*



This application is able to read a text (in the online version only a predefined text) and while doing so updates the letter-pairs it finds in real time. Those letter-pairs are represented by bubbles, the more often those pairs appear the bigger the bubbles get. For different languages different, but quite characteristic, graphs are generated. The offline version facilitates OpenGL® acceleration.

Link: [http://www.m-i-b.com.ar/mib/letter\\_pairs/eng/](http://www.m-i-b.com.ar/mib/letter_pairs/eng/)



### Genome Valence by Ben Fry

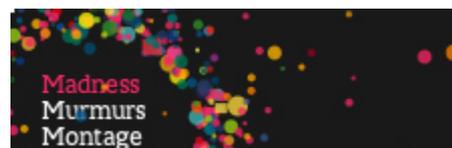
*Category: Visualization*

To easily find out if genome sequences are used in other organisms, Ben Fry created this applet, which displays the 9 letter parts of genome that are usually used in Genetics visualizations (BLAST algorithm[24]). This application is part of a bigger project focusing on how to build visual representations of very large data, in particular genomic information. The user can input the sequence by selecting it from visual strings, that either displays the letters or other specifics. Those components are then placed on the bottom of the screen in a sequence.

Link: <http://acg.media.mit.edu/people/fry/genomevalence/>

**We Feel Fine** by Jonathan Harris and Sepandar Kamvar

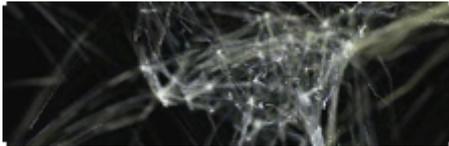
*Category: Visualization*



This visualization tries to display a huge data collection of feelings from the internet. The information is gathered from blogging services like Blogger and their large number of blogs. The posts are searched for sentences containing the words feeling or "I feel", these are then stored in a huge database, with information about the blogger, which is retrieved from their profile page. The database is then scanned and checked against a predefined list of approximately 5000 feelings which are assigned different colors. If

images are found in the posting those are saved too. The system scans thousands of blogs every 10 minutes and so gathers between 15000 and 20000 feelings per day, which are then statistically computed. Processing is used to visualize that data in one big applet. The data is then shown in 6 different movements, which vary from just colored pixels ("madness"), over pictures with phrases superimposed ("montage"), to mounds which just show the overall feeling of "the internet".

Link: <http://www.wefeelfine.org/>



### **Flight Patterns** by Aaron Koblin *Category: Visualization*

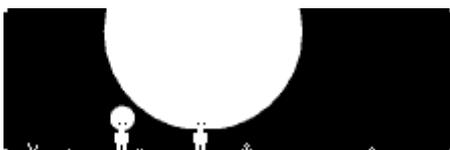
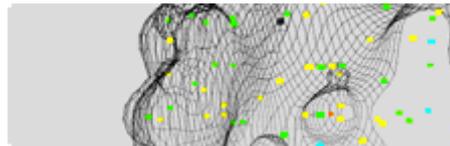
This visualization tries to plot the information from the US Federal Aviation Authority (FAA) about all the flights onto a map or a sphere representing the world. Out of this project the "Celestial Mechanics" project[25] was created which tries to map and show all vehicles flying on and off the earth.

Link: <http://www.aaronkoblin.com/work/faa/>

### **Base26** by Toxi *Category: Visualization*

This is spatial overview over 4 letter words in the English language. Those words are arranged in 4 dimension based on the characters they consist of. The first letter gives the "time" dimension whereas the other 3 give the x, y, and z axis of the word. Around the 3 letter word a wire frame structure is generated which indicates how often that 3 letter combination is used in all 4 letter words.

Link: <http://www.processing.org/exhibition/works/base26/index.html>



### **Inequality** by Josh On *Category: Visualization*

In 2001 American CEOs were paid 411 times more than the average worker. This applet tries to visualize the development over the last 50 years. Although kept quite simple the message is quite clear.

Link: <http://www.processing.org/exhibition/works/inequality/index.html>

### **Mutualism** by David Pereira *Category: Visualization*

Mutualism is an information aggregator for a shared public server. It displays captured information to create strings and lines to visualize connections and fluxes between the users and the objects they access on the server.

Link: <http://jamesnsears.com/applets/spies/>





### **Visual Scratch** by Jesse Kriss *Category: Visualization*

Visual Scratch does, as its name suggests, visualize the scratching of a music record.

