Alexei Nosenko

Senior software developer

Personal info

Location: Moscow, Russia
Phone: +7 (916) 205-2104
E-mail: alexey.nosenko@gmail.com

Skype: antilopka2011

WWW: http://alexkroll.github.io

GitHub: https://github.com/AlexKroll/GpuGalactica.git

YouTube: https://www.youtube.com/watch?v=bWZGgK3u4pE
https://youtu.be/X4wn4jCeDfM

Summary

A senior programmer with experience in multimedia development. This includes: graphics, sound, video. Small skills in 3D modeling. Minor experience in mobile development. Hobbies in amateur electronics.

Skills

OS: Windows, Android and iOS (little bit).

Programming languages: C++.

Development environment: Microsoft Visual Studio, xCode.

Frameworks, SDK, API: DirectX SDK, HLSL, Android NDK, OpenGL ES 2, GLSL, OpenCL 1.2, OpenAL, 3ds Max SDK, ffmpeg, Socket API (UDP/TCP), BlackMagic DeckLink SDK, NDI SDK. Embarcadero Delphi.

Experience

January 2021-
Present

Ascon https://ascon.ru

Team **Pilot-BIM** https://ascon.ru/products/1297/review

Senior software developer

Software development for building, constructing, etc.
Pilot-BIM is a shared data environment for BIM projects for automatic generation and collaboration with consolidated models.

Development of render (DirectX11) for huge amount of objects (up to 10 millions) in constructing: wall, roof, ground elements, engineering sub-systems, etc.
Optimizations: graphics buffers batching (groups), instancing.
Replace graphic effects to shader and render-state management.
Reduce allocations: memory pools.
Multi thread processing, multi thread rendering (deferred contexts DX11).
Billboarding, text-atlas rendering.
Resolve graphics problems (DX debugging).

October 2017-
January 2021

Jet-Tech Production

Moscow

Senior software developer

Software development for "Match-TV" channel. Sports TV broadcasting.

Making decisions within a small team.
Acting coach. Setting goals, participate in the implementation.
Integration of components into a single system (C ++, Delphi).
Development of the graphic part (DX11).
Migration of software to Vulkan / Unreal Engine (two RnD branches).
Security protection (S.M.A.R.T technology).

Last interesting tasks:
- Compressor (OpenCL, palette+RLE) and real-time GPU decompressor of image sequences.
- External control of broadcast software (Atmel AVR microcontrollers). Receiving signals from different sensors and generate events through COM port.
- Transfer video stream over network using NDI SDK.

Scope of technologies:
Microsoft Visual Studio, DirectX11, BlackMagic Decklink SDK, Arduino IDE, OpenCL GPU-computing.

December 2016-
October 2017

Topcon www.topconpositioning.com

Moscow

Senior programmer

Working with the camera of mobile devices through DirectShow and Windows media foundation.

Modification and troubleshooting the graphic part of the navigation software.
Bringing new ideas to colleagues, realization in limited technical conditions.

Optimization of the graphics engine, migrating the functionality to the DirectX 11 API.

Development of code for displaying big data of for geodetics. Implementation of various elements in the mapping of geodetic maps.

Operation graphics studio "Sport-Image" (TV channel "Russia-2")

Moscow

December 2005-December 2016

Programmer

Graphics development for the sport TV broadcasting.

Development of graphics engine, export tools (3DS Max SDK), TV scene editor.

Programming for the TV signal output and capture (SD/HD) of BlackMagic and BlueFish444 adapters.

Developing utilities: software protection, bitmap fonts editor, installer.

Interaction with programmers of client applications and designers of TV content. Making of technical requirements.

Migrating of graphics engine to x64/DX11 platform.

Developing graphics applications for mobile platforms: Android, iOS.

Network applications: TCP, UDP sockets, client and server.

Video encoding/decoding: ffmpeg for x86 and ARM platforms, H.264.

Continuous maintaining and modification of existing software.

February-March 2015

GLU.

Moscow

Tools programmer

Development tools for game engine: screen saving to file from game render.

Unity3D, C#.

September 2005-December 2005

NIKITA

Moscow

Graphics programmer

Migrating of graphics engine from DX7 to DX9.
Development of graphical utilities for the company: shader linker, particle editor, face morph editor.

Reid 2001

Moscow

April 2002 - February 2005

Game programmer

Development of game projects, related utilities.
Programming of game modules: graphic and sound system, logics, physics and elementary AI.

