Traffic Sign Recognition

Group 1

Xuhao SunGY WuDi WuTY

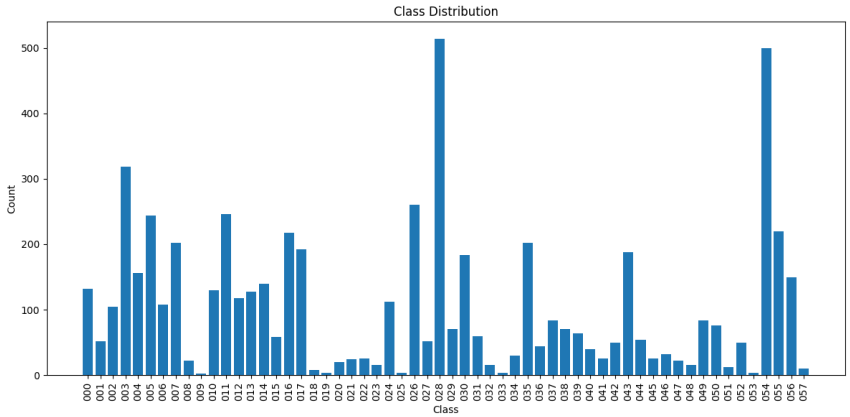
NUS Summer Workshop

July 23, 2023

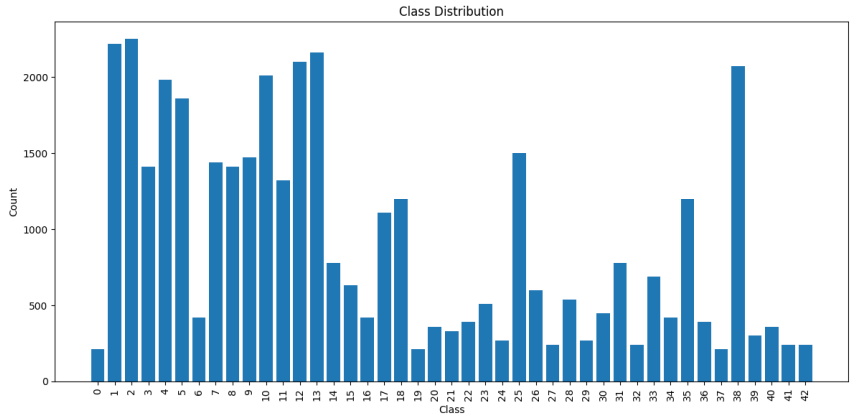
Presentation Overview

1. Challenge Identification: What is our problem?
2. Methods & Tricks: How can we do it?
3. Visualization & Comparison: Peeking inside & Compare Performance
4. Conclusion: In short, how did we finish the problem.

Both datasets are imbalenced



(a) Dataset1



(b) Dataset2

Figure: Class Distribution

Dataset2 is more imbalanced than Dataset1

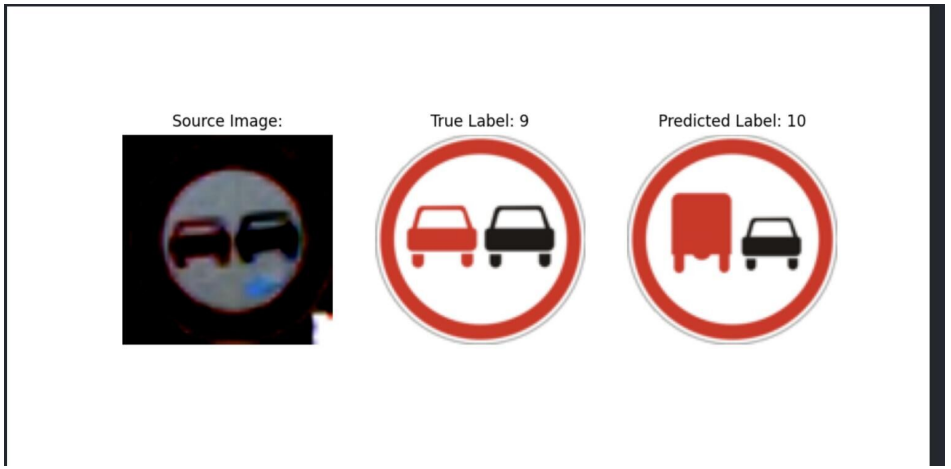
	Standard Deviation	Coefficient of Variation
dataset1	109.62570087579336	1.060068464620876 2
dataset2	687.7177601932608	0.754211117047367

Hard Cases in Dataset2

Challenge Identification

Occlusion, High Similarity, Blurriness etc. These dataset limitations can negatively affect the performance and generalization ability of ML models. Above make Dataset2 more challenging.