The sound was recorded on turntables by a professional DJ and then visualized with Processing. It uses the Ess, PDF and MovieMaker libraries to display the velocity of the turntable, the frequency content, and the volume of the audio.

Link: <http://jklabs.net/projects/visualscratch/>

### **Blinks & Buttons** by Sascha Pohflepp *Category: Visualization & Hardware*



This thesis project tries to explore deeper the information gathered while taking pictures. This information is usually saved in the Exif tag[26]. This project consists of two parts, the first called Blinks is a visual installation that allows users to scan through their photos, thereby remembering moments. To this picture which can be selected by some prism that can be put on the projected photo other pictures will be displayed on the walls around the table. These pictures have been taken at the same time as the picture you selected but by different people. The user can then see what others were doing in that moment. The prism allows also scrolling through the other pictures which are such positioned that they represent the 24 time zones.

The second part of the project called Buttons is trying to remove the coupling of picture and time. It is a camera with just a button and no optical elements. It just stores the time the user pressed the button and starts searching for pictures that were taken at exactly this moment on the internet. This project wants to create connection between people that pressed "the button" at the same time.

Link: <http://www.blinksandbuttons.net>

## **Hardware**

### **PixelRoller** by rAndom international *Category: Hardware*



The PixelRoller project created a hardware utility very similar to a paint roller. In fact it is a point roller which paints pixels. To do that digital information such as pictures or text are uploaded to the PixelRoller device, then the user uses it like a regular paint roller. This gadget provides users with the ability to print digital information very rapidly and on a very broad variety of surfaces. The PixelRoller webpage says this about it: *"The option to print anywhere, onto anything, is an important factor which will provide the users with the ability to print digital content onto floors, walls and ceilings, both indoors and outdoors. Further substrates will include surfaces such as paper, brick, concrete, glass, and grass. And*

*all of them will be printed onto in one stroke, without the need of changing the process in which to print." [27]*

Processing was used for developing the software running on the first prototype.

Link: <http://www.random-international.com/pixelroller-overview>

## Video



### **Yo soy la Juani** by Aleix Fernandez

*Category: Video*

The main title sequence of Bigas Luna's feature film *"Yo soy la Juani"* (2006) was generated with a Processing applet created by Aleix Fernandez, working at onionlab.com. The applet generates a set of lines and with keystrokes their properties can be changed. For example how many lines there are, how far the lines are apart, the "bending" of the lines, and so on can be controlled. With another set of keys (or a joy pad) the camera can be moved. After the lines were generated the movie was saved as a set of pictures and was later post processed by postproduction movie software. In this step was the title and the text introduced to the video.

Link: <http://www.onionlab.com/processing/YSLJTtitles.html>

### **Getting Away with Murder**

by Gabriel Dunne

*Category: Video Animation*



The music video to Papa Roach's "Getting Away with Murder" was realized by Motion Theory. They shot the video in front of a green screen and added animations that would look like stock tickers in the background. Those were generated by a Processing applet. Firstly animations in solid state pixels were generated for a few sequences (like bombs, explosions, jets, and texts). From these the applet generated the ticker-like animation of images and fixed sequences. These were then put into the green screen areas in the music video.

Link: <http://www.processing.org/exhibition/works/papa/index.html>



### **Nike One** by Motion Theory

*Category: Animation & Video in Advertising*

For Nike's new ads for their golf equipment Processing was used to visualize the thoughts and mechanics of the creative process of the Nike engineers behind the new equipment. The visualizations should show the thought process, science, and the imagination of the engineers and therefore fluidly be placed in the video of the ad. The various floating elements were subclasses of one class developed within a team of 4 at Motion Theory. They were using customized common rendering frameworks with shared data protocols and versioning to allow having

multiple floating elements, e.g. TechBabble, charts, text, spring boxes, flux diagrams and so on.