July 2001 - March 2002

“Bolshevik”, Foreign Investment Public Corporation, (DANONE Group company)

Moscow

Technical support specialist

Solved problems for the users in the company.
Technical support and maintained software and hardware.

June 1997 - July 2001

Army duty service

Rostov-on-Don

Programmer

Development hardware and software for receiving input signals from radio receiver. Development control electronics and software for external devices (by LPT/COM ports).

Development software for the topographic maps.

Education

Higher

1992-1997

Academy for Engineering and Space of A.F.Mozhayskiy, St. Petersburg.

Automated space control systems, Diploma.

Brief about experience.

Multimedia development.

Experience in the following fields: computer graphics, sound, video processing, networking, GPU computations, mobile development, simple design in 3D modeling.

Tools:
Windows: VisualStudio (C++), DirectX 7-11 (Direct3D, DirectSound, DirectInput, DirectShow, DirectCompute). HLSL.

Graphics.
Render geometry, animation (skeletal, vertex, texture), shadows (shadow map), reflections (planar, cube), glow, particle systems.
API: Direct3D / OpenGL ES 2, HLSL / GLSL.
3DS Max. Simple modeling, export data (geometry, animation).

Mobile.
Visual Studio + Android NDK, VisualStudio + Marmalade SDK, iOS (xCode, C++).
OpenGL ES 2.0, GLSL.

GPU computation: DirectCompute, OpenCL.

Sound: DirectSound, Fmod, OpenAL.

Game mechanics: logic behavior of units, simple AI.

Physics: different types of motion of bodies, the collision. Tasks of orientation, aiming.

Networking: video transmission - jpeg images, H.264 stream, UDP protocol.

Multithreading: synchronization primitives (mutex, event, wait, interlocked functions), inter-process exchange (file mapping memory).

Debugging of graphics pipe-line (Visual Studio, PIX).

DirectX experience: DX5-DX11. Direct3D, DirectSound, DirectInput, DirectCompute, DirectShow, HLSL.

CPU architecture: SSE, asm, Microsoft intrinsics.

Delphi Pascal (beginner).

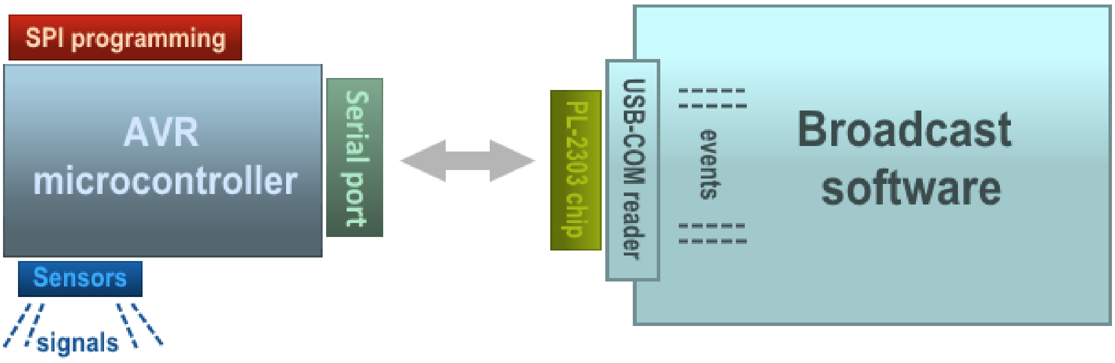
Others: NSIS (installer), BlackMagic DeckLink SDK (video playback/capture),
3DS Max SDK (export geometry and animation) NDI SDK (video playback through network).

Amateur electronics.
Developing and constructing of simple devices based on AVR microcontrollers and ESP8266 chips.