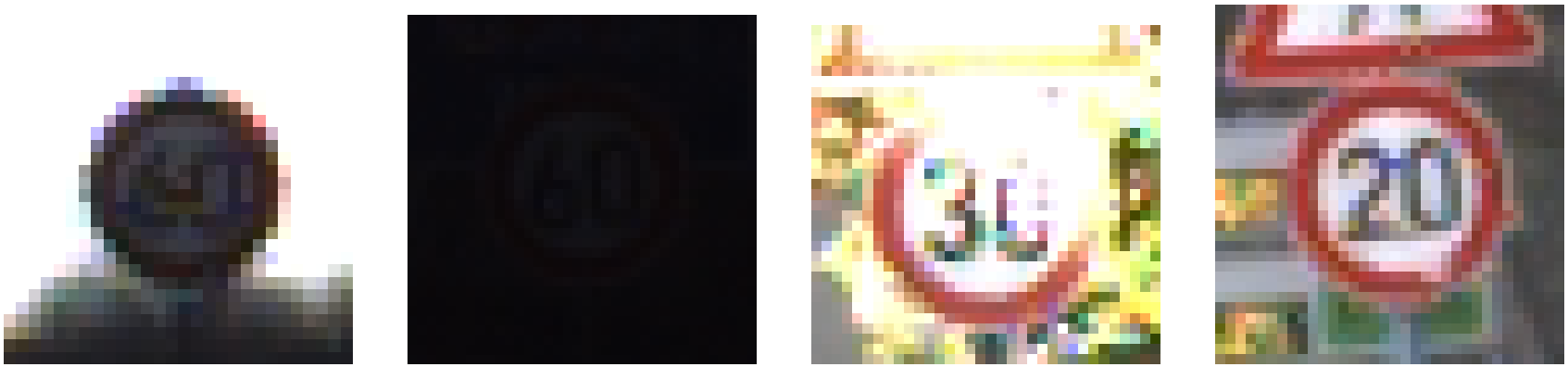
(a) Occlusion



(b) High Similarity

Hard Cases in Dataset2

Challenge Identification



(c) Blurriness (d) Underexposure (e) Glare Effect (f) Interference

Figure: Hard Cases

Expert Level

Methods & Tricks

- Data preprocessing methods?
- Feature extraction methods?
- Different classification models we use?
- What tricks we can use to enhance the model performance?

Data preprocessing methods

- Resize
- Histogram Equalization
- GaussianBlur
- Normalization

Feature extraction methods

- Feature Pyramid: Use Gaussian pyramids to extract image features layer by layer.
- HOG(Histogram of Oriented Gradients): Extract local texture and edge information.
- LBP(Local Binary Pattern): Encodes the local relationship of pixel intensities with their neighbors.
- FFT(Fast Fourier Transform): Extract the frequency domain information of the image.
- SIFT: Detects and describes key points in an image,
- Concatenation: Concatenate all the features above.

Different classification models

- SVC
- Random Forest Classifier
- KNeighbors Classifier
- Decision Tree Classifier
- Gaussian Naive Bayes
- MLP Classifier

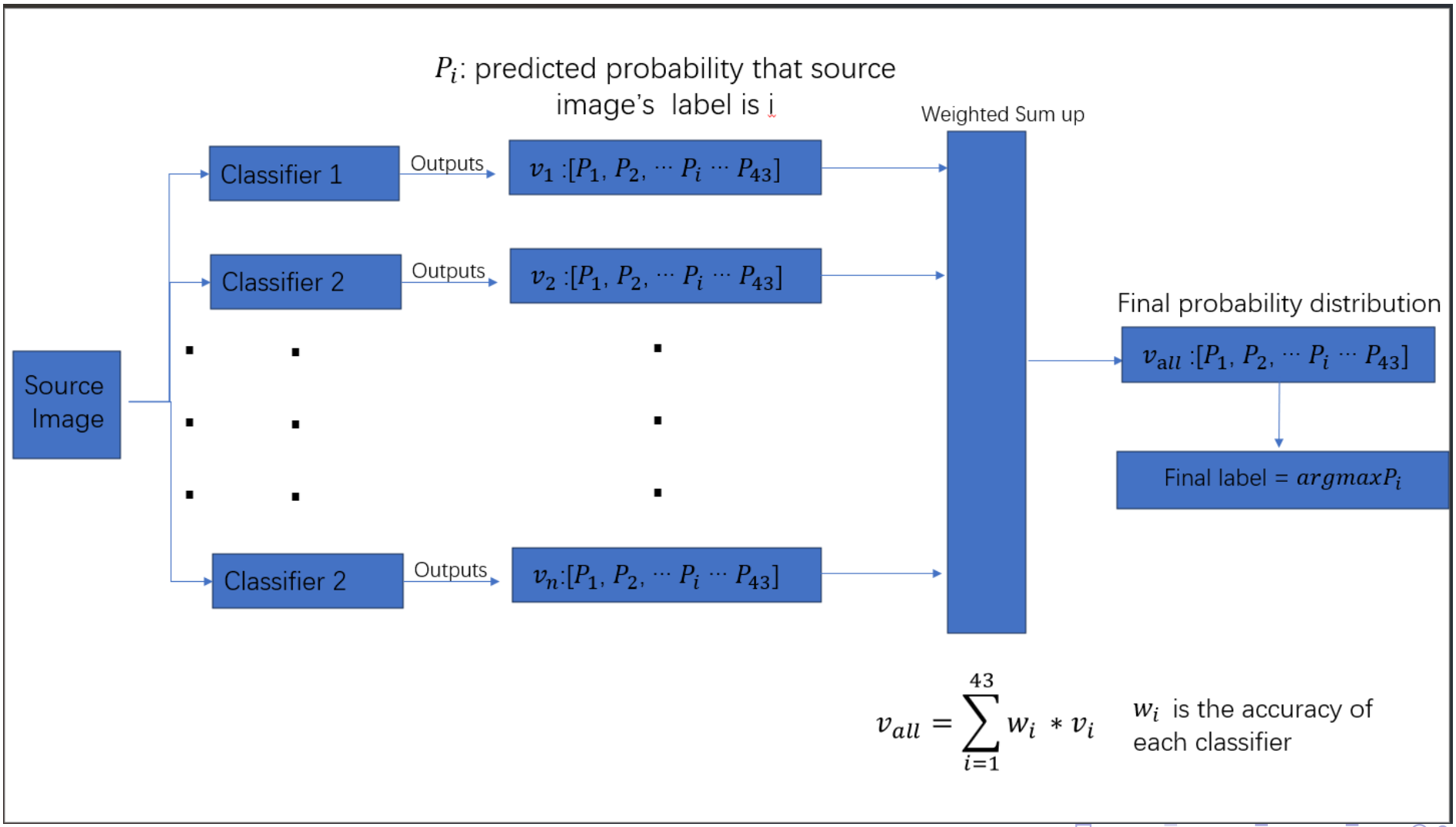
What tricks we use to enhance the model performance

- Concatenating features: Not good here, it doesn't help improve the performance.
- Voting: It can enhance the model performance by 2% - 5%.