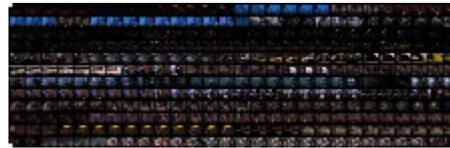
Links: [http://www.motiontheory.com/work/nike\\_one/](http://www.motiontheory.com/work/nike_one/)  
<http://dev.motiontheory.com/nikegolf/>

### **Cinema Redux** by Brendan Dawes

*Category: Video Processing*

This visualization called Ciname Redux tries to explore what happens if a whole movie is distilled into one single image. It is using data from 8 different movies from which it takes a frame sample every second. This sample is then converted into an 8x6 pixel image which is then placed into the final image. Each row therefore represents one minute of the movie. The end result is a kind of unique fingerprint for this movie, which not only shows the color hue, and therefore the overall feeling, of the movie, but also enables a comparison between different editing styles from slowly paced Coppola movies to the faster paces in Lumet's.

Link: <http://www.processing.org/exhibition/works/redux/index.html>



## **Animation**

### **Grass** by The Barbarian Group

*Category: Animation/Visualization*

For the presentation of the new Saturn hybrid cars at Wired Magazine's NextFest in 2006 the Barbarian Group realized 45 feet of virtual grass swaying in a virtual breeze. The animation was done with Processing and then displayed via four DLP projectors on a rear projection screen. The visitors could input text from computer terminals which were then attached to one grass blade and swayed along with the grass. The swaying was coupled with an infrared camera that recognized the visitors walking by, so the visitors were creating the breeze by walking in front of the projection screen.

Single blades of grass were visualized with a simple system of variable decay, i.e. that every blade was constructed of fixed number of segments with height and width being a fraction of the segment the current segment is sitting on. This leads to a progression of the max width to zero and the max height to the half of the given height. The bending is also done by a similar variable decay framework.

For the creation of the breeze the programmers used perlin noise[28], which is a random number generator, which has the property that it produces smoothly changing set of numbers without too much sudden change. A perlin noise generator is built-in into Processing.



Link: <http://portfolio.barbariangroup.com/nextfest/index.html>



## Shadow Monsters

by Philip Worthington

*Category: Interactive Animation*

Shadow Monsters is an interactive installation that tries to display and visualize shadow monsters cast by the hands of the person in front of an infrared camera. The gestures are redone into characters like wolfs, birds and a Rastafarian which open their mouths and start squeaking if the people move their hands. It uses similar technology to another project called Ants[29] also by Philip Worthington. The Shadow Monsters project was since then applied to various locations and events like e.g. Nokia's "Leave No Trace" road show, where it was shown at locations in Holland, Italy, Poland, Croatia and Finland.

Link: <http://www.worthersoriginal.com/viki/#page=shadowmonsters>

## Visual and Interactive Installations



## Ping Genius Loci

by Aether Architecture

*Category: Interactive Installation, Hardware*

Ping Genius Loci (PGL) is an architectural installation project that tried to build "a network into the poetics of the place". The installation was built 300 radio networked, solar powered, self sustainable intelligent analogue pixels, that are placed on a 20 by 20 meters grid. The function of the pixels is optimized for working in bright sunshine. The pixels interact with the people walking through the pixel grid.

Link: <http://www.aether.hu/pgl/usingprocessing.html>

## Khronos Projector

by Alvaro Cassinellirass

*Category: Interactive Installation & Demo*



The Khronos Projector project is an interactive installation that allows users to control movies with a tangible interface. The projection screen can be touched and with this the movie is played forward or backwards and a time-lapse can be seen. For testing the algorithms and presenting them to the internet community the project tea used Processing to show pre-loaded time-lapse sequences. The input in the case of the applets can be given using the mouse.

Link: [http://www.k2.t.u-tokyo.ac.jp/members/alvaro/Khronos/Khronos\\_P5](http://www.k2.t.u-tokyo.ac.jp/members/alvaro/Khronos/Khronos_P5)



## Net Worth by Osman Khan

*Category: Interactive Installation*

With this installation the user can see how much they are "worth" on the Internet. Af-

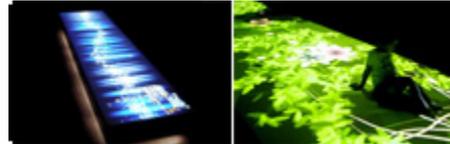
ter reading their name via a magnetic card reader from e.g. a credit card, a search with the user's name is sent to Google. The number of results for the user's name is returned and according to that the name is displayed on a big projection screen, with people who have the most results being on top. For reference purposes famous people are also included. This gives a chance to somehow gasp the state of the internet, for example Paris Hilton is ranked way higher than William Shakespeare.