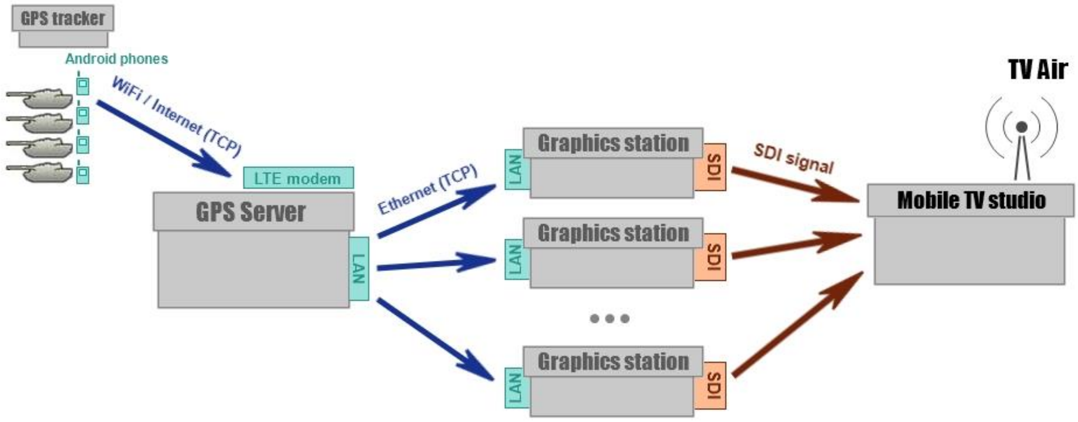
Demos:
https://github.com/AlexKroll/GpuGalactica.git
http://alexkroll.github.io

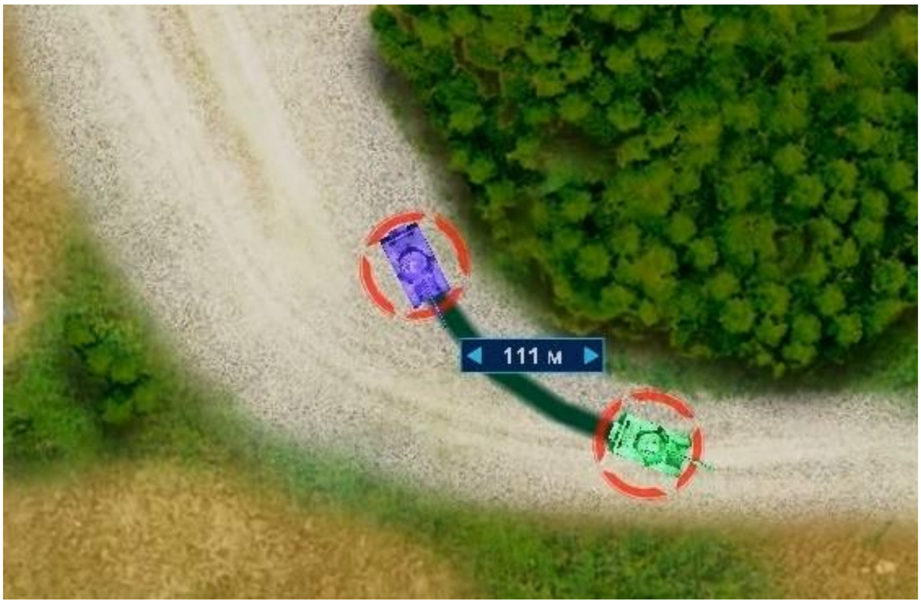
The most interesting projects.

External hardware control of broadcast software.
Self-made device initiates events to software from input signals (optic sensors, temperature sensors, TV dispatcher buttons).
It communicates with PC through Serial-USB-COM pipeline.

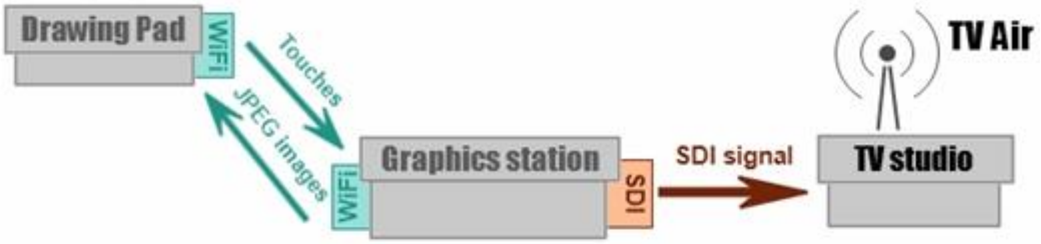


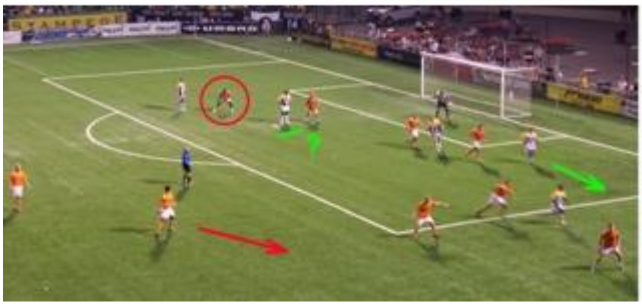
Tank biathlon on Sports TV channel.
GPS navigation of tank units: GPS tracker <-> GPS server <-> TV graphic stations.
Showing situation and tank crews on graphical map.
Win application, Android application (Marmalade SDK), TCP protocol.