Trick: Voting

Methods & Tricks(Expert Level)

Let the best combinations vote together.



Visualization & Comparison

Visualization & Comparison(Expert Level)

All combinations of different feature extraction methods and classifiers.

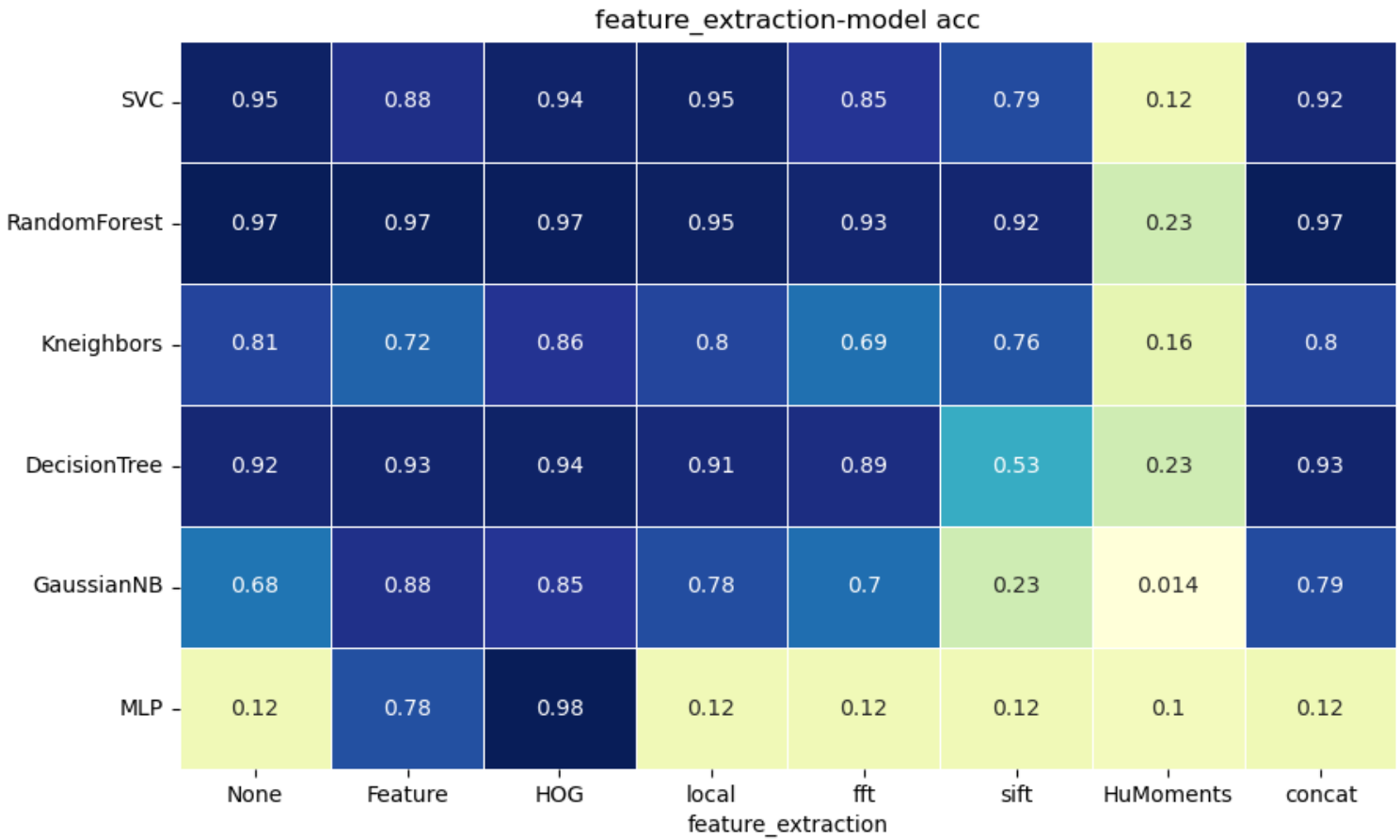


Figure: feature extraction-model

Visualization & Comparison

Visualization & Comparison(Expert Level)

