BOSCH TRAFFIC SIGN RECOGNITION INTER IIT CHALLENGE REPORT

German Traffic Sign Recognition Benchmark Dataset description:

- Training set : 39209 images
- Test set: 12630 images
- No of classes: 43

Training WorkFlow:

- Trained *Baseline network* on the original training dataset.

  Specifications:

  - Models: Resnet18
  - Preprocessing:
    - Resize image to size (64,64)
    - Normalize image
  - Training parameters:
    - Epochs : 30
    - Loss : CrossEntropyLoss
    - Optimizer : Adam
  - Results:

Metrics	Value
F1 Score weighted	0.9872
F1 Score macro	0.9816
Accuracy score	0.9872
Precision score	0.9821
Recall score	0.9817

- Tested this network on **original test dataset** + **added augmentations** generated from dataset creation UI

➢ Results :

Metrics	Value
F1 Score weighted	0.9690
F1 Score macro	0.9610
Accuracy score	0.9690
Precision score	0.9685
Recall score	0.9554

- Added 5 new classes to the dataset, with images sources from the internet.

Class Name	Class Number	Sample Image	Number of Images
No left turn	43		246
No Horn	44		192
Car	45		514
Pedestrian Crossing	46		202

No stopping	47		500

20% of these images were added in the test set and remaining in the training set.
Updated German Traffic Sign Recognition Benchmark Dataset: In the updated dataset, there are 40532 training images and 12961 test images.
[Link to images]

- The **baseline model** was **modified** for **48 classes** (using finetuning) and evaluated on the updated test set.

  - **Result:**

Metrics	Value
F1 Score weighted	0.9858
F1 Score macro	0.9762
Accuracy score	0.9857
Precision score	0.9739
Recall score	0.9803

- **Tested** this **network** on **updated test dataset of 48 classes + added augmentations** generated from **Dataset Creation UI** :

  - **Results :**

Metrics	Value
F1 Score weighted	0.9697
F1 Score macro	0.9523
Accuracy score	0.9687
Precision score	0.9516
Recall score	0.9603

- *Based on intuition from Post Evaluation UI:*

  We studied the results of the above model on post-evaluation UI and got intuition for further experiment regarding the performance of the model.
  Made necessary changes in the network training parameters and dataset and trained further.

  Experiments done to improve metric :
  - Added unseen train time augmentations
  - Handled class imbalance by penalizing loss.
  - Applied Early stopping
  - Regularization using CoarseDropout
  - Tried different networks

  Tested above model on updated test dataset of 48 classes + added augmentations generated dataset creation UI

  - ➢ **Results :**

Metrics	Value
F1 Score weighted	0.9815
F1 Score macro	0.9719
Accuracy score	0.9813
Precision score	0.9677
Recall score	0.9772

Summarized Results :

Note: Baseline model was trained on 43 classes and was fine-tuned for evaluating on 48 classes.

- *Baseline Network evaluation on test dataset results:*

Number of Classes	Added Dataset Creation UI Augmented Images ?	F1 Macro	Accuracy
43	No	0.9816	0.9872
43	Yes	0.9610	0.9690
48	No	0.9762	0.9857
48	Yes	0.9523	0.9687

- *Improved Baseline Network evaluation on test dataset results :*

Number of Classes	Added Dataset Creation UI Augmented Images ?	F1 Macro	Accuracy
48	Yes	0.9719	0.9813

[Link to Notebook]

Data Creation UI :

Features:

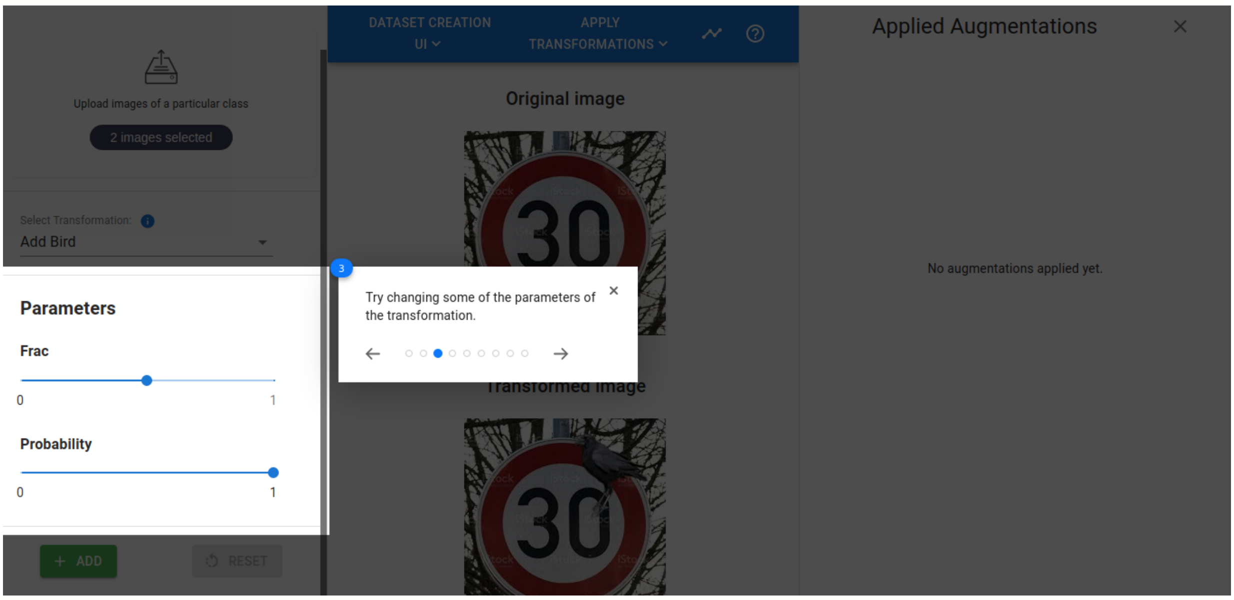
- ➔ Applying Augmentations
- ➔ Balancing Dataset
- ➔ Splitting Dataset

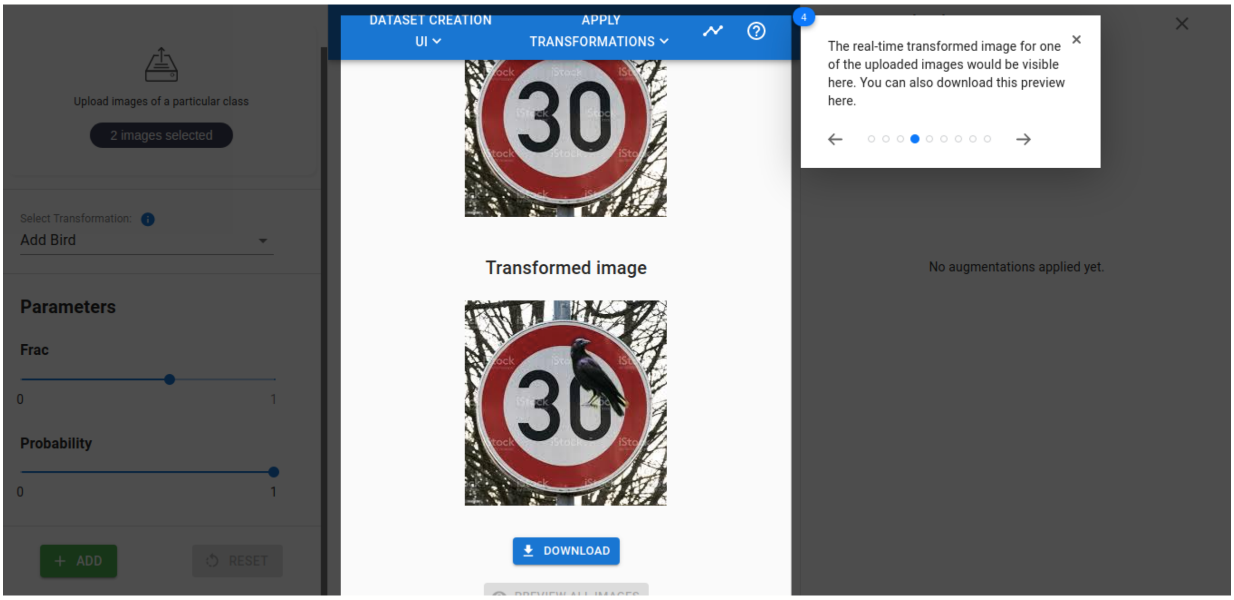
- *Applying Augmentations:*
  This feature enables users to try different augmentations among the 70 listed augmentations and view the preview in real-time.
  It supports multi images and also multiple augmentations can be applied.
  Users can download an augmented image or can directly add it to the dataset.

- *Balance dataset:*
  This feature balances the dataset. It achieves this by oversampling the examples from the minority class by applying a set of transformations to the images. For every class. where the number of samples are less than the provided threshold, the tool upsample that class images by applying a set of transformations.

