

## „Maria im Schnee“

### Computer Installation

Wir sehen eine Mutter mit Kind. Vor ihr fällt Schnee. Scheinbar. ...

Warum ist das kein Video?

Ein Video zeigt uns zu jedem Zeitpunkt dasselbe Bild. Wenn ich das Video-Tape zum Beispiel bei einer Minute stoppe, wird es immer dasselbe Bild zeigen, egal wie oft ich das Abspiel wiederhole. Immer ist das Bild zur gleichen Zeit dasselbe!

Anders in meiner Computeranimation (Schnee). In einer Processing Applikation\* wird permanent ein Bild generiert. Es ist zufallsgeneriert und basiert auf Java. So kommt es, dass wir in jedem Moment des Sehens ein anderes Bild sehen. Wenn ich die Animation nach einer Minute stoppe, sehe ich immer ein anderes Bild, nie dasselbe.

Das ist der entscheidende Unterschied zum Video / das erklärt den Aufwand und die Einzigartigkeit des Computerbildes.

Hinzu kommt die unmerkliche Bewegung des Monitors selbst, der ja frei von der Decke hängt.

Durch reine Anschauung vermittelt sich dieser Sachverhalt nicht. Möglicherweise unbewusst.

Meditieren sie: Setzen sie sich so, dass Monitor und der Marienaltar eine Sichtachse bilden. Schauen sie auf den Monitor / schauen sie zu dem dahinter stehenden Marienaltar. Die rastlose Bewegung auf der Stelle / die Ruhe jahrhundertalter Malerei. ...



## \* Processing Code:

```
import ddf.minim.*;
import ddf.minim.analysis.*;
import ddf.minim.effects.*;
import ddf.minim.signals.*;
import ddf.minim.spi.*;
import ddf.minim.ugens.*;

import processing.sound.*;

ArrayList <PVector> p = new ArrayList();

PImage bg;
int y;
import ddf.minim.*;
AudioPlayer player;
Minim minim;//audio context

void setup() {
  size(1920, 1080);

  bg = loadImage("maria.jpg");
  smooth();
  minim = new Minim(this);
  player = minim.loadFile("sample.mp3", 2048);
  player.play();
}
{
}
void stop()
{
  player.close();
  minim.stop();
  super.stop();
}
void draw() {
  background(bg);
  int i = 0;
  float translation = map (mouseX, 0, width, 0,60);
  noStroke();
  while (i < p.size())
  {
    fill (lerpColor (180, 255, map (p.get(i).z, 1, 10, 1, 0)));
    ellipse (p.get(i).x+translation*sin (map (p.get(i).z, 1, 10, 0, PI/2)), p.get(i).y, p.get(i).z, p.get(i).z);
    p.get(i).y = p.get(i).y + map (p.get(i).z, 1, 10, 0.1, 1);
```

```
if ( p.get(i).y > height+p.get(i).z/2) p.remove (i);
else i++;
}
if (p.size () < 1000)
{
float dice = random (100);

if (dice < 50)
{
p.add (new PVector (random (width),-10, random (1,10)));
}
}
}
```