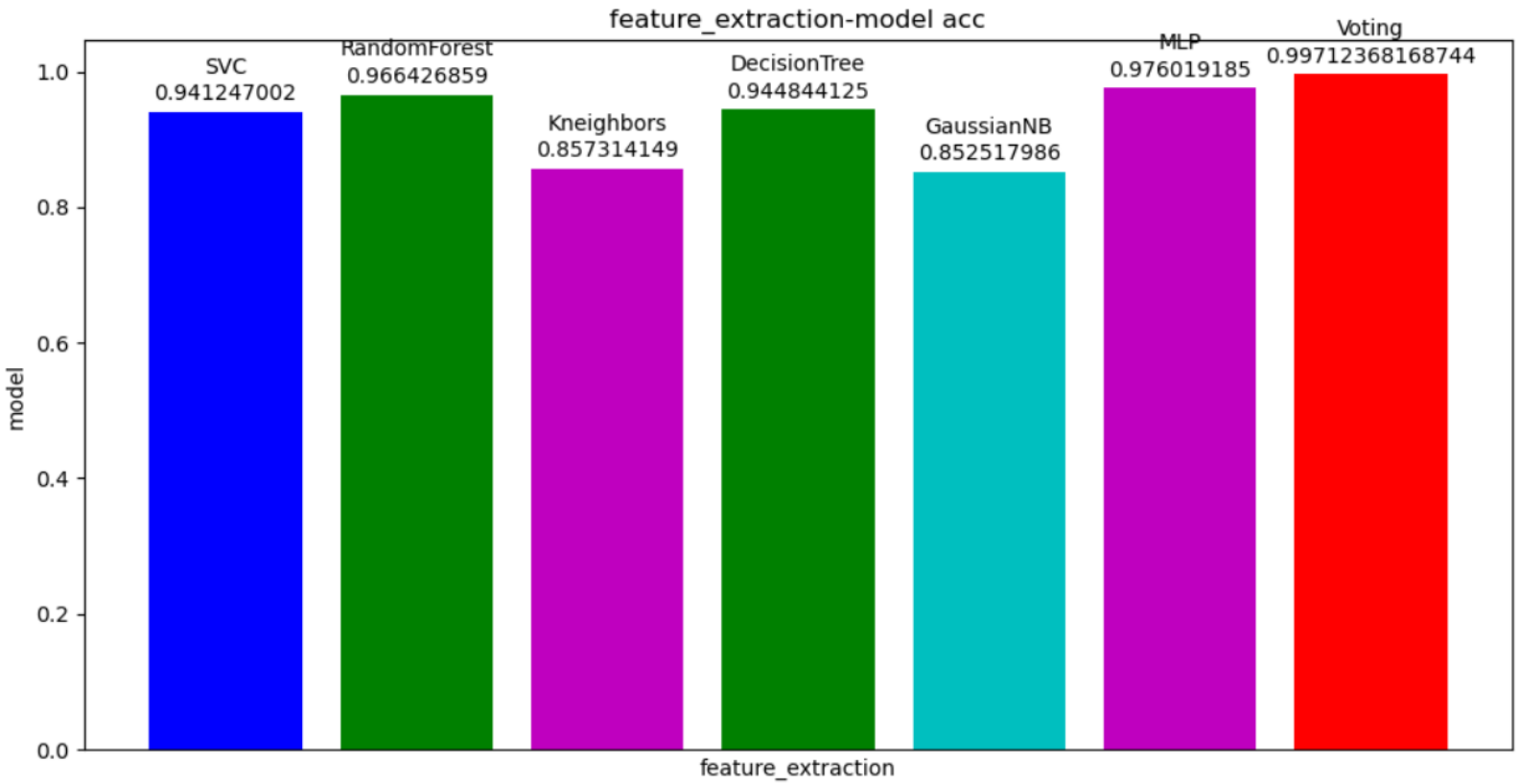


Figure: Models and Voting results

After voting, the final accuracy on the Dataset 1 reaches 99.71%. We use our voting model to classify Dataset 2, and get an accuracy of 76.90%.

Is this the end?

Welcome to Bonus Level!

Core difference: instead of designing features on our own,
let the neural networks find features for us!

Bonus Level

Methods & Tricks(Bonus Level)

- Data preprocessing
- Neural Networks
- Visualization
- Performance Analysis
- Failure Analysis

Data Preprocessing

Methods & Tricks(Bonus level)

Data preprocessing methods

- Crop
- Resize
- Data Augmentation? (rotate? brightness & saturation? balabala)
- ToTensor
- Normalize

Neural Networks we tried

- Alexnet
- Lenet
- my-net(build on our own from scratch)
- Resnet18
- Squeeze-net
- Vgg16

? What happened inside the Neural networks?

ResNet-18 as an example

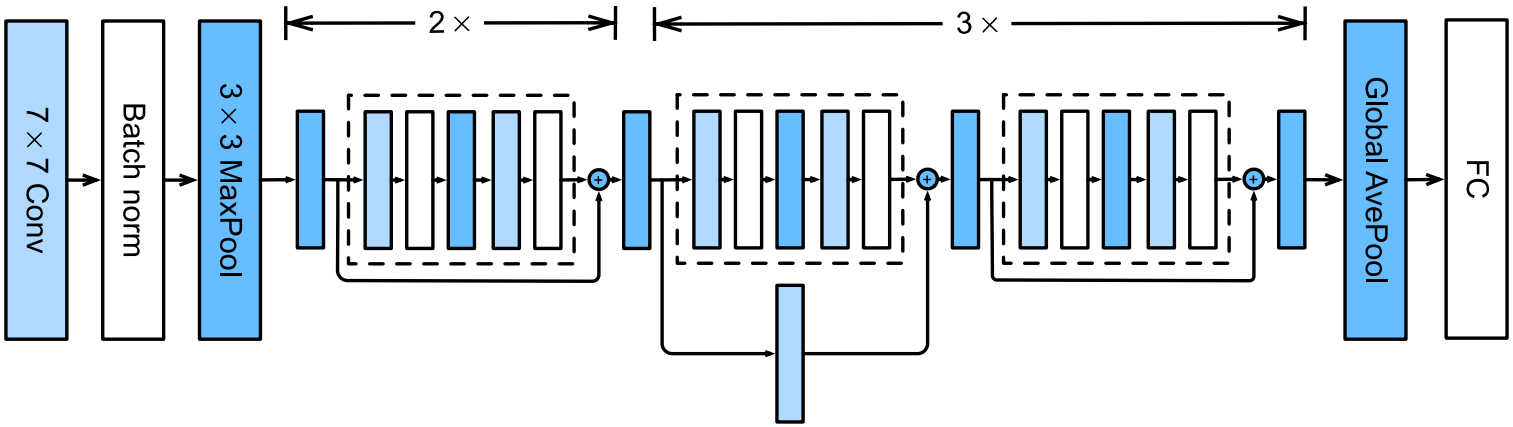


Figure: The ResNet-18 architecture.