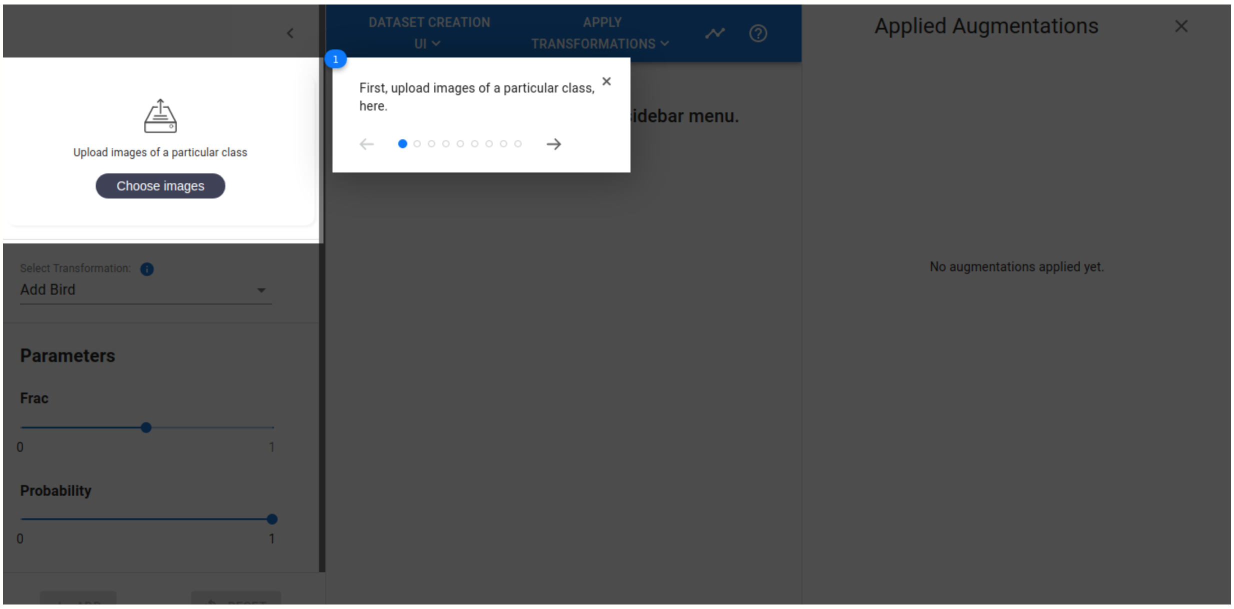
- *Split Dataset:*
  This feature assists you to split the dataset into train and test sets. A stratified split would be created maintaining the class-wise ratio in both train and test sets.

Screenshots:

