Artem Baguinski

Schoonoord, Netherlands
+31629394010
femistofel@gmail.com
http://www.v2.nl/archive/people/artm-baguinski

*A polyglot-programmer and fast learner, I've spent the last 12 years working in a dynamic environment with cross-disciplinary teams of artists, technologists, scientists and engineers on a wide variety of cultural and art projects involving computer technology - from **custom video players** and **virtual/augmented reality environments** controlled by **sensor grids** to **film sound track editor** for children workshops and **public space audio/video installations** in the city of Rotterdam. Within most projects I am involved in **system design** (hardware, software, network choices), **production** (setting up the hardware, installing and setting up the 3rd party software) and **development** (software development, site-specific customization, maintenance). Working in a small development team, I have to **automate** any non-project specific tasks as much as possible, to be able to quickly start and switch between projects.*

Technical Skills

Like: c++, boost, javascript, ruby, linux

Dislike: windows

Experience

Software Engineer – V2_Lab, Rotterdam, the Netherlands *2000 - Current*

- R&D for technology-rich artistic and cultural projects
  - feasibility research
  - system design: software, hardware
  - programming
  - maintenance
  - production: system installation and set up on exhibition site, modification for a particular set up
- consultancy for artistic and cultural research and projects
  - feasibility assessment
  - technical advise
  - conceptual criticism
- internal software development infrastructure
  - code repositories setup, maintenance, migrations
  - development of internal libraries and frameworks used across projects

Software Engineer – SPONG, Smolensk, Russia *1999 - 2000*

- embedded software programming for industrial devices (power industry, transport)
- R&D:
  - sensor data compression in distributed SCADA systems
  - artificial neural network based fault detection in power plants

Education

Engineer in Computer Technologies – Engineer in Computer Technologies *1997 - 1999*

graduated with honors

B.S. Computer Science – Moscow Power-Engineering Institute *1993 - 1997*

- graduated with honors
- statistical computing support to pediatric research by dr T. V. Kosenkova of Smolensk State Medical Academy.

Projects & Interests

GitHub - ExplodedViews – https://github.com/artm/ExplodedViews *November 2011 - June 2012*
c#, perl, unity3d

The 3D point clouds of popular tourist destinations are reconstructed from flat images mined on the internet and placed in a virtual environment where they are treated as dynamic systems reacting on approach of the visitor.

Programming post point-cloud reconstruction:

- **data preprocessing:** a perl script converting multiple related PLY-format objects into a custom compact binary representation
- **editor tools for working with a massive point cloud database:** custom format clouds import, streaming preview, composition from sub-clouds
- **real-time point cloud streaming:** only the relevant subset of the dataset is loaded, the visible density of the clouds is modulated by current distance from the camera, the point clouds are further divided into smaller chunks that the graphics card can deal with
- **interactive animation**: GLSL/ShaderLab shaders and C# scripts for material animation in response to visitor's activity
- **sensor interfacing**: a laser scanner tracks visitors of a gallery and controls the environment

More info: project description on the lab's site.

GATC / Life (2005) – http://www.v2.nl/archive/works/gatc-life
c, opengl, performer, cave-vr

Interactive virtual reality installation with spatial sound for the opening of I-Space (CAVE-like VR room) at Erasmus Medical Center's I-Space.

- software design
- development infrastructure: desktop development (Linux, cave emulator), I-Space deployment (SGI Irix, I-space)
- programming
  - installation infrastructure (FreeVR / CaveLIb configuration, Performer-based graphics engine, communication with the spatial audio synthesizer over OpenSoundControl)
  - generative graphics / animation: morphing / dynamic environment, non-player character-models

See also: the code repository

Exercise in Immersion IV (2006) – http://www.v2.nl/lab/projects/exercise-in-immersion-4
c++, ogre3d, augmented-reality, virtual-reality, hmd, tracking

An augmented / virtual reality art-game by Marnix de Nijs: chasing semi-intelligent blobs of goo in a mix of actual and virtual environment.

- R&D lead
  - tracking hardware
  - stereo HMD
  - stereo camera
  - sensor data fusion
- software design
- programming
  - build system
  - AR / game engine
  - in-game GUI
  - IO (video-in, tracking data in, stereo rendering, HMD calibration)
  - Augmented Reality: stereo-video backdrop, real-world / virtual-world physics integration
  - NPC-behavior
  - GPU shaders: NPC's non-rigid body physics
  - Tooling: Blender / 3ds Max scripts for level editing, model export, tracking-data integration (for environment calibration).

See also: code repository

BioFame (2011) – https://gitorious.org/biofame
c++, qt, finite-state-machine, computer-vision

A camera toting robot on a circular track pays attention to the exhibition visitors.

