CalPolyPomona

F23 CS4210.02 Machine Learning and Its Applications

A Supervised-Learning Facial Expression Classification Convuluted Neural Network Model

Research: Alex Mariano; Dr. Hao Ji

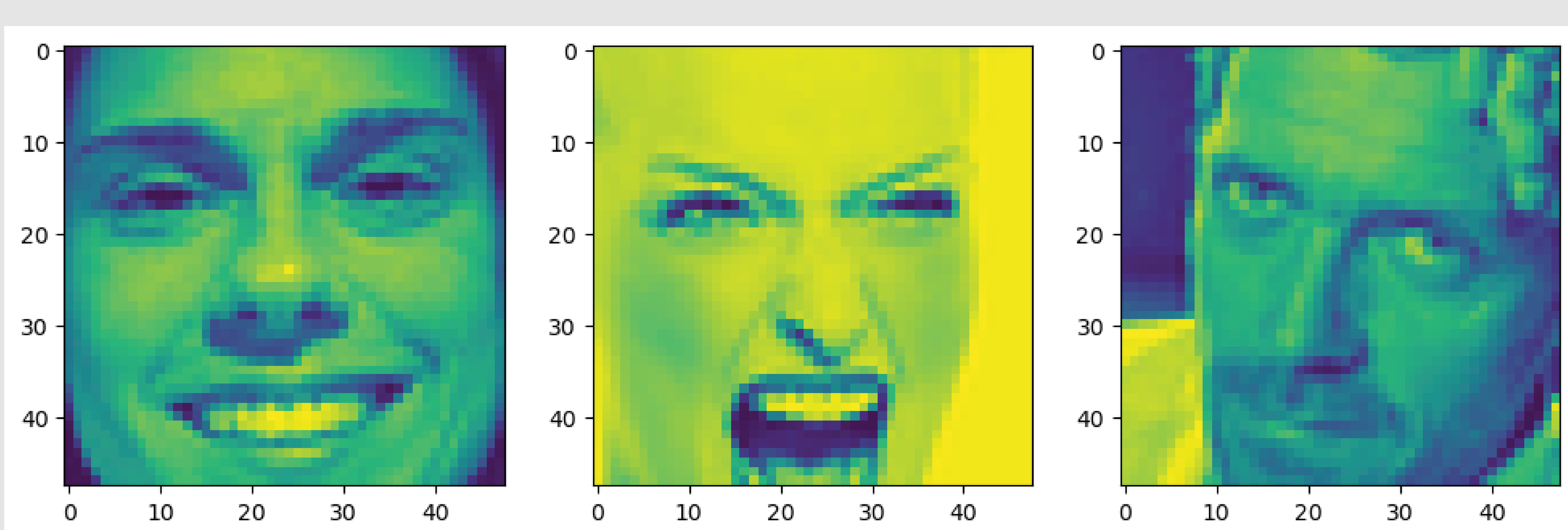
Strategic Focus Area: Supervised Learning

Abstract

Using a convuluted neural network design, a facial expression classifcation model is created to classify a given image of a face into 3 different classes: Happy, Angry, or Neutral.

Background

The goal of this project was to create a supervised machine learning model to classify a set of input facial images into 3 classes: Happy, Angry, or Neutral. Each of the dataset is a headshot of a person in any of these 3 states. The age of individuals in the input dataset vary, including infants and the elderly. The input dataset also consists of facial expressions where the individual is not directly looking into the camera, or their overall head may be tilted slightly.



A Convoluted Neural Network (CNN) design was chosen to do the facial expression classification. This is because CNNs are generally more adaptable to recognizing patterns that are smaller than inputs. This should help detect small features, such as mouth or eyebrow expressions. Additionally, when combined with pooling, the CNN saves computational time.

Results

The model is trained and the best scoring based on epoch is saved to a master model list. Training is generally stopped when the best score does not increase for a set amoutn of time, or the 1000 epochs have elapsed. This master model is then reloaded and tested with the final testing dataset.

Training and Validation Dataset Score: 0.908501
Kaggle Competition Score: 0.79546

