Qingguo Xu

CONTACT INFORMATION	6515 W 87th St. Los Angeles, CA 90045	626-491-5239 qingguo.xu88@gmail.com https://qingguo-xu.com

PERSONAL STATEMENT

As a full-stack software/research engineer, my interests and expertise are in the area of image processing, 3D geometry processing, physically-based rendering, computational photography, and operating system. I have experience and skills in software/app development (iOS & Android), server management, deep learning, modern rendering pipeline, computer vision, and graphics related research programming.

WORK EXPERIENCE

Pinscreen Inc.   Dec 2017 - present

During the time at Pinscreen, I contributed to both products and research projects, selected projects as follows:

- Pinscreen mobile app development.
- Server management on Amazon Web Service (AWS)
- WebRTC streaming
- Unity Rendering (Server side & offline rendering)
- Pinscreen 2D face benchmark database for landmarks detection
- Pinscreen 3D face database preparation
- Pinscreen lifelike avatar creation
- Normalized Avatar Synthesis using StyleGAN

University of Kentucky

Before graduation from University of Kentucky, I contributed to several research projects including my master thesis and developed 5 applications for different departments (labs).

- **Master thesis**: 3D Body Tracking using Deep Learning
- **Mask-off**: Synthesizing Face Images in the Presence of Head-mounted Displays
- **Develeopped applications**: LittleHelper (Google glass app), VFE (iOS), eCROPS (iOS & Android), KTDID (Windows), JTK (Windows)

SKILLS

- Programming languages: C/C++, Python, Java, Matlab, NodeJS, Objective-C, C#, PHP, HTML/CSS
- Commercial softwares: Unity, Unreal, Maya, Wrap3, Agisoft, Nuke
- Expertise skills: Computer vision, Deep learning, Mobile app development (iOS & Android), Amazon Web Service (AWS), WebRTC streaming, VFX composition, Web development

EDUCATION	Master, Computer Science	Sep 2015 - May 2017

University of Kentucky
- Thesis Topic: 3D Body Tracking using Deep Learning
- Advisors: Ruigang Yang

PEIK program, Electrical and Computer Engineering   Sep 2013 - May 2015
University of Kentucky
- Power and Energy Institute of Kentucky (PEIK) certification

Visiting student   May 2012 - May 2013
Institute of Computing Technology Chinese Academy of Sciences
- Advisors: Jianfeng Zhan
- Topic: Software aging

Master candidate, Computer Science   Sep 2011 - 2012 May
Xi'an Jiaotong University
- Advisors: Yong Qi
- Topic: Software aging

Bachelor, Computer Science   Sep 2007 - May 2011
Xi'an Jiaotong University

AWARDS	**Teaching Assistant**	Sep 2016 - Dec 2016
	Kentucky Opportunity Fellowship	July 2015 - June 2016
	Teaching Assistant	Sep 2014 - June 2015
	PEIK Tuition Scholarship	Sep 2013 - June 2014

PUBLICATIONS

1. Huiwen Luo, Koki Nagano, Hanwei Kung, **Qingguo Xu**, Zejian Wang et al. "Normalized Avatar Synthesis Using StyleGAN and Perceptual Refinement" 2021 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

2. Yajie Zhao, **Qingguo Xu**, Weikai Chen, Chao Du, Jun Xing, Xinyu Huang, Ruigang Yang. "Mask-off: Synthesizing Face Images in the Presence of Head-mounted Displays." 2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR).

3. **Qingguo Xu**, Sen-ching Samson Cheung, et al. "Littlehelper: An Augmented Reality Glass Application to Assist Individuals with Autism in Job Interview". Asia Pacific Signal and Information Processing Association (APSIPA), December 2015

4. Pengfei Zheng, **Qingguo Xu**, Yong Qi. "An Advanced Methodology for Measuring and characterizing Software Aging". International Workshop on Software Aging and Rejuvenation (WoSAR), 2012

SELECTED
PROJECTS

Pinscreen mobile app development
- **Intro**: Pinscreen is an mobile app that takes a frontal selfie as input and generates a 3D virtual avatar.
- **Role**: I implemented user registration and password management, including server side code. I also compiled the Android version app and fixed bugs for Android version. I ran stress test for the app and fixed some bugs.