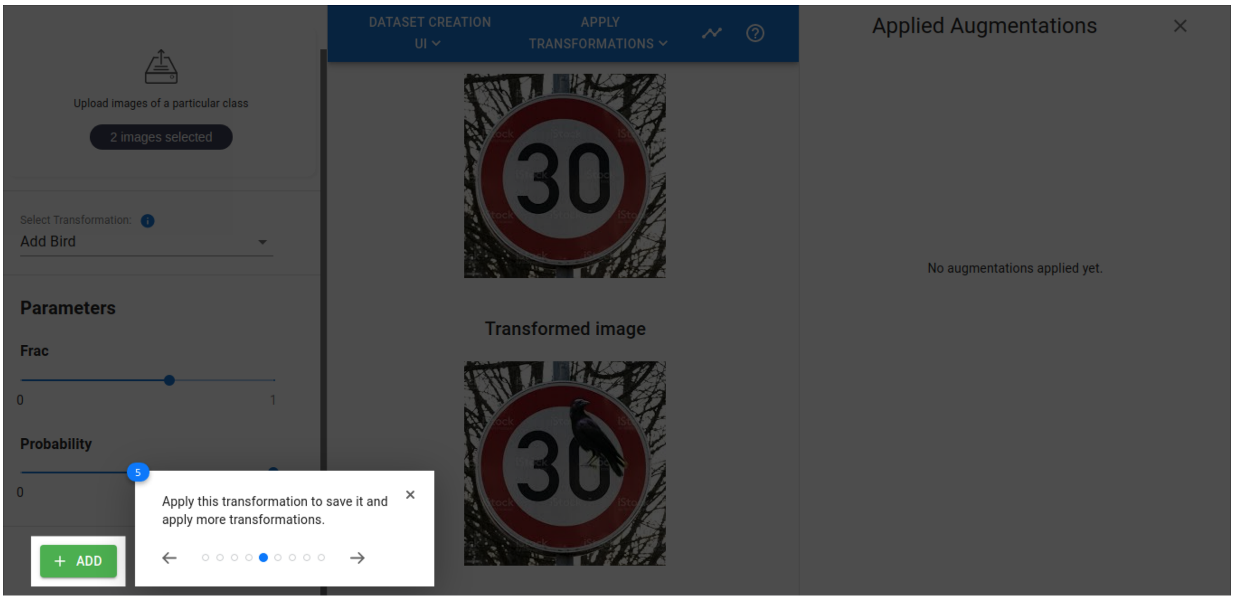




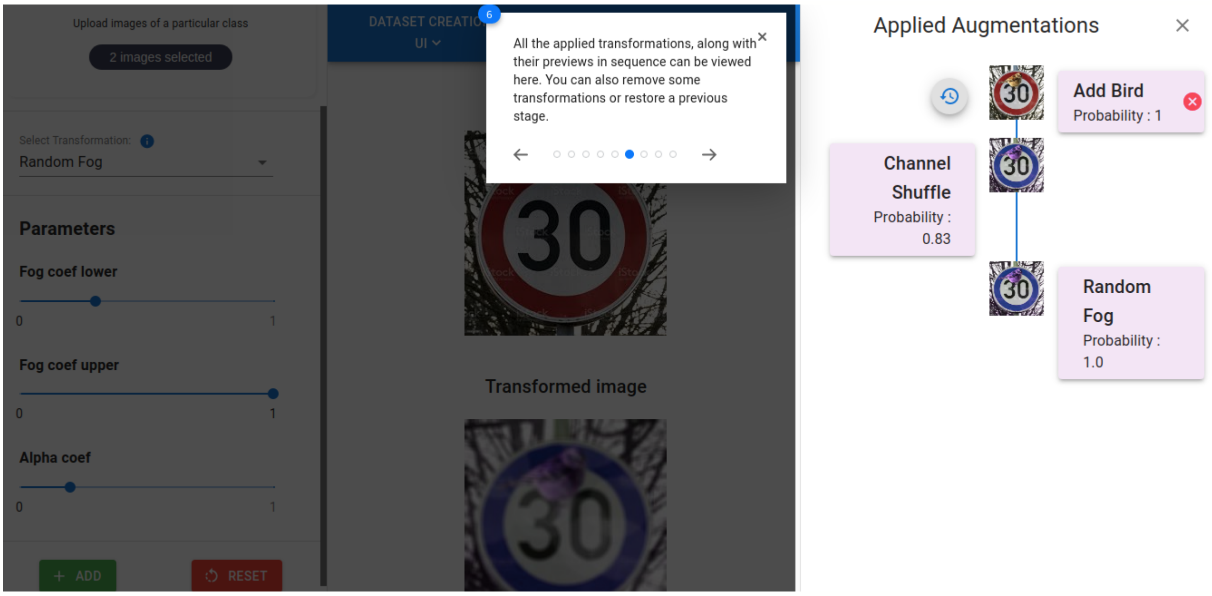
DATASET CREATION UI
APPLY TRANSFORMATIONS
Upload images of a particular class
2 images selected
Select Transformation:
Add Bird
Parameters
Frac
0
1
Probability
0
1
ADD
RESET
Transformed image
DOWNLOAD
The real-time transformed image for one of the uploaded images would be visible here. You can also download this preview here.
No augmentations applied yet.