Drawing tablet for TV studio (oct.2014).
Analysis of game moments by TV presenters on video replays. Anchorman draws shapes (arrows, lines, rects etc) over the a still image. And describes an alternative situation.
Win app - graphics station (VS2013, DX11, jpeg-turbo API). Android app (Marmalade SDK, OpenGL ES 2.0). UDP protocol.



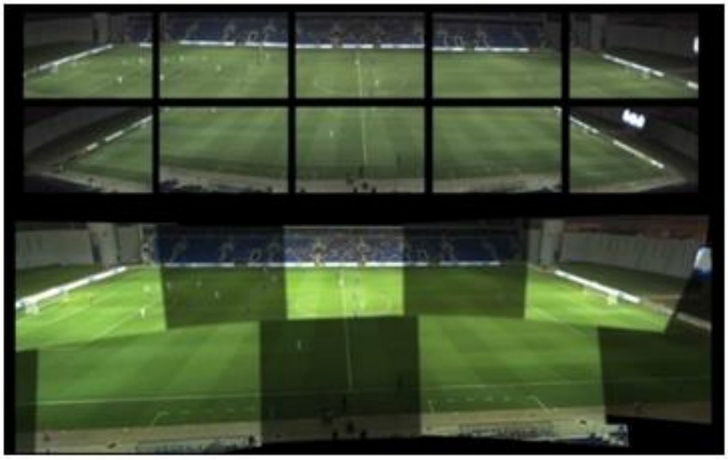


Live sports events on mobile devices (jun.2015).

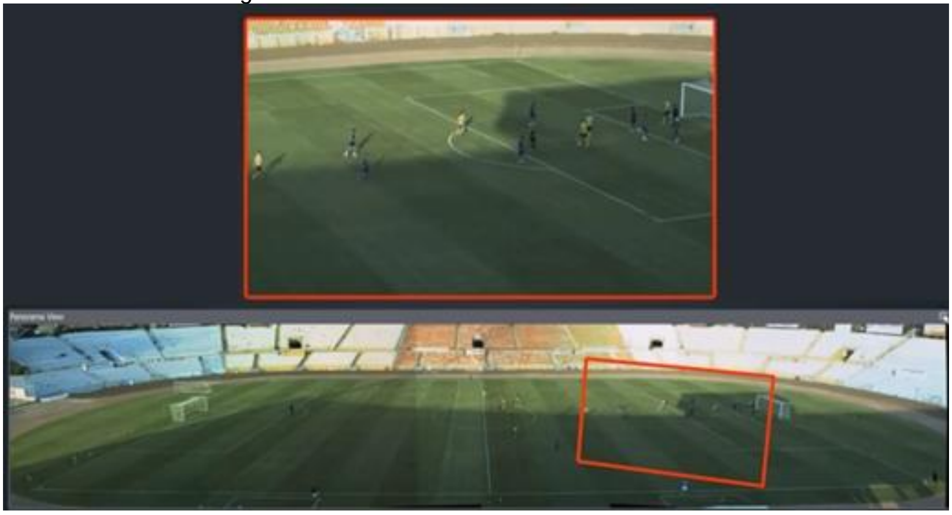
TV broadcasting huge image from several cameras. Images are capturing form cameras and stitching to a huge image.
Then encoding to video stream, sending to server. The server distributes the individual video to mobile devices.
User on mobile device can move and zoom video on huge image of video stream..
This task is analogous to **Pixellot** product (https://www.youtube.com/watch?v=ZnKjglsidUI&rel=0&width=800&height=450).

Win app - graphics station (VS2013, DX11, ffmpeg, jpeg-turbo API, GPU computation).
        - server (VS2013, sockets).
Mobile app - Android, iOS. Marmalade SDK, xCode. OpenGL ES, ffmpeg.
Video stream types: H.264, jpeg, DXT1.

Cameras block. Stitching of captured images..



..Then TV broadcasting:



Substrate of graphics into TV image.

Embedding the graphics into the TV stream using of template image and current image (for example football field before the match as template and current image during the match).

On video sample below: stitching of video stream from to cameras and embedding the graphics under moving cars.

Video: youtube://VirtualGraphics.