Link: <http://www.osmankhan.com/networth/>

### Process by ART + COM

*Category: Visual Installation / Virtual Reality*

ART+COM did a few visual installations with Processing. These include a 9 meter long interactive table, which only shows numbers but when they are touched it reveals their secrets, a complete virtual garden, as well as a whole exhibition dealing with alien life. The user can interact with the alien life and through changes in the environment change the lifeform.



Link: <http://www.artcom.de/process/>

## Art and Artistic Presentations



### VJ by Robert Hodgins

*Category: Artistic Presentaion*

This is a visualization utility that was implemented and used for VJing performances at the Bridge, Toronto. Real time graphics mixed with predefined structures are generated and can be controlled by using Griffin Powermates knobs.

Link: <http://www.flight404.com/version8/index.html>

### Metropop Denim

by Clayton Cubitt and Tom Carden

*Category: Artistic Presentaion*

Visualizations generated by attractors and pixels running in a physically correct surrounding generated random pictures which then where used for fashion photography shootings to give the pictures are more artsy look.

Link: <http://www.processing.org/exhibition/works/metropop/>



Another artistic visualization concept is the "chronotext" concept, which is a collection of software experiments that tries "exploring the relation between text, space and time"[30]. Some very interesting projects have spawned from this work of Ariel Malka, a designer and programmer based in Tel-Aviv.

## Image Processing



### **Dreamlines** by Leonardo Solaas *Category: Image Processing*

This project focuses on visualizing a dream-like sequence of images that fit a user input. These pictures are never completely shown nor drawn. 1500 autonomous particles generate the lines of the pictures by starting from a random point and then reaching their original point.

Link: <http://www.solaas.com.ar/dreamlines/p5/>

## Audio Processing

### **Sonic Wire Sculptor** by Amit Pitaru *Category: Audio Processing*

This application allows drawing "lines" with a tablet onto a "rotating table", a 2D structure that rotates. These lines are represented as sound, where the pitch is defined with the position and the volume by the thickness of the lines. Nice features are the turnings of the line in 3D, e.g. mirror, rotate in 3D, tilting, and so on. These allow for quite interesting phenomena out of simple sculptures.

Link: <http://www.pitaru.com/sonicWireSculptor/framed/>

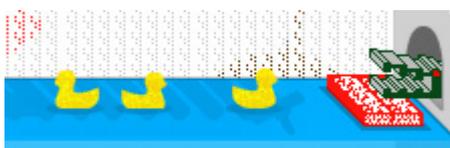
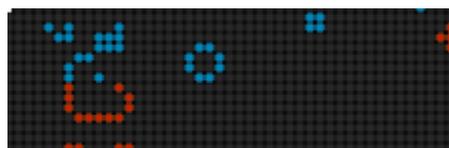


## Games

### **Life vs. Life** by Justin Bakse & Eric Williams *Category: Game*

This game called "Life vs. Life" is a two player adaptation of the mathematical game of Life[31], which was invented by John Conway and is based on cellular automata. The applet explains the rules of the game and then allows you to pick two boards (one per player) which then play against each other. The one who can take the most cells per generation wins the game.

Link: <http://www.lifevslife.com/>



### **Friendster Pachinko** by Krister Olsson, sound by Yoshi Sodeoka *Category: ....*

This applet is using the user's friendster[32], a social network website, for enemies[32] and produces a Pachinko game.

out of her/his friends. This Japanese game is similar to pinball and in this variant the users play with friends instead of metal balls.

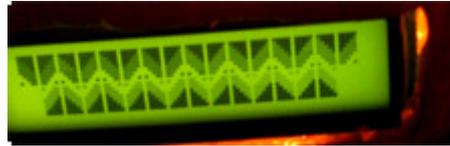
Links: <http://www.tree-axis.com/072904/pachinko.php>

### **Jack O Lantern** by Elise Co

*Category: Game & Hardware*

In this game the user gets points for feeding a pumpkin with candy. The internet version has a flying pumpkin which candy can be thrown into but points are only generated if the pumpkin has an open mouth. In real life this mouth is realized using an off-the-shelf serial LCD module which is built into a pumpkin.

Link: <http://www.processing.org/exhibition/works/jackolantern/index.html>



### **Thinking Machine 4**

by Martin Wattenberg

*Category: Game & Interactive Installation*

This version of a chess game tries to explore and make visible to the user the invisible nature of thought. The user plays chess against a computer intelligence, whose moves and thought process are visualized by arcs. The computer thinks ahead a few moves and calculates the users and its own moves and tries to find the best strategy. The calculations of the fitness of the next moves are calculated and represented in the strength and thickness of the arcs. This internet version has different variants, e.g. one that is an actual chess table with a touch screen for user interaction.

