Iconnissance

Computer Vision

Image recognition system for video game items

Henry Fan

Introduction

Objective, motivation, and approach

Objective

Icons!

Icons are used all the time in video games, graphic interfaces, etc, to represent objects and items.

Objective

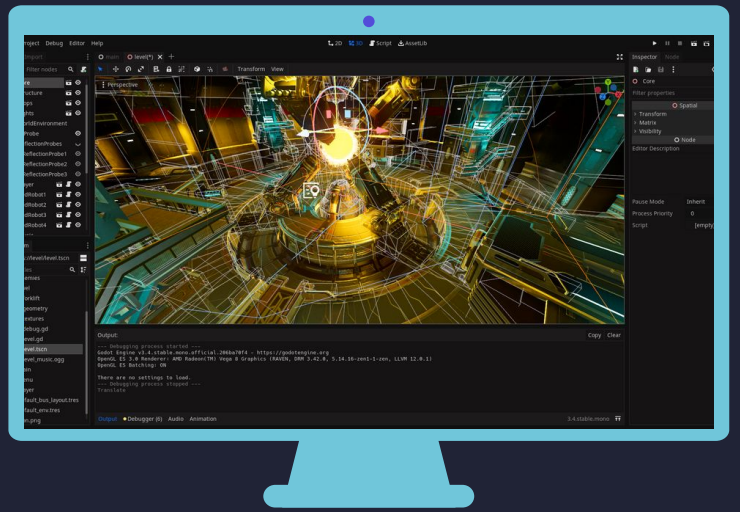
Recognize

Small 2D square icons that represent common items in video games.

INPUT	OUTPUT
	“Hammer”
	“Ring”

Game Development

- Not just coding
- Involves art, sound, storytelling, etc.
- Not everyone can do all of that!



Approach to doing this?

Real world concepts represented by icons

- Vary heavily in shape and orientation
- Vary heavily in color and texture
- Recognition techniques such as hough transform will not work for many cases!

Approach to doing this?

CNNs!

- Adaptable to classes of different icons.
- Already have a dataset of icons to train neural networks on.

Goals

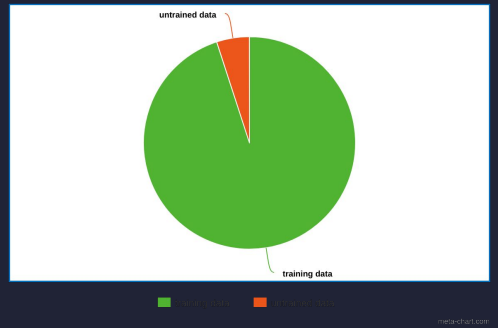
Benchmarks and outcomes

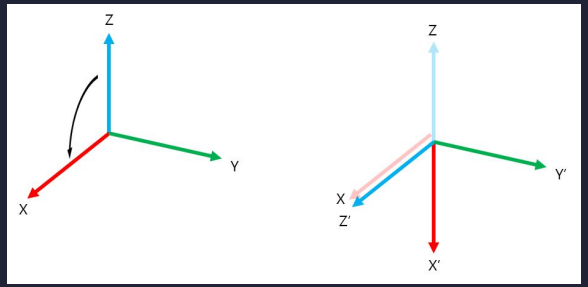
Benchmarks

Accuracy

- To test the accuracy and ability of the model to categorize icons, a portion of the dataset will be reserved for testing the model performance on untrained data.

- Unique images in the training set will be augmented with random transformations to improve performance with a small dataset.





Categorization Goals

- Identify common video game objects for the following 12 categories:
  ['boots', 'axes', 'shields', 'helmets', 'ingot', 'potions', 'pants', 'swords', 'gloves', 'armors', 'rings', 'gems']

- **Reach goal: be able to identify objects with very different art styles from training data, such as *Minecraft***

Resources

A list of resources used for this project

Information:

- https://datagen.tech/guides/image-classification/python/#
- https://www.geeksforgeeks.org/python-image-classification-using-keras/

Libraries:

- Tensorflow: https://www.tensorflow.org/tutorials/images/classification
- Keras: https://keras.io/
- Numpy: https://numpy.org/
- Matplotlib: https://matplotlib.org/
- Pathlib: https://docs.python.org/3/library/pathlib.html

The end!