Visualization & Analysis

with Feature Map

Feature maps constructed by the Neural networks:

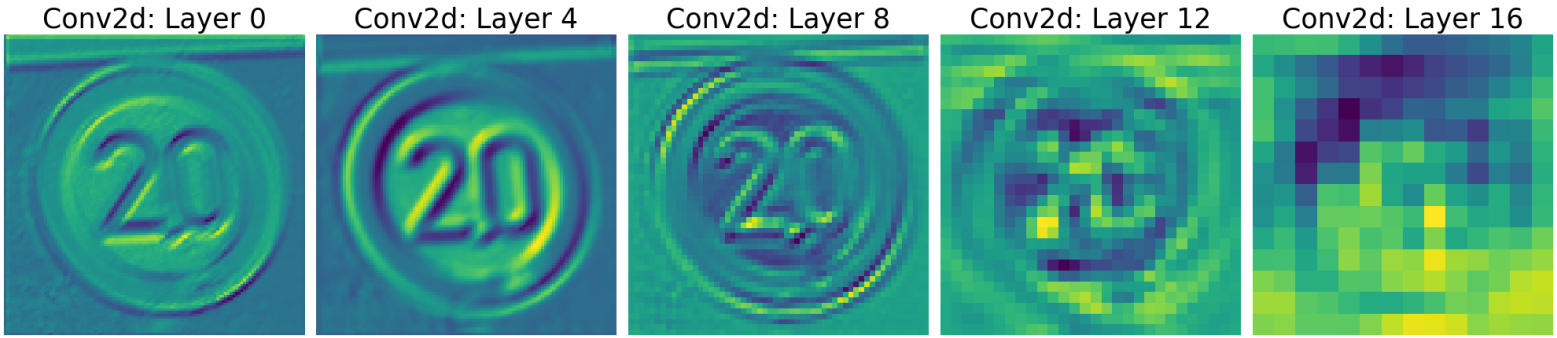


Figure: ResNet-18 Feature Maps with pretraining

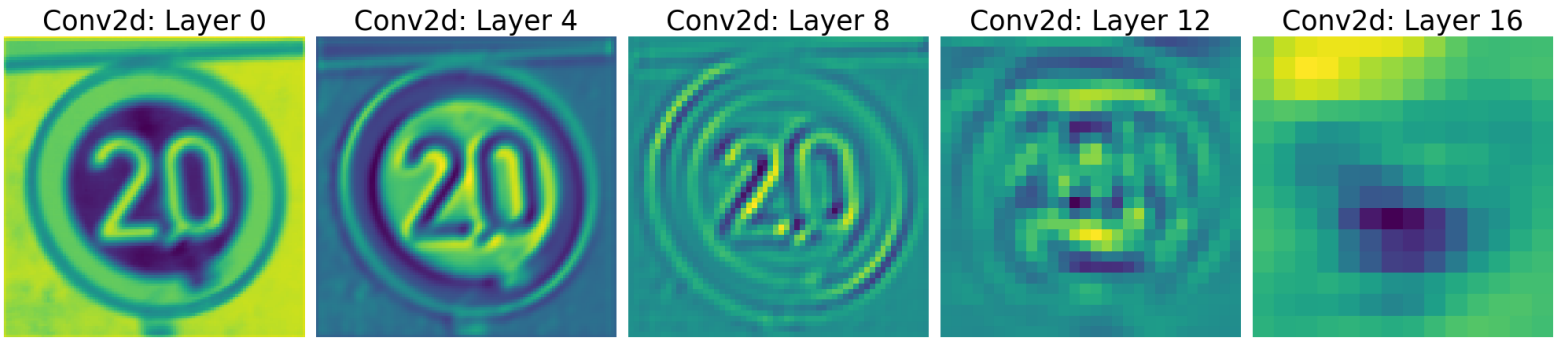


Figure: ResNet-18 Feature Maps without pretraining

🔍 Visualization & Analysis

with Grad-CAM Heatmaps

Figure: with pretraining 👍

Figure: without pretraining 👎

Performance Analysis

Visualization & Comparison(Bonus level)

Figure: Train loss

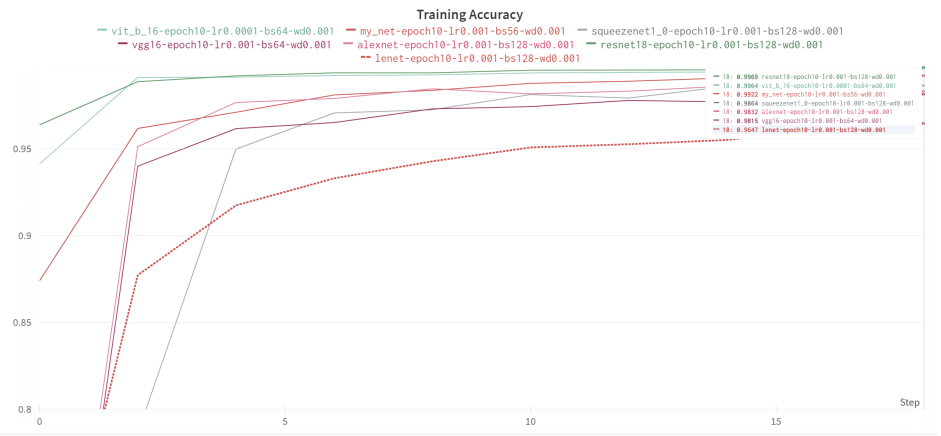


Figure: Train Accuracy

Performance Analysis

Visualization & Comparison(Bonus level)

