Parallel Massive Dataset Cleaning

Jianmeng Yu

Supervised by: Bob Fisher

Project Motivation

- What is Motivation of the Project?
  - In 2015, Pugh[1] developed a cleaning algorithm to remove False Positives from a 1.6 TB dataset.
  - However, the cleaning was not applied due to the time cost (25,000 hours on a 40-core machine)[2].
- The Goal of this project?
  - The Goal is to apply this cleaning algorithm on the 1.6 TB dataset, reducing the false detections, hence it's size.
  - Sub-Goals including translating the algorithm into Python, developing parallel frameworks, and evaluating the cleanliness.

Background - Dataset

- The Fish4Knowledge (F4K) project[3] collected 5 years of recording at underwater coral reef areas in Taiwan.
- A species recognition algorithm were used, and the dataset is reduced to a size of 1.6 TB, containing 839 million detections.
- However, 60% of the reduced dataset are False Positives (Non-Fish detections recognized as Fish).

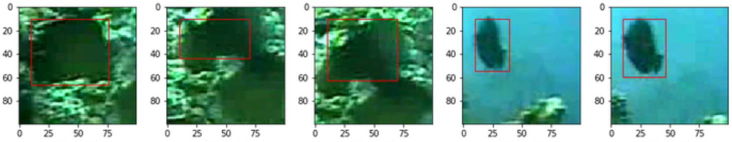


Figure: Sample frames of the dataset

Background - Cleaning Algorithm

- The project uses the cleaning algorithm, the Pipeline Classifier developed by Matthew Pugh[1].
- The majority of the project's challenge came from the translation and optimization of the pipeline, as this classifier is still at experimental stage.
- For example, the SQL server is not used due to the limit of resources. And the CNN part is abandoned due to mistakes during previous training stages.

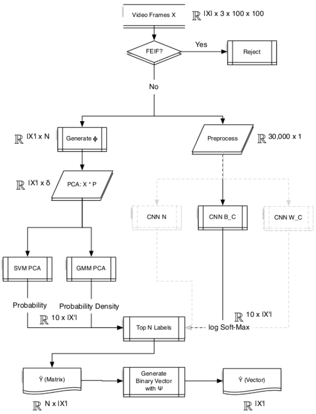


Figure: Pipeline Classifier

Project Outcomes

The project has made following contribution on cleaning the dataset:

- Using standard I/O stream script to extract data from SQL dump file, removes the cost of maintain a server.
- Translated most part of the Pipeline Classifier into Python.
- A MPI application distributing tasks over 200 DICE machines.
- Attempts on voting strategy to utilize classifier results.

The final cleaning removes 40% of the False Positives at the cost of 10% of True Positive. This reduces the Dataset by 28%.

Details - Parallel Distribution

The first challenge is the distribution of the task.

- This project uses the student lab DICE machines in Appleton Tower for the cleaning algorithm.
- There are 350 DICE machines in total. Due to auto-sleeping setting of some machines, about 220 can be used for cleaning.
- A Python script is used to “scan” for idle machines so the cleaning won’t disrupt other students.
- With the shared file system - AFS, the more portable MPI is used for the project. This gives fast communication between the cleaning processes, hence achieves the demand of task distribution.

Details - Standard I/O Stream Extraction

Another problem of the project is the lack of a SQL server.

- The F4K project's dataset contains a 500 GB `.sql` dump file.
- Computing Support recommends me not to load the dump file using school's PostgreSQL service.
- Since the records needed are independent, it's possible to partition the data into csv file with a standard I/O pipeline.
- After parsing, loading relevant information of a 2000-frame video from AFS took less than 1 seconds now.
- It also gives more portability to the project.

Details - Translating MATLAB code into Python

Originally, one of the main goal is to translate the Pipeline Classifier into Python.

- The project have translated most of the parts into Python.
- Except for the Feature Extraction in MATLAB and the Neural Network in Lua.
- After some benchmarking it is estimated the translation could take more time than running the code directly.
- PyMatlab and Lutorpy library is used to call the untranslated functions.

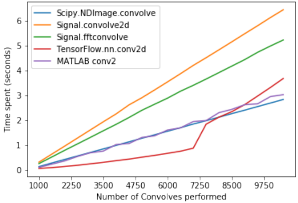


Figure: Performance of different convolution algorithms

Details - Ground Truth

In order to train the classifiers used in the pipeline, Pugh marked 60,000 detections manually using a 10-class classification schema.

In this schema, only class 6 and 8 are accepted fishes, while others will be rejected from the dataset.

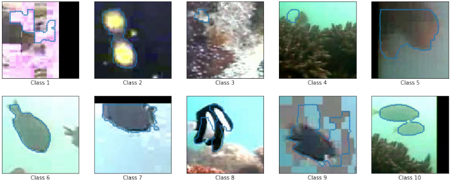


Figure: Sample fish from each class

Another 20,000 detections are marked in this project for complementing the missing cases, and performance evaluation.

Details - Classifiers

Pugh's pipeline classifier uses Support Vector Machine (SVM) and Convolutional Neural Network (CNN) for the classification of fish detections.

- The SVM performed the same as Pugh's expectation.
- Unfortunately, the CNNs trained does not work as intended. This is caused by color space issues in OpenCV. This leads to wrong transformation and normalization of the color spaces.
- Retraining the CNN would take a long time due to DICE machine's lack of CUDA support.

Figure: Normal Image in RGB space

Figure: OpenCV output, in BGR space

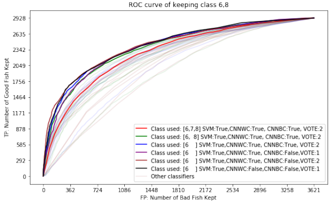
Details - Voting Strategy

Even after the CNN failure mentioned above, the CNN might still be useful. As it's giving 80% accuracy on class 6 (Fish) prediction.

In order to utilizing the result from CNN, an attempt of voting strategy using the classifier outputs were made.

To show the relationship between True/False Positive Rate, the ROC curve of different strategies are plotted.

With the need of keeping most of True Positives, the best strategy is using SVM results only.



Result - Outcome

The project manages to apply the cleaning algorithm developed by Pugh, with following statistics:

- 28% of the 1.6 TB dataset is removed.
- 40% of the False Positive is marked as Non-Fish.
- Finished the task in 11 days of computational time.

Result - Sample True Positive/ False Positive

<insert samples here>
<Forum power down - need access to project space to add images>

Lessons Learned

Future Work

Bibliography

[1] Matthew Pugh.
Removing false detections from a large fish image data-set.
Msc dissertation, The University of Edinburgh, 2015.

[2] Qiqi Yu.
Adding temporal constraints to a large data cleaning problem.
Msc dissertation, The University of Edinburgh, 2016.

[3] Robert B Fisher, Yun-Heh Chen-Burger, Daniela Giordano, Lynda Hardman, Fang-Pang Lin.
Fish4Knowledge: collecting and analyzing massive coral reef fish video data.
Springer, 2016.