Link: <http://www.turbulence.org/spotlight/thinking/>

A more extensive and constantly updated list can be found at the "Exhibition" section of the processing website[33].

Apart from being used in projects Processing has been used in various lectures, at universities around the globe, as introduction to programming or for media designers, to show them the idea behind using computer programming in a way to help artists and designers. Courses to mention are the ones offered by the Massachusetts Institute of Technology (MIT)[34] and the New York University (NYU)[35]. There have also been quite a lot of workshops about and around Processing on various meetings and conferences, e.g. on the New Media Meeting in Sweden[36].

### 3. Extensions and Rewrites

Although Processing has finally released its version 1.0, in April 2005, it has already spawned a lot of alternatives, rewrites and extensions. Those listed here are the most well known and extensive ones.

#### ***Mobile Processing***

Mobile Processing[37] is a project initiated by Francis Li[38], a California-based interaction designer and software engineer, in 2005. It tries to change the Processing programming environment to fit for people wanting to design, sketch and prototype software for mobile phones. The idea is that sketches programmed using Mobile Processing run on Java™ Powered mobile devices. The code generated by Mobile Processing is using J2ME (Java™ 2 Micro Edition) and the so called MIDlets run with the releases CLDC 1.0 and MIDP 1.0 of J2ME. Mobile Processing currently only works with Mac OS X and Windows but other ports are in development.

#### ***Wiring and Arduino***

Wiring[39] is a project by Hernando Barragán[40], from the Architecture and Design School at the University of Los Andes, Colombia, which is based on Processing but also tries to integrate easy handling of electronics I/O boards. This enables better development of tangible media and generally real world interfaces, as well as teaching and learning computer programming and prototyping with electronics. The Wiring I/O board[41], which is special electronics hardware, allows all kinds of interaction with sensors and actuators. It includes 43 digital I/O pins, 5 digital I/O ports, 8 analogue inputs, and 6 PWM (pulse width modulated) analogue outputs. The board can be connected to any computer using an USB connector.

Arduino[42] is an offspring of the Wiring project, which uses a different set of hardware, but the same programming language. The Arduino board can have multiple variants, but most use the Atmel chips ATmega8 or ATmega168. The standard Arduino board is a cheap, robust I/O board based on the Atmel ATmega8. It has 13 digital I/O pins (3 of which allow PWM output) and 6 analog inputs. The boards can be built by the users themselves or can be bought preassembled. In a sense it is an extension of the Wiring software to work with a bigger amount of different I/O hardware.

## ***Nodebox***

The Nodebox[43] project tries to do the same with Python, that Processing did with Java™. It is a programming environment for Nodebox code, which is based on Python, but tries to ease visual programming with it. The generated visuals can be saved as QuickTime movie or PDF document. Currently Nodebox is available in version 1.0RC7 and only for Mac OS X. Nodebox only supports 2D visualization so far but is actively developed and will be constantly extend its functionality.

## **4. Conclusion and Outlook**

Processing can be seen as new programming language to ease the creation of creative code, like visualization and animation. The list above shows the big areas of computation that Processing can be used for. Although visualization play a very big role, tangible interfaces and interactive installations are also quite easy to implement. This is also due to the emerging of other projects, like the above mentioned Arduino project, which helps in using electronically controlled sensors and actuators. A project that uses these massively is the Proximity Lab[44].

The whole area of visual programming, visualization and more generally the connection between design and computation is a very sprouting area in computer science. In my opinion this is also in connection with the evolution of the WWW, which is not just HTML anymore but uses a lot of designing especially in the growing field of Web2.0 applications. On the other hand more and more designers, architects, and artists are using computers to not only extend their productivity but also their creativity. This can also be seen by number of websites dedicated to design and creativity with computers. For example the website [visualcomplexity.com](http://visualcomplexity.com), which by its creator is "to be a unified resource space for anyone interested in the visualization of complex networks"[45]. This project's main goal is to leverage a critical understanding of different visualization methods, across a series of disciplines, as diverse as Biology, Social Networks or the World Wide Web. I truly hope this space can inspire, motivate and enlighten any person doing research on this field". A collage of the things published to [visualcomplexity.com](http://visualcomplexity.com) can be seen in Image 3.