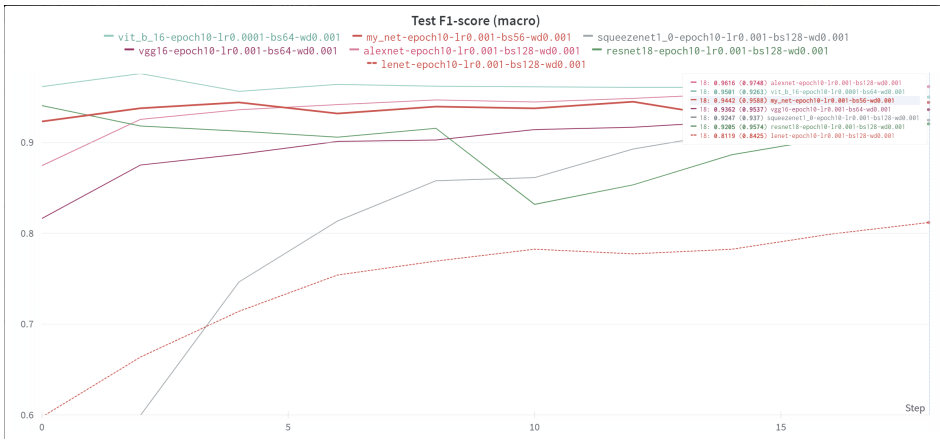


Figure: Test f1-score

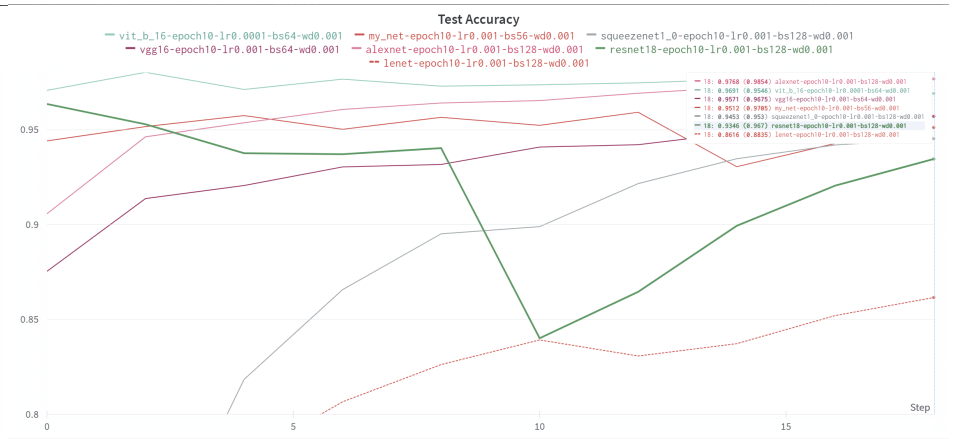


Figure: Test Accuracy

Confusion Matrix

Visualization & Comparison(Bonus level)

