Beomsu Kim

Los Angeles *bek001@ucsd.edu* *469-818-4342* *Instagram* *Linkedin*

Skills

Computer Graphics Machine Learning Software Engineering Computer Security C/C++ Java Python System Verilog Vulkan OpenGL RenderDoc Lua Dear ImGui PyTorch TensorFlow Google Colab AWS Sagemaker Transformer GANs Firebase Spring JQuery

Education

University of California San Diego Sep.2019-Jun.2023
BS CSE: Computer Engineering
ROK National Forestry Cooperative Federation Scholarship Dec.2022-Jun.2023

Experience

Junior Developer

- Spring • Java • SQL • MVC • JQuery • Deep Learning • PyTorch • Software Engineering

Fountain Valley **Solomon America** Feb.2024-Now

- Managing and developing solutions for Hyundai and Kia's product quality management system(PQMS) using JQuery, Oracle SQL, and Spring MVC structure on Tomcat
- Collaborating with the senior member to reconstruct a webapp version of the solution
- Preparing to participate in a machine learning project with Hyundai Glovis

Software Engineer

- HTML • ComputerVision • Machine Learning • Deep Learning • GANs • PyTorch
- Software Engineering

Los Angeles **Mitaa** Sep.2023-Feb.2024

- Built a website for memory clinic
- Managed patient dataset to use SSD to analyze medical images to catch Alzheimer's disease
- Built an appointment management system with **Google Firebase**

Machine Learning Engineer Intern

- Deep Learning • Machine Learning • TensorFlow • PyTorch • Computer Vision
- Image Augmentation • YOLO

Busan, South Korea **AT Solution** Apr.2022-Jan.2023

- **TensorFlowed** for data engineering at a commercial industry field using **YOLO**
- Generated a model to detect a pattern recognition in an image dataset
- Served as an exhibitor for the company’s booth at CES 2023 in Las Vegas and Collaborate with the team members to convince the potential customer of the products

Computer Emergency Response Team

- Powershell • SQL • Computer Security

Incheon, South Korea **ROK Marine Corps** Oct.2020-Apr.2022

- Managed military vaccine servers for security updates using linux
- Drilled for a possible security breach situation including hacking mail, ransomware training
- **Powershelled** routine or repetitive works for efficiency using **REST APIs**
- Allocated IPs and controlled access to the networks for security

Projects

vktutorial beta version

- Vulkan • Computer Graphics

Jan.2024-Feb.2024

- Dynamic Rendering for code simplicity and Interactive camera
- RenderDoc for Debugging
- Abstraction based on https://vkguide.dev

Dive into Deep Learning

- Machine Learning • Python • PyTorch • TensorFlow • MxNet • d2l

Jul.2023-Oct.2023

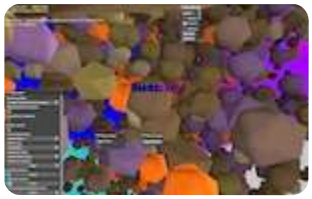
- Implemented the concept of machine learing following the guide in *Dive into Machine Learning*
- **Recommender System**, **GANs**, **Transformer**, **Deep FM**
- Colabed for producing immediate output and feed back

Particle System with subdivision

- Computer Graphics • C/C++ • Multithread • The Forge • Vulkan

Aug.2023-Sep.2023

- Advanced GPU rendering based on TF engine(**AAA**) **Vulkan** and **DirectX** pipeline simulating particle collision responses
- Instanced, indirect, and indirect with compute **shader rendering options** to compare the performance between them
- **LOD** strategy used for performance acceleration

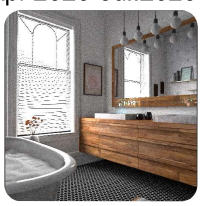


Ray Tracing and Tone Mapping

- Computer Graphics • C/C++ • Multithread • lldb

Apr.2023-Jul.2023

- Ray tracing renderer with **BVH** and multithread acceleration
- **Importance sampled** for area lights and used GoogleTest
- Several **Tone Mapping** strategies are used



ETA Predictor

- Machine Learning • Python • Colab • Deep Learning • PyTorch

Apr.2023-May.2023

- Feature engineered based on traffic information of the traffic data set in **Kaggle**
- Used **d2l** library referencing the book Dive into Deep Learning to solve real world problems by analyzing the pattern
- Planning to leverage error function than sigmoid to be robust against outliers

Sonette Predictor

- Machine Learning • Python • Colab • Deep Learning • LLM

Apr.2023-May.2023

- Given the data set of Shakespearean sonnets and the first 40 characters of hint, generated what will be following the seed
- Generated sequences using **attention** model to solve the natural language processing problem
- Used d2l module on **PyTorch**, following the Dive into Deep learning tutorial