Of course Processing is not the solution to all problems; it eases the effort for programmers to create visual programming interfaces and has the ability to create new and easy to use (graphical) user interfaces, but that does not necessarily lead to easier-to-handle applications. The visualization done by Lisa Strausfeld and James N. Sears, that was used by The New York Times Magazine, did a great job in visualizing complex connections and the pictures used for the magazine were pretty neat, but the application especially the user interface of it was quite a bad one. Only after extensive use or knowing the backgrounds one could

generate those wonderful images. So Processing does not always create usable or intuitive UIs this is still a job for the programmer or more specifically the User Interface Designer.



**Image 3: Taken from [visualcomplexity.com](http://visualcomplexity.com)**

In my opinion Processing brings lots of possibilities to the users, I can imagine further projects with Processing (or one of its variations) that try to mimic an interactive table like the one presented by Jeff Han on his website[46], maybe not as sophisticated in a technical sense but with more community developed (visual) applications. A lead in this direction could be the MultiTouch project by c-base[47] in Berlin, which also tries to rebuild a multitouch table with an open technology behind it.

A nice extension for Processing would be the implementation of a general multi-touch library, to implement current (or legacy) multi-touch interfaces[48] or develop new ones easily, maybe at first simulated by multiple mice input, or generally more libraries to enable programmers to use Processing for their next generation user interfaces or man-machine-interfaces. The groundwork has already been done by Jens Wundering who implemented a multi touch library but based on video input[49].

Another idea for extension would be to build a fully functioning Bluetooth library for Processing to e.g. use the Nintendo Wii™ Remote to control a processing application. This would enable another degree of freedom in user input (like shown with this Google Maps[50] demo). I was trying to find implementations of this as part of my practical work for this thesis but did not find anything that would work well. There are people working on this[51] but nothing of this seems to be working enough to be considered beta state.

## References

Apart from using the homepage of Processing: <http://www.processing.org>, which is quite a resource for finding Processing-related materials the following references were used:

- [1] Ben Fry at Carnegie Mellon University, [http://www.design.cmu.edu/show\\_person.php?t=f&id=BenFry](http://www.design.cmu.edu/show_person.php?t=f&id=BenFry)
- [2] Casey Reas at University of California, Los Angeles, <http://users.design.ucla.edu/~reas/>
- [3] aesthetics + computation group, Massachusetts Institute of Technology, <http://acg.media.mit.edu/>
- [4] Develop Processing (Examples), <http://dev.processing.org>
- [5] Libraries for Processing, <http://processing.org/reference/libraries/>
- [6] Contribute to Processing, the community around it, <http://processing.org/contribute/>
- [7] Design By Numbers, <http://dbn.media.mit.edu/>
- [8] Presentation of the Prix Ars Electronica 2005 Winner in the category NetVision, [http://www.aec.at/en/archives/prix\\_archive/prix\\_projekt.asp?iProjectID=13390](http://www.aec.at/en/archives/prix_archive/prix_projekt.asp?iProjectID=13390)
- [9] Download Processing, <http://www.processing.org/download/>
- [10] Processing community forum, [http://processing.org/discourse/yabb\\_beta/YaBB.cgi](http://processing.org/discourse/yabb_beta/YaBB.cgi)
- [11] NanoXML is a small XML parser for Java, <http://nanoxml.cyberelf.be/>
- [12] Java™ Binding for the OpenGL® API, <https://jogl.dev.java.net/>
- [13] Music composition in Java™, <http://jmusic.ci.qut.edu.au/>
- [14] OpenSound Control Home Page, <http://www.cnmat.berkeley.edu/OpenSoundControl/>
- [15] The Java™ Media Framework API, <http://java.sun.com/products/java-media/jmf/>
- [16] reactIVision 1.3, <http://www.mtg.upf.edu/reactable/?software>
- [17] environmentXML datastreams, <http://www.haque.co.uk/environmentxml/live/>
- [18] A\* search algorithm, [http://en.wikipedia.org/wiki/A%2A\\_search\\_algorithm](http://en.wikipedia.org/wiki/A%2A_search_algorithm)
- [19] Nintendo Wii™: Controller, <http://wiiportal.nintendo-europe.com/13.html>
- [20] Elias Pampalk & Masataka Goto, "MusicRainbow: A New User Interface to Discover Artists Using Audio-based Similarity and Web-based Labeling", in the Proceedings of the ISMIR International Conference on Music Information Retrieval, 2006.
- [21] Travelling salesman problem, [http://en.wikipedia.org/wiki/Travelling\\_salesman\\_problem](http://en.wikipedia.org/wiki/Travelling_salesman_problem)
- [22] New at Pentagram: New Work: The New York Times Magazine, [http://blog.pentagram.com/archives/2006/12/new\\_work\\_the\\_new\\_york\\_times\\_ma.php](http://blog.pentagram.com/archives/2006/12/new_work_the_new_york_times_ma.php)
- [23] Visualcomplexity: Rewiring the Spy, [http://www.visualcomplexity.com/vc/project\\_details.cfm?index=396&id=396&domain=](http://www.visualcomplexity.com/vc/project_details.cfm?index=396&id=396&domain=)
- [24] Basic Local Alignment Search Tool (BLAST) algorithm, <http://en.wikipedia.org/wiki/BLAST>
- [25] The Celestial Mechanics project, <http://www.cmlab.com/>
- [26] Exchangeable image file format (Exif), <http://en.wikipedia.org/wiki/Exif>
- [27] The PixelRoller project, <http://www.random-international.com/pixelroller-overview>
- [28] Perlin noise, [http://en.wikipedia.org/wiki/Perlin\\_noise](http://en.wikipedia.org/wiki/Perlin_noise)