- software design
- software implementation:
  - **low-level robot control:** hardware calibration, sensor input, motor control
  - **computer vision:** face detection and tracking with a moving camera
  - **high-level robot behavior:** how to act in different states, how states change, chasing the tracked face
  - **art-installation logic:** what to do with gathered data
  - **robot UI:** knobs and sliders to fine tune the robot
- **display software** (not in this repository): a separate software receiving pictures taken by robot and searching for similar pictures in the database
  - **computer vision:** *detuned* face recognition (misusing 3rd party library)
  - **visualization:** the results are presented on a projection screen

More info: on the lab's site.

GitHub - WatchThatSound – https://github.com/artm/WatchThatSound *June 2011 - September 2012*
c++, qt, portaudio, ffmpeg, cmake

Soundtrack editor for children's workshops by Watch That Sound

- software design
- programming
  - Qt based editor GUI
  - audio in / output
  - multi-track mixer
  - video file import / export
  - graphical soundtrack score - for sketch-like visual composition and as a visual aid during recording
- automation
  - build automation on osx and windows
  - release scripting

More info: Watch That Sound!.

---

max.sick.lms100 (2009) – https://gitorious.org/max-sick-lms100
c++, boost, javascript, boost-spirit

LMS100 LIDAR (a.k.a. "laser scanner") support for Max/MSP programming environment and a device control GUI.

- software design
- implementation
  - communication with the device over **TCP**
  - boost::spirit based **parser for device protocol**
  - Max/MSP **API**: the object is wrapped for use from Max visual programming language
  - LIDAR UI demonstrating the whole API and visualizing scanned environment
  - **computer vision**: background learning and subtraction, blob detection and tracking

More info: Full description on the lab's site.

Max++ (2009) – https://gitorious.org/maxxx
c++, boost, template-meta-programming, preprocessor-meta-program

A template library for writing externals for Max/MSP environment in C++.

The library is written out of frustration with Max's externals (= plugins) C API which requires a lot of redundant typing resulting in hard to maintain libraries. With Max++ the amount of typing is drastically reduced thanks to the C++ templates and preprocessor meta-programming. I'm sole designer and implementor of the library.

GitHub - n0things-app – https://github.com/v2lab/n0things-app *August 2012 - September 2012*
objective-c++, opencv, ios

n0things by Constantijn Smit: an iOS app for abstracting geometric shapes from reality.

- **conceptual design:** brainstorming the artistic concept with the artist and the lab team, advise on computer vision
- **prototyping:** Qt+OpenCV based desktop app demonstrating computer vision and user interaction possibilities
- **software design:** in collaboration with the iOS developer
- **computer-vision:**
  - user-intention guessing object contour selection
  - contour description / conversion to abstract 12-dimensional space

---

GitHub - n0things-ai – https://github.com/v2lab/n0things-ai *August 2012 - September 2012*
ruby, octave, machine-learning, amazon-simpledb, rake, virtualbox

Geometrical shape collection cluster analysis: data processing part of the n0things project by Constantijn Smit.

- **software design**
- **implementation**
  - **cluster analysis** a gnu octave program to perform **k-means clustering** and **cluster silhouettes** calculation
  - **command line interface** in ruby/rake to manually perform various stages of data processing
- **production**
  - a **virtual machine** with **linux**, ruby and octave installed and configured for the artist's convenience.

GitHub - VisionThread – https://github.com/artm/VisionThread *June 2011*
c++, qt, opencv

Hiding video capture / processing thread from the rest of the Qt application

The library was written in response to the increasing amount of computer-vision / augmented-reality based projects in the V2_Lab that required a lot of experimentation and fine tuning (hence the need for integration with a GUI framework). The code was eventually integrated into our Qt+OpenCV application template.

I was the sole designer / implementor of the library.

---

natty (2009) – https://gitorious.org/natty
c++, qt, opengl, tracking

A visualizer of 3D tracking data from custom devices

The software was used internally for debugging V2_Lab's custom 3d tracking sensor grids. It included display of tracked objects and their trajectories in 3D as well as graphs of sensor measurements arrival in time. I was its sole designer and implementor.

GitHub - Osc.cs – https://github.com/artm/Osc.cs *November 2011*
c#

Minimal OpenSoundControl specification implementation in C#

- software design and implementation
- testing against other OSC implementations

GitHub - CastorAndSean – https://github.com/artm/CastorAndSean *December 2011 - January 2012*
c++, cinder, opencv

Face matching experiments

Wrote to practice machine learning algorithms I've learned in an online course

---

GitHub - Cyrus – https://github.com/artm/Cyrus *November 2011*
c#, supercollider, osc, unity3d

Getting sound spectrum via SuperCollider into Unity3D over OSC/UDP

Wrote this for a single time VJing gig and to demo a visualisation technique I've come up with

GitHub - Tetris (2007) – https://github.com/artm/tetris *March 2012*
c++, sdl, opengl

tetris clone

Programed in free-time to learn using SDL + OpenGL.