Alexander **Lelidis**

Software Engineer · General computer scientist

Bülachstr. 7 c, 8057 Zurich, Switzerland

31.05.1994

(+41) 768286993 | alex.lexus.info@gmail.com | www.alexus37.github.io | alexus37 | alexander-lelidis-445842b6

"There are two major products that come out of Berkeley: LSD and UNIX. We don't believe this to be a coincidence"

Education

ETH Zurich — *Zurich*

Master of computer science — *Sep. 2017 - Sep. 2020*

- Study track general computer science
- Thesis: Invisible to Machine Perception: Attacking Pose Estimators with Attribution Methods (1.3)

Technical University Berlin — *Berlin*

Bachelor of Science in Computer Science (1.9) — *Oct. 2012 - Apr. 2016*

- Exchange program at ETH Zurich
- Thesis: Structure-aware Surface Reconstruction with Sparse Moving Least Squares (1.0)

Werner-von-Siemens Gymnasium — *Berlin*

High school diploma (German Abitur 2.1) — *Mai. 2006 - Mar. 2012*

Skills

Programming	**Python**, **javascript** (ES6), TS, C/**C++**, Ruby, JAVA, VBA, LaTeX, PHP, Assembler (mios, x86), GLSL, Bash, Opal
Databases	SQL, MongoDB, couchDB, pouchDB, postgres
Libaries	AngularJs, Ionic, **React** (Native), Django, tornado, flask, **Redux**, **React**, **HTML5**, **CSS3**, Node.JS, **leaflet**, **mapbox**, ros, bootstrap, jquery, Three.js, loadash, axios, yarn, boost, cgal, igl
General	SVG, openGL, webGL OSM, OSG, mapbox, OSRM, CAD, ESRI AGO, Metabase
Languages	English (Full professional proficiency) , German (native language), Spanish (4 years of high school classes), Bulgarian (basic knowledge)

Experience

ESRI — *Zurich, Switzerland*

Software Engineer — *Spring. 2019 - Fall 2020*

- ArcGIS Urban (React)
- Telemetry gathering with redshift and metabase

Antavi GmbH — *Zurich, Switzerland*

Software Engineer — *Spring. 2016 - Winter 2018*

- Crowd management (angular)
- Development of command, control and communications system in react
- GIS data analysis (leaflet, mapbox)

Undergraduate Research, Dept. of Information Technology and Electrical Engineering — *Zurich, Switzerland*

Research Assistant at IFE — *Fall 2014 - Spring 2016*

- Mobile crowd analysis
- Development of web based software for crowd analysis (js, python)
- Server implementation
- Web design (HTML, CSS)

WeltWeitBau GmbH — *Berlin, Germany*

Software Engineer — *Fall 2012 - Spring 2016*

- Development of software civil engineering informatics (vbs, java, C++)
- Automatisation of software quality assurance (VBA)
- Web design and marketing

Projects

Asteroid field simulation

C++, cgal, OSG

University — *https://alexus37.github.io/asteroidField/*

- Simulating planets and asteroids in space is an intersecting multi-dimensional challenge. Due to the nature of the setup, we have to solve a few hard challenges to achieve a real-time engine. The first part of the problem is the numerical computation of gravitational forces. This problem is today only solved analytically for 2 bodies. Since our objective is to have an asteroid field we using numerical methods to approximated these forces. Once the bodies start to move around the second challenge is to handle collisions and compute physically correct responses. Again this needs to be done in rather fast fashion to be able to run in real-time.

tripTrackr

Ruby on rails, js, OSG

Private — *https://www.triptrackr.de/*

- The motivation behind this project was to understand the full-stack development of an app with including backend. Therefore I choose to create a travel app, where users can create a personal webpage with the travel trajectory and share it with friends.

WebGL interface for the NORI renderer

Python, django, three.js

Private — *http://alexus37.github.io/NoriV2Webinterface/*

- Nori Web interface is a web platform, functioning as a frontend for the Nori Raytracer. It features a user management system, allowing users to save and load scenes, which they can edit in a 3D editor in the browser. Scenes can then be rendered by the platform and will be streamed, piece by piece to the browser. The server uses Django to provide a REST API which is used by an Angular Web App. THREE.js is used for the 3D Editor. The rendered image is streamed to piece by piece via a Websocket.

Thermal augmented reality chess

C++, python, ROS

University — *http://alexus37.github.io/pdf/report.pdf*

- The goal of this project was to create an augmented reality chess game. We used two cameras - an RGB-D camera and a thermal camera. The RGB camera is used to track a paper checkerboard with augmented reality markers which are used to estimate the pose of the camera. The video with the resulting camera matrix are used by OpenGL to augment the video with the virtual game objects. We use a thermal camera for the detection of the user input.

Research

Invisible to Machine Perception: Attacking Pose Estimators with Attribution Methods

Zurich, Switzerland

Master thesis — *Mai. 2020*

- Neural networks are currently the most accurate techniques to tackle computer vision problems. However, with the increase of accuracy, an increase in complexity has emerged leading to black- box systems. This is especially problematic in safety-critical situations. In this thesis, we are using various attribution methods in 2D or 3D to understand a human pose estimation model. During this process, we develop a new method for the 3D attribution case, called 3D Saliency map. To test the robustness of this model we identify adversarial examples in 2D and propose a new method in the 3D domain for computing adversarial texture for clothing. We show that our approach works by rendering a video of simulated meshes with and without the adversarial texture and feeding into the human pose estimator. To quantify the results we develop three different metrics, measuring various aspects of the attacks.

Geometry Representations for Big Geometry Data with Unsupervised Feature Learning

Hong Kong, China

Big Data and Smart Computing (BigComp), 2016 International Conference — *Jan. 2016*

- In this paper, we present an exploration of analyzing geometries via learning local geometry features. After extracting local geometry patches, we parameterize each patch geometry by a radial basis function based interpolation. We use the resulting coefficients as discrete representations of the patches. These are then fed into feature learning algorithms to extract the dominant components explaining the overall patch database. This simple approach allows us to handle general representations such as point clouds or meshes with noises, outliers, and missing data. We present features learned on several patch databases, highlighting the utility of such an analysis for geometry processing applications.

Structure-aware Surface Reconstruction with Sparse Moving Least Squares

Zurich, Switzerland

Bachelor thesis — *Aug. 2015*

- Reconstructing the surface underlying a given point cloud is a fundamental problem in geometry processing. Moving least squares solves this problem efficiently using local fits. However, locality comes at the expense of losing a global view of the geometry, leading to inferior results when there is missing data or significant amount of noise or outliers. Global methods are more robust, but they are expensive to compute. In this thesis, we will combine global and local methods in an efficient manner by using learned local geometry bases and sparse moving least squares fits.