

Tim (Tetsuya) Matsumoto

✉ t.matsumoto@alumni.ubc.ca • 🌐 github.com/TimMatsumoto

Education

University of British Columbia

Bachelor of Science, Honours in Mathematics with minor in Computer Science

September 2016 - Ongoing

Expected May 2020 | Vancouver, B.C, Canada | Major GPA 3.90/4.33

Stanford Pre-Collegiate University-Level Online Math & Physics

Student of Mathematics

August 2015 - January 2017

Skills

Programming Languages: Python, C++, Matlab, Java, C#, TeX, XML

Industry Skills: Git, OpenGL, Unity, IDE tools and libraries.

Academic Knowledge: Machine Learning/Deep Learning, 3D graphics, Linear Algebra, Numerical Operations, Differential Equation, Differential Geometry

Projects

4D SLASH

C#, Unity/Visual Studio

Unity3D

August 2018 - Ongoing

A 3D game in Unity3D. A player moves along a predefined trajectory in 3D space avoiding obstacles on the way. The player also has a blade to slay and cut the obstacles. Some obstacles are thrown targeted at the player, and the player cuts the obstacle to avoid crashing. There are also objects from which the player gets some kind of rewards. As extra features, I may add more assets to play with such as guns, beams Japanese sword and so on. The trajectories will be implemented in C++ and combined into C# file.

Stock Price Prediction

Python/keras

Deep Learning

August 2018

Predict Google stock price for next 60 business days from past data using RNN with keras library. Construct a data structure from data set from Google homepage into a csv file where k_{th} row represents the stock prices from the day to 20 days prior to the day. Then, Recurrent Neural Network (RNN) is applied to learn the data set, and the result is plotted with the actual changes in the stock price for comparison. This is implemented in Python using Keras library.

Movie Recommendation

Python/Pytorch

Deep Learning

August 2018

A movie recommendation system using Restricted Boltzmann Machine (RBM) with Pytorch library. Algorithm 2 from "An Introduction to Restricted Boltzmann Machines" by A. Fischer and C. Igel is implemented using Pytorch library. Although RBM is a unsupervised learning algorithm, the data set was divided into training set and test set to test the RBM classifier. The final result was approximately 75 % in accuracy.

Blurring Tree

C++

Algorithms and Data Structures

May - June 2018

A tree is constructed from an image to blur. In prior to tree construction, for every pixel, the average and variance of RGB colours in the rectangle where the upper left corner is (0,0) of the image and lower right corner is the pixel. Then, The tree splits either horizontally or vertically based on the statistics. This process continues until the leaf node corresponds to each pixel. By pruning this tree, we get a pruned image.