Pinscreen 2D face landmarks benchmark database
- **Intro**: The 2D face database is used for training face tracker and facial landmark detection.
- **Role**: I downloaded public face data and trained a robust face tracker. I first ran a general landmark detector over all the images and got 1st version result with many failure cases. Then added correct results and partial fixed failure images to training,

which can get a better and more accurate detector. By repeating this process, I trained a robust and accurate face detector.

Pinscreen 3D face model database
- **Intro**: 3D face model database is used for building 3D morphable model and avatar creation.
- **Role**: I processed all the raw scan data and wraped faces into same mesh topology. I built an automatic pipeline to wrap face model and built textures.

Pinscreen lifelike avatar creation
- **Intro**: Pinscreen builds lifelike avatars with Deepfake technology. By using Deepfake, I can map a real person's face to a CG character and drive the real person with the CG character.
- **Role**: I built an automatic pipeline for deepfake and generated most Pinscreen's Deepfake results. For all the Deepfake projects, I worked on all procedures: processing data, training models, inferencing raw results, and compositing final results with alpha blending.

Normalized Avatar systhesis using StyleGAN
- **Intro**: This is a research project, and the system takes a single frontal face as input and generates a normalized avatar using StyleGAN.
- **Role**: I prepared the training data (3D meshes and textures), and ran some comparison experiments.

Master Thesis: 3D body tracking using deep learning
- **Intro**: This is a research project and at that moment, there is no other similar works. The system uses Kinect as input capture device and can track 3D body movements almost in realtime (10 fps).
- **Role**: I built the whole system and captured data with Kinect. With Kinect, I can get 2D images and aligned 3D depth at the same time. I first run body tracking on images and get joints' position on 2D, since 3D depth is aligned with 2D image and the camera parameters are pre-calibrated, I can map 2D position into 3D space, that's how I can track body movement in 3D with deep learning.

Mask-off: Synthesizing Face Images in the Presence of Head-mounted Displays
- **Intro**: This is a research project, and the system can recover photoreal face images from faces wearing head-mounted displays.
- **Role**: I built the whole hardware, synchronized all 3 cameras (one for frontal view and two for each eye), helped capture data and ran some experiments.

Littlehelper: Using Google Glass to Assist Individuals with Autism in Job Interviews
- **Intro**: This is research project and LittleHelper takes advantage of Google glass, which has a camera and a very small display. The app can do simple face tracking and find where the face is, then it will show an arrow pointing to the detected face and notice the individuals with autism which direction to look at. It can also calculate the sound volume, if the individuals speak too loud, it can notice him/her speak lower, and vice versa.
- **Role**: I implemented the Google glass (Android) app and did some experiments in an office room. After many experiments, I pre-defined some thresholds for "too loud" and "too soft".

Software app development: VFE, eCROPS, KTDID, JTK
- **Intro**: These 4 software application are for different labs in University of Kentucky. **VFE** is short for Voice Function Exercise and it's an iOS app that help individuals with voice disorders do exercises at home and improve their voice functions. This app is used by patients in UK clinical voice center; **eCROPS** is short for electrical Cost-effective Roll-Over Protective Structures, which is an app that teaches high school students about truck safety. This app is used by thousands of students in 20+ high schools in 6 states; **KTDID** is a windows software that used by Kentucky Transportation Center (KTC); **JTK** is a windows software and uses the Jonckheere-Terpstra-Kendall (JTK) algorithm to analyze raw temperature data and figure out the amplitude and period of

the temperature trend automatically

- **Role**: For VFE (iOS) and eCROPS (iOS & Android), I implemented the app and the sever side management all by myself. For KTC, I implemented and upgraded 3 out of 6 main components in C#. For JTK, I implemented the simple GUI and the JTK algorithm in Matlab.