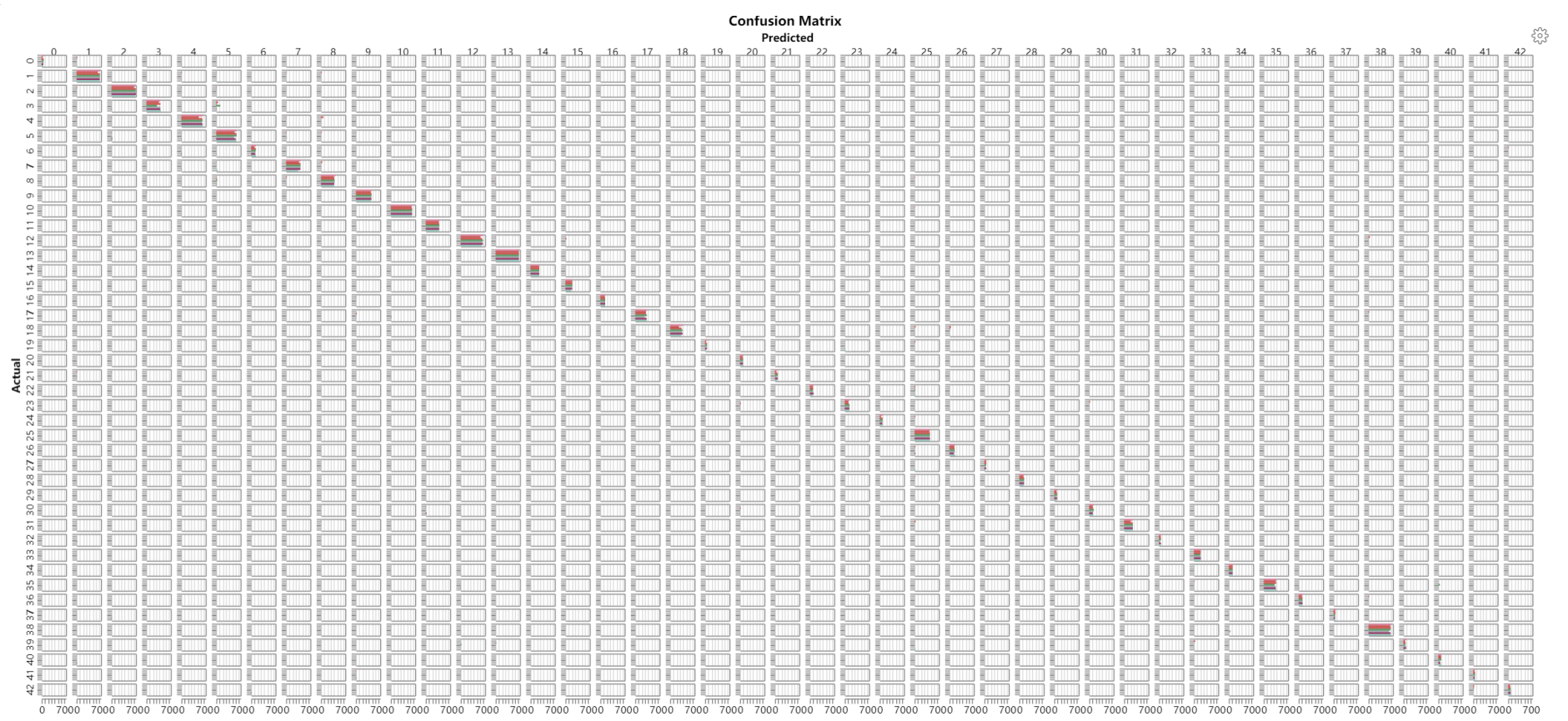


Figure: Confusion Matrix

Compare Performance

Visualization & Comparison(Bonus level)

Table: Best performance of models
(Hyperparameters set: epoch=10, learning rate=0.001, weight_decay=0.001, batch_size=128)

Alexnet	Lenet	my-net	Resnet18	Squeezenet	Vgg16
0.9854	0.8856	0.9736	0.9759	0.9696	0.9675

Overall, the performances of different models are between 96.7% to 98.7%.

Failure Analysis

Visualization & Comparison(Bonus level)

Question: What are those failed predictions? (Top Left: Source Image, Top Right: Processed Image, Bottom Left: True label, Bottom Right: Predicted Label)

(a) Tilted (b) Underexposure

Figure: These failed cases are within our ability to solve!

What tricks we use to enhance the model performance? Here's 2 ideas:

- Data augmentation: We can expand the training dataset, the model may have more data to learn from, thus being more robust to different situations.
- Voting: We can integrate all the different models, combined their predictions to make the final decision.

Data augmentation

Methods & Tricks(Bonus level)

We can set a transform function on the training dataset: randomly rotate, randomly adjust brightness and saturation and etc.

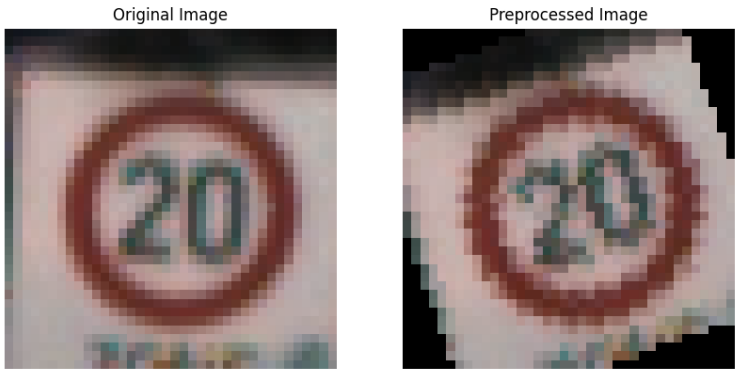


Figure: Random Rotation

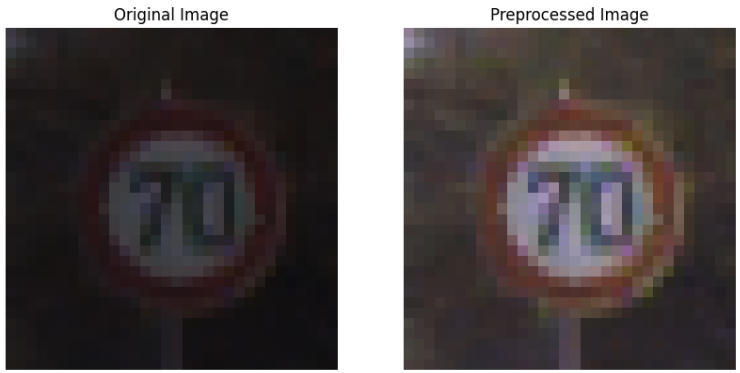


Figure: Color Transformation

With this approach, we effectively addressed the issues of "Underexposure" and "Rotation" cases.

Voting

Methods & Tricks(Bonus level)

