Graphics: THREE JS: Defining the Scene – explain camera frustum

Machine Learning: Add something on NPM web package

Methods: Machine Learning: CNN: Add maths on convolution with a kernel, explain the way it works; Maths of convolution for example how to calculate the dimension of activation maps:

<https://www.saama.com/blog/different-kinds-convolutional-filters/> Useful explanation of how it works

Background: Add papers on usage on CNN and Computer vision, such as the MNIST Dataset etc.

Source for maxpooling

https://medium.com/technologymadeeasy/the-best-explanation-of-convolutional-neural-networks-on-the-internet-fbb8b1ad5df8

Outline:

Intro, abstract.

Background – 1. SLE; 2. Existing Systems; 3. NLP; 4. CNN – usage etc.

Methods – 1. Overview of the entire architecture; 2. Graphics; 3. Machine Learning

Graphics – scene, Phong shading etc.

Machine Learning – data pre-processing, briefly on networks used.

Implementation – 1. Graphics System; 2. Machine Learning System; 3. Feedback System

Graphics System – Lib, Scene, Load model, GLSL, linking, rendering, training data

Machine Learning System –

TensorFlow;

MLP: how it is built + network diagram

CNN: network - Important concepts: convolution, layers explained – why we choose it

Multi-label CNN: Difference in network

Concepts + diagrams + implementation. Cross-validation

Hyper-parameters tuning

Feedback System – List out the classes/feedback 2pg

Evaluation – 5pg

Future Works – 5pg

Diagrams (10 pages):

Methods:

Coordinate sys

Space transformation

4 grid phong shading

Specular term pictures

Training data max 3 pages

ML – MLP

ML – CNN

AutoTeacher – architecture

UI – figure

Graphics pipeline

Graphics System:

Shader lab framework

Custom built graphics system diagram

Camera frustum

Lighting position

Machine Learning:

Diagram for my networks – MLP, CNN, MultiLabel CNN 1pg