M10 Project
Kaggle: I'm Something of a Painter Myself

Johnathan Wang, Nicholas Charles Vitellaro, Makenna Shae Owens, and Marc Mahanna
COSC523: Artificial Intelligence, December 3, 2024

Abstract - **This project focuses on using machine learning (ML) and generative GAN approaches to implement models classifying images of handwritten numerical digits. The format of this project is that of a Kaggle competition** [1]**. Goals of this project are to 1) implement a baseline score and 2) try to improve on the baseline score.**

DESIGN CHOICES AND IMPLEMENTATION

The approach selected was to implement a baseline and an improved model intended to improve score beyond the baseline. CycleGAN is the default implementation and is provided in the Kaggle competition tutorial [2]. CycleGAN is also the improved model providing the best performance after much team experimentation. For each experiment and subsequent baseline and improved implementation, data was loaded, data was cleaned, features were explored and selected, models were trained, and test results were produced.

The baseline model followed the Kaggle tutorial. Minimal adaptation was required to get the baseline to execute. Data cleaning and preparation was minimal. Execution was on Kaggle using GPU accelerators. 25 epochs were used to generate the results. Resulting performance when submitted to Kaggle was 57.12277%

The improved model was the result of much team experimentation. One approach changed data preprocessing, scaling, resizing, and normalization but did not improve results. This included a pre-trained CycleGAN which was briefly evaluated but deemed outside the scope of this project. The selected approach was evaluation continued based off the baseline model through parameter exploration. Due to limits in Kaggle submission approach for this project, the team parallel submitted a few permutations of which a few parameters caused Kaggle to timeout or crash. Successful permutations including a noticeable shift in performance that ran to completion on Kaggle were epochs which were decreased to 12 (67.17586%) and increased from 25 to 50 (54.7767%) and 100 (46.99433%). The CycleGAN composition was left "stock" as was each of the generators and discriminators. Resulting best performance when submitted to Kaggle was 67.175.86%.

CHALLENGES AND OBSTACLES

The project group members are familiar with similar machine learning concepts and have some prior experience through coursework and other projects with similar model types.

For this assignment, Kaggle runtimes were used. While The baseline notebook would execute via a CPU only runtime, execution time was lengthy, so accelerators were evaluated. The TPU accelerator option was not readily available due to demand but resulted in repeatable crashes on Kaggle when used. The P100 provided a runtime of about 37 minutes at 74 seconds per epoch (25 epochs, used for baseline submission). The dual T4 was not evaluated as the P100 was successful and generally available (team has experience with multi T4 configurations, this is a good second alternative). Kaggle submissions on this project appear to only accept versioned notebooks in Kaggle and are limited to 5 submissions per day versus prior challenges having 10 submissions per day greatly reducing the opportunity for experimentation.

The baseline tutorial Jupyter notebook started with medium high performance out of the box which proved a challenge to improve upon. Increased epochs showed a tendency to overfit and other parameters showed little promise due to runtime errors on Kaggle. Due to the nature of the competition and limitations on submissions per day, most days' submission count was tied up with evaluation of different parameter combinations.

DISCUSSION AND FUTURE WORK

The team found similarity between Computer Vision (CV) techniques such as image pyramid based blending and morphological operations and this project. The team believes that combining computer vision with machine learning in the form of GAN models is an incredibly powerful approach affording image processing operations in near real-world applications. Application of CV and models is different than other ML approaches requiring different techniques. Additionally, the team did find it true in the case of this project versus prior projects, most real-world CV applications generally require greater CPU (and GPU) resources and runtime versus other ML models.

This project would be well served with a local runtime allowing debugging and experimentation, the benefit would greatly be outweighed by the detriment of having to move a large image set submission back into Kaggle. The team is under the assumption that the CPU and GPU burn for each submission must be more cost effective versus the several hundred-megabyte upload (or multi-gigabyte upload) cost. The team would be interested in knowing the performance difference between P100, T4, and other local runtimes such as MLX on Apple Silicon.

The team attempted several approaches to the application of these concepts through experimentation and learned a few things that worked and many that did not improve overall performance beyond the already performant baseline.

SUMMARY

This project provided the opportunity to compare the performance of different machine learning and GAN models within the context of CV blending and morphological operations. A consumable data set and a challenge issued in the form of a competition provided an enriching and rewarding experience for the project group. Processing challenges limited project exploration, but these were partially mitigated through brute force parallel submissions to Kaggle.

REFERENCES

[1]   "I'm Something of a Painter Myself," [Online]. Available: https://www.kaggle.com/competitions/gan-getting-started.
[2]   "Monet CycleGAN Tutorial" [Online]. Available: https://www.kaggle.com/code/amyjang/monet-cyclegan-tutorial.