CMPE 255 - Final Project Report

(Application Option)

Devansh Modi[#1], Abhighna Kapilavai[#2]

[#]*Department of Software Engineering*
San Jose State University
1 Washington Sq, San Jose, CA 95129

I. Objective

Indoor scene recognition is a challenging open problem in high level vision. Most scene recognition models that work well for outdoor scenes perform poorly in the indoor domain. The main difficulty is that while some indoor scenes (e.g. corridors) can be well characterized by global spatial properties, others (e.g., bookstores) are better characterized by the objects they contain. More generally, to address the indoor scenes recognition problem we need a model that can exploit local and global discriminative information.

The objective of this project is to support indoor scene recognition for kitchen remodeling domain. It is a novel idea as there exists no datasets for kitchen layouts today, yet there is an ever-increasing need for kitchen remodeling services.

II. Datasets

The dataset used for this project included a custom built and labeled data for classifying countertop material and kitchen layout.

Label Classifiers include:

- Wood Countertop
- Stone Countertop
- L-Shaped Layout
- U-Shaped Layout
- One-Wall Layout

As per data from major retailers, these classifiers comprise over 90% of the single family home kitchens in the United States.

In order to classify kitchen appliances and kitchen objects, we use MIT Indoor Scene Recognition open dataset and COCO Object Recognition dataset.