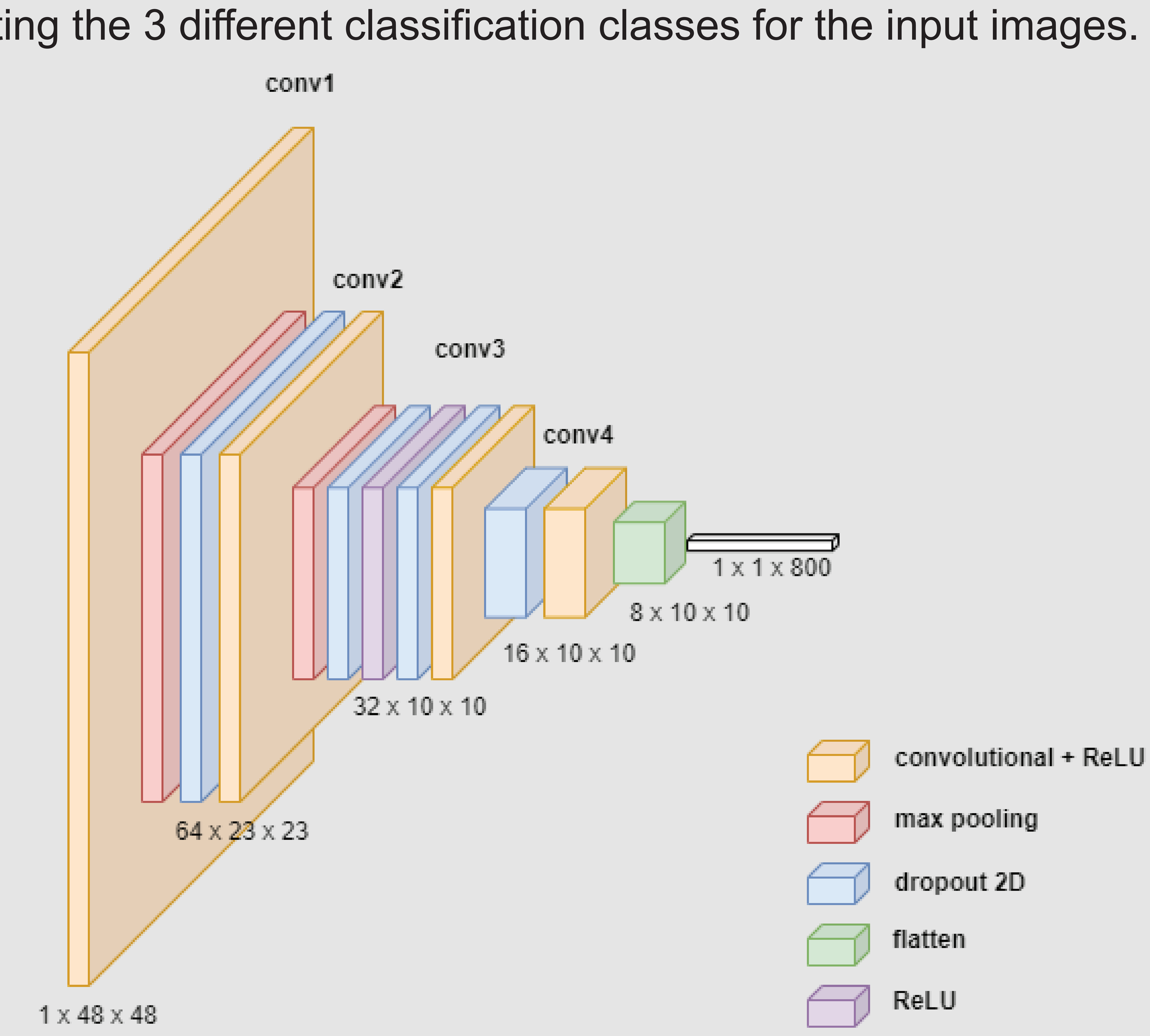
These results showcase that this model can accurately identify facial expressions up to this degree. Of course, this is limited to a subset of data where the facial expression is clear and mostly in-frame. However, additional pre-processing can help mitigate minor issues like these, which would allow the model to function most accurately.

California State Polytechnic University, Pomona
3801 West Temple Ave, Pomona, CA 91768

www.cpp.edu

Methodology

The model receives input through a 1 channel 48 x 48 pixel image. The model uses 4 convolutions to make its prediction. Convolusions are partnered with a rectified linear unit (ReLU) activation function, with some surrounding dropout functions to aid in the approach. Each convolution halfs the input channel. The first and second convolutions use a kernel size of 5 x 5, while the remaining convolusions use a kernel size of 3 x 3. The remaining input is a 1 x 1 x 800, channel, desinged to output a 0, 1 or 2, representing the 3 different classification classes for the input images.



The model was trained using a 80 - 20 split of training data and validation data constisting of a total of 16175 images. The loss function used for the training loop is a standard pytorch cross-entropy loss function. The model was trained for 1000 epochs with a shuffled batch size of 64 images. Preprocessing was also done on the training dataset, such as flipping or rotating the images to provide some variety into the pool.

Additional training was done on multiple machines with various graphics cards to utilize pytorch and cuda. The primary training unit is a custom personal computer constisting of an i7-11700k Intel processor and an NVIDIA RTX 2080 GPU. The mobile unit was a dell precision laptop, using an i7-12700H Intel CPU and a NVIDIA RTX A2000 8GB GPU. Both computers run Ubuntu Linux 20.04.

Conclusion

The goal of this project was also to serve as a practice in understanding machine learning concepts and convuluted neural networks. Training methodology and fine tuning parameters is overall complicated and an interesting learning experience. The difference in the training and validation score and the kaggle competition score demonstrate familiar concepts of machine learning. It may be easy to train the model, but in real world scenarios, it may be difficult to achieve the accuracy given an unfamiliar set of data.