- [29] The Ants project from a collection of works by Philip Worthington,  
*<http://www.worthersoriginal.com/wiki/#page=ants>*
- [30] Chronotext homepage, *<http://chronotext.org/>*
- [31] Life or Conway's Game of Life, *[http://en.wikipedia.org/wiki/Conway%27s\\_Game\\_of\\_Life](http://en.wikipedia.org/wiki/Conway%27s_Game_of_Life)*
- [32] Friendster, *<http://www.friendster.com>*
- [33] Exhibition of projects done with Processing, *<http://www.processing.org/exhibition/>*
- [34] OPENCODE web-based programming environment, *<http://opencode.media.mit.edu/>*
- [35] Course: Graduate Multimedia, Fall 2006, Media Research Lab, New York University,  
*<http://mrl.nyu.edu/~perlin/courses/fall2006mm/>*
- [36] New Media Meeting 2006, *<http://newmediameeting.se/>*
- [37] Mobile Processing, *<http://mobile.processing.org/>*
- [38] Francis Li, *<http://www.francisli.com/>*
- [39] Wiring, *<http://wiring.org.co/>*
- [40] Hernando Barragán, Interaction Design Institute Ivrea, Italy  
*<http://people.interaction-ivrea.it/h.barragan/>*
- [41] The Wiring I/O board description, *<http://wiring.org.co/ioboard/>*
- [42] Arduino, *<http://www.arduino.cc/>*
- [43] NodeBox, create visual output with Python programming code, *<http://nodebox.net/>*
- [44] The Proximity Lab project, *<http://www.proximitylab.org/>*
- [45] About visualcomplexity.com, *<http://www.visualcomplexity.com/vc/about.cfm>*
- [46] Multi-Touch Interaction Research, Jeff Han, New York University  
*<http://cs.nyu.edu/~jhan/ftirtouch/>*
- [47] mtc – Multi Touch Console by c-base, *<http://www.multi-touch.de/>*
- [48] Bill Buxton, Microsoft Research: Multi-Touch Systems that I Have Known and Loved,  
*<http://www.billbuxton.com/multitouchOverview.html>*
- [49] How to build a (camera tracking) multi-touch via Pixelsumo,  
*<http://www.pixelsumo.com/post/jens-wunderling-multitouch>*
- [50] Video: Controlling Google Earth with the Wii Remote,  
*<http://www.handcircus.com/2006/12/15/controlling-google-earth-with-the-wii-remote/>*
- [51] Processing forum: Topic: using DarwiinRemote in Processing,  
*[http://processing.org/discourse/yabb\\_beta/YaBB.cgi?board=os\\_libraries\\_tools;action=display;num=1166025320](http://processing.org/discourse/yabb_beta/YaBB.cgi?board=os_libraries_tools;action=display;num=1166025320)*

Internet resources were again checked for existence and content in February 2007.