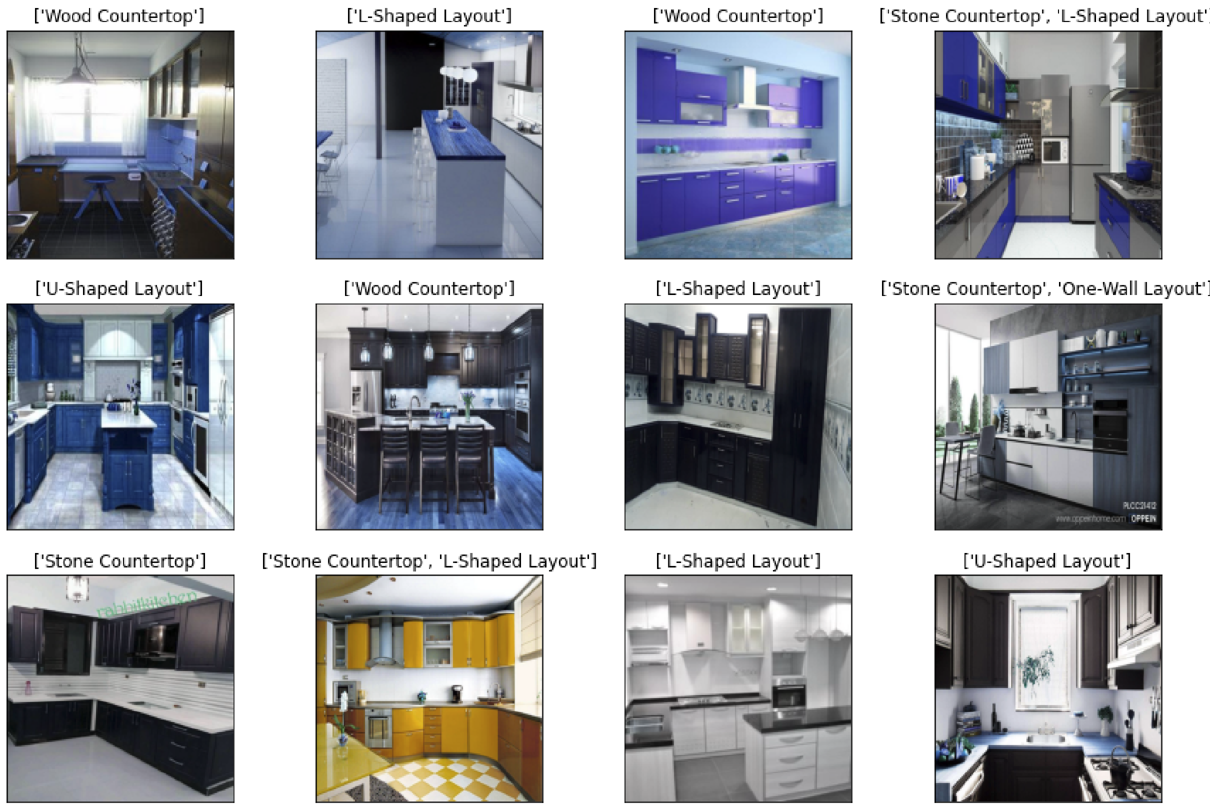


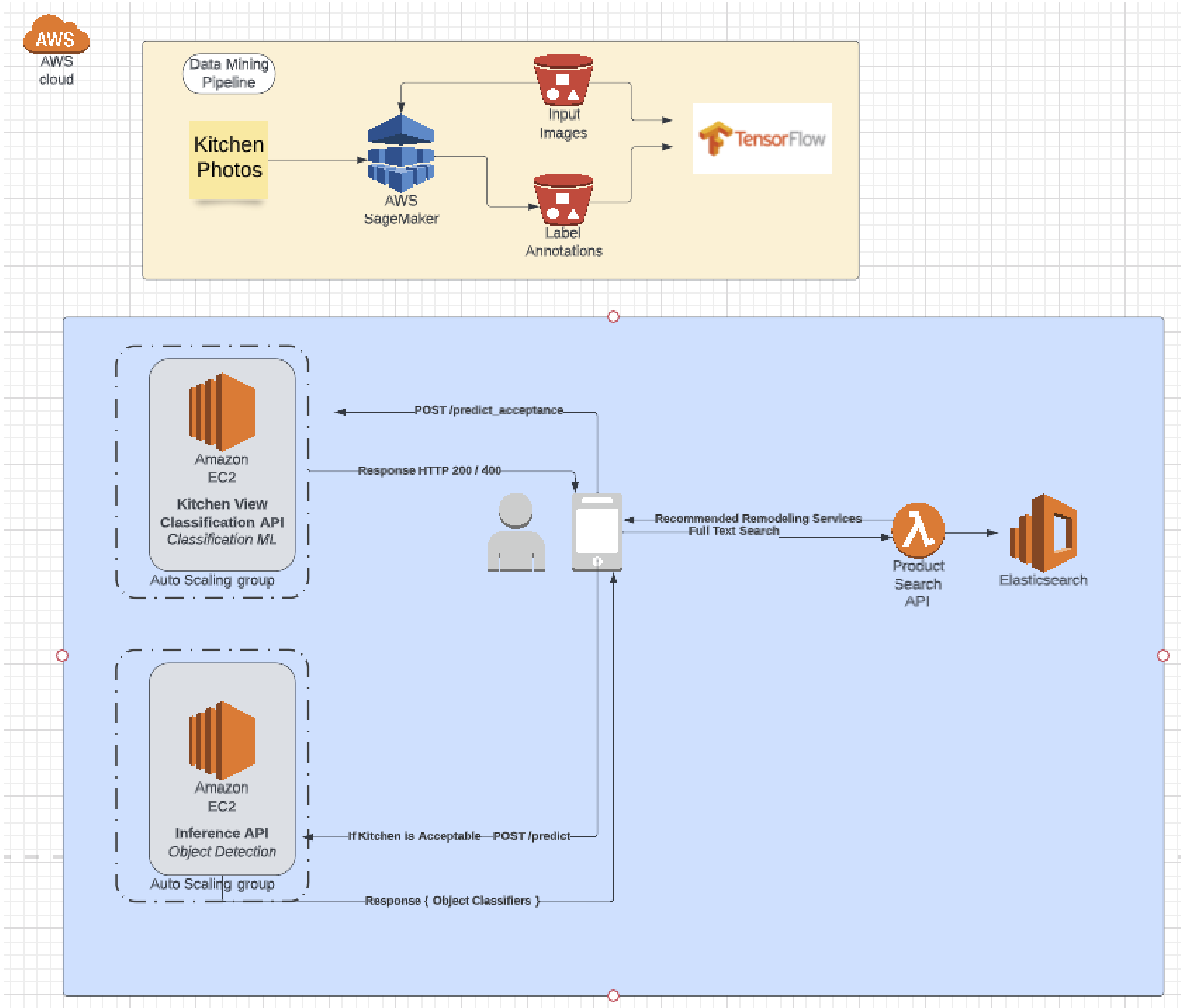
Figure: Kitchen Layout Labels

Figure: Cabinet and Objects

Data Preprocessing included steps such as:

- Image Resizing
- Shuffling of dataset
- Filtering of unsuitable images.
- Multiple iterations of label verification.

III. Architecture



IV. Modeling

We used multiple models to compare performance. Here is the list:

1. Self-built Sequential Model
2. EfficientNet
3. ResNet50
4. MobileNetV2
5. AWS SageMaker Auto Algorithm

Hyperparameters

You can use hyperparameters to finely control training. We've set default hyperparameters for the algorithm you've chosen.

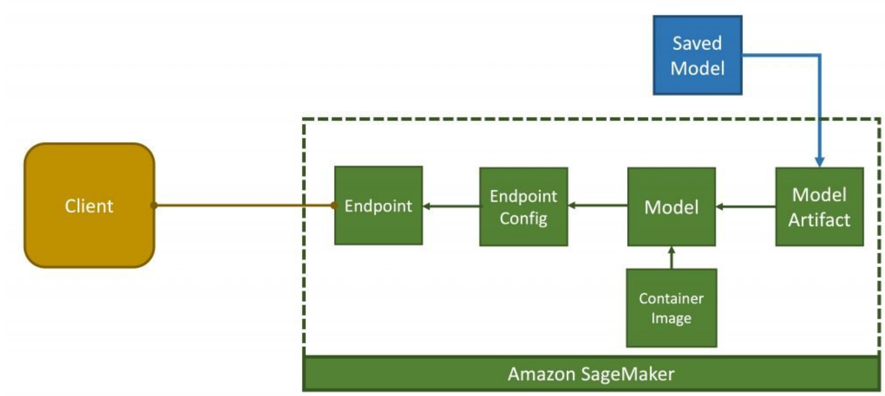
Key	Value
min_val_samples	20
learning_rate	0.001
momentum	0.9
model_depth	101
settings	Multi-Label
batch_size	0
lr_decay	0.1
lr_patience	5
max_patience	10
n_epochs	30
result_second_interval	3
thresholdValue	0.5
score_result_threshold	0.5
num_result_tags	5

V. Model Evaluation

VI. Cloud Technologies

We have made heavy use of Cloud Computing and AWS specifically.

Our backend architecture is completely based on AWS, SageMaker, and Flask along with custom code. The frontend mobile app uses Android along with Tensorflow.



VII. Conclusion

All in all, this project was exciting as it allowed us to work on a complete data mining pipeline. We were able to evaluate multiple models and test inference response time via multiple mechanisms. Our end choice was to use our custom model instead of SageMaker due to cost optimization and higher performance using GPU.

VIII. References

Kaggle Notebook
https://www.kaggle.com/code/gpiosenka/utensils-f1-test-score-98

Example Project by a Startup
https://www.abtosoftware.com/blog/image-recognition-kitchen-furniture-appliances (They achieved 78% accuracy without using CNN)