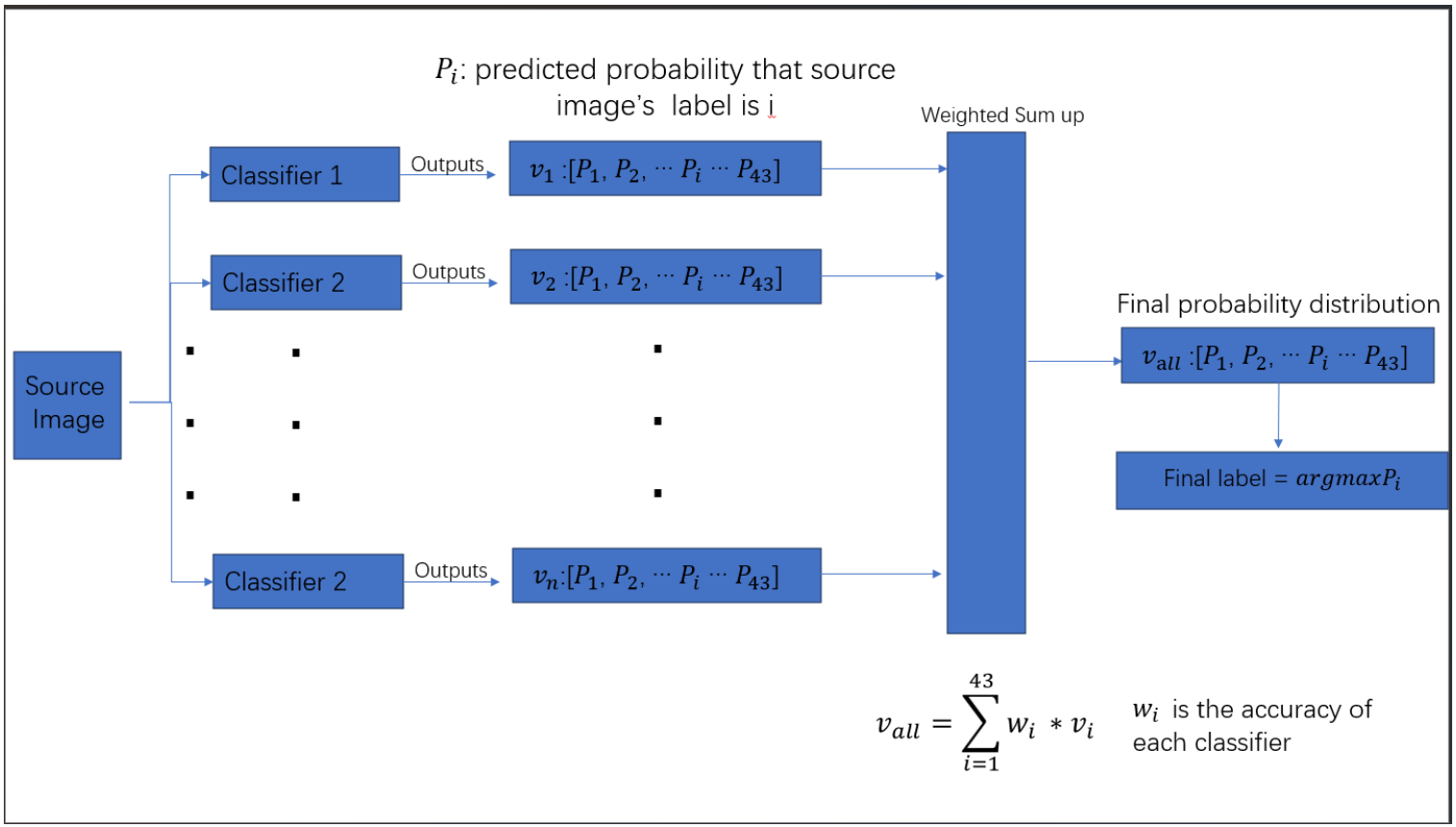


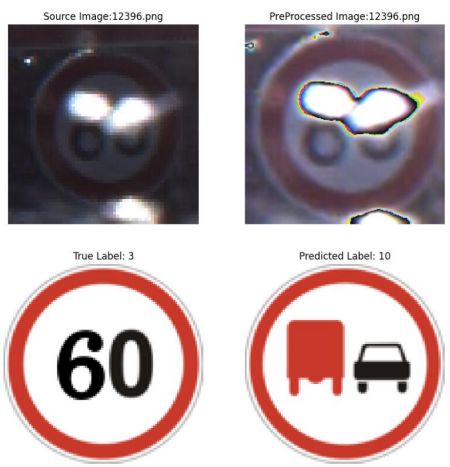
Figure: schematic of voting

After voting, the final accuracy on the Dataset 2 reaches 99.58%.

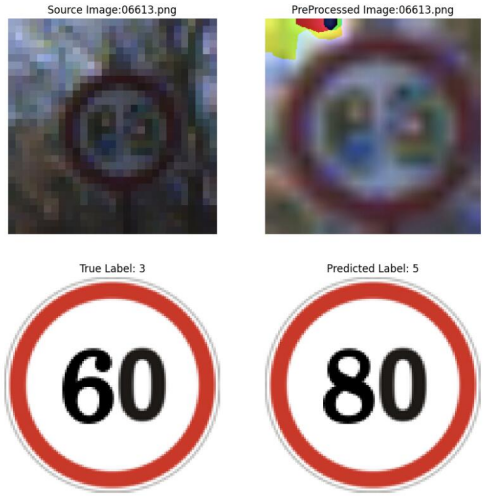
Failure Analysis & Future work

Visualization & Comparison(Bonus level)

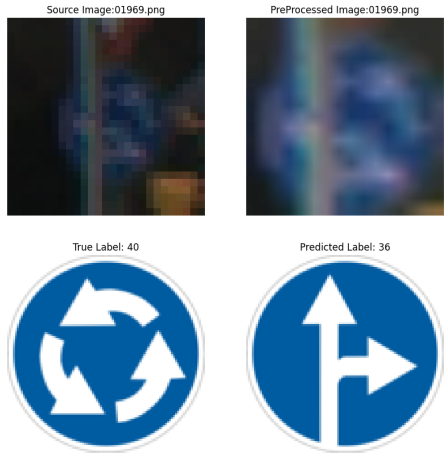
Question: What are those 0.42% failed predictions? (Top Left: Source Image, Top Right: Processed Image, Bottom Left: True label, Bottom Right: Predicted Label)



(a) Glare effect



(b) Blurriness



(c) Occlusion

Figure: Most failed cases are even beyond human's ability to recognize!

Conclusion

Our works includes:

- Analyze our 2 Datasets.
- Adopt lots of data preprocessing methods, feature extraction methods, classifiers, neural networks and tricks.
- Visualize and analyze their performances

Dataset 1, we use traditional methods and get an accuracy of 99.71%.
Dataset 2, our integrate model gets an accuracy of 99.58%.

The End

Questions? Comments?

Appendix

Future Work

1. Explore approaches to solve the remaining failed cases
2. More Analysis on Deep Neural Network
3. More explore in Ensemble Methods
4. Evaluate Real-Time Performance
5. More cameras on car