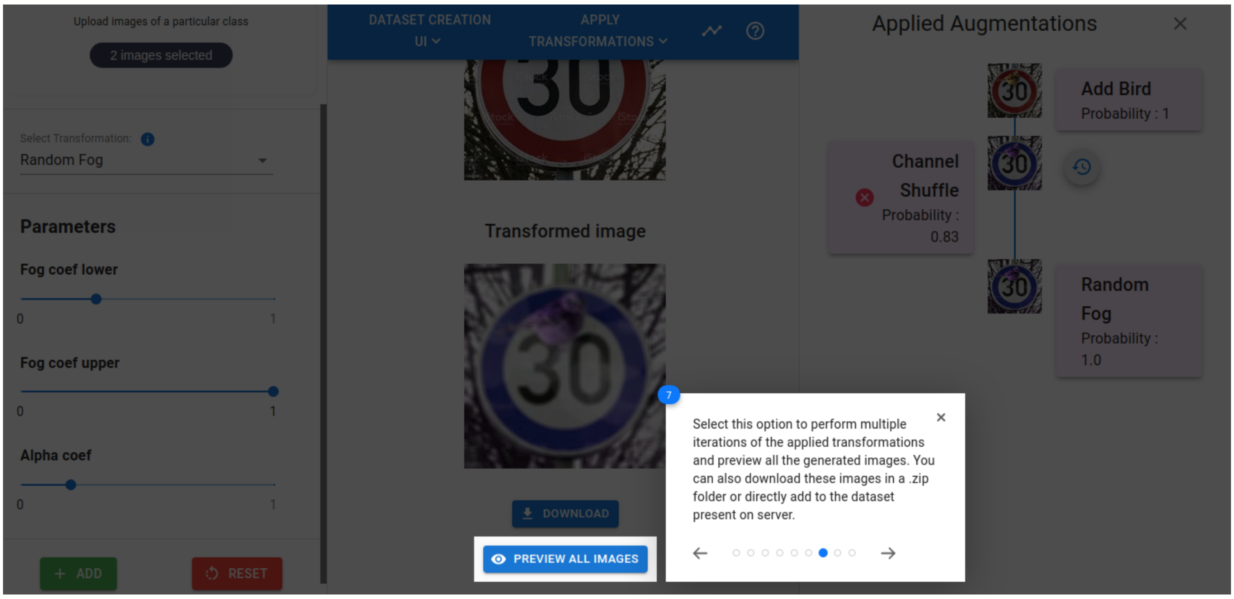


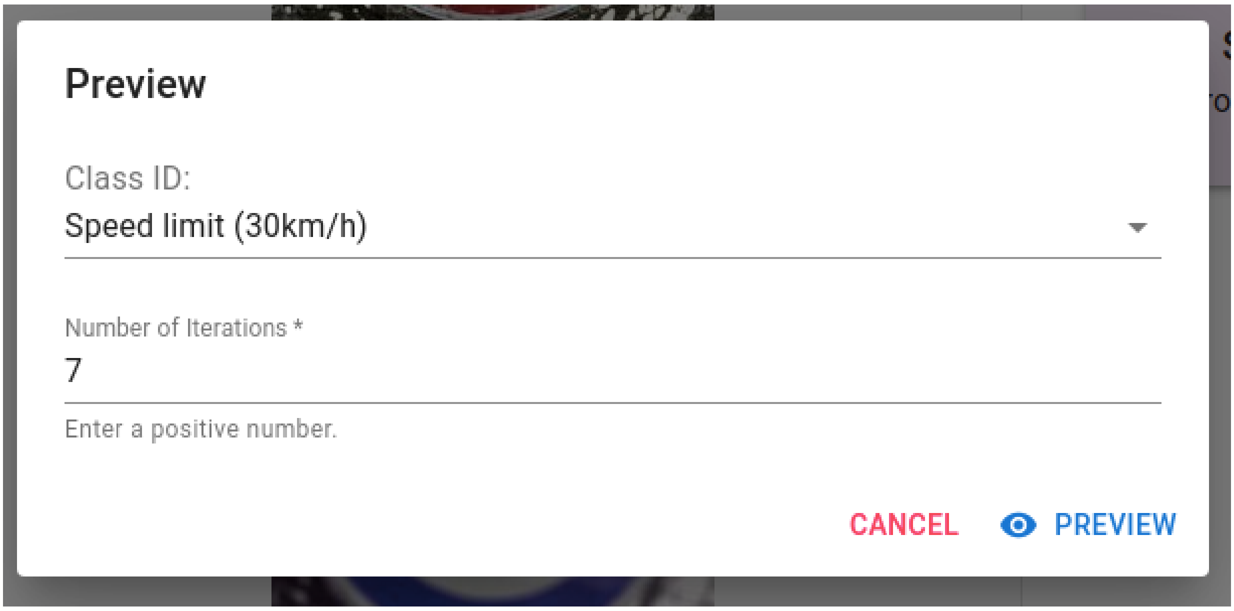
DATASET CREATION UI
APPLY TRANSFORMATIONS
Applied Augmentations
Upload images of a particular class
Choose images
First, upload images of a particular class, here.
idebar menu.
Select Transformation:
Add Bird
Parameters
Frac
0
1
Probability
0
1
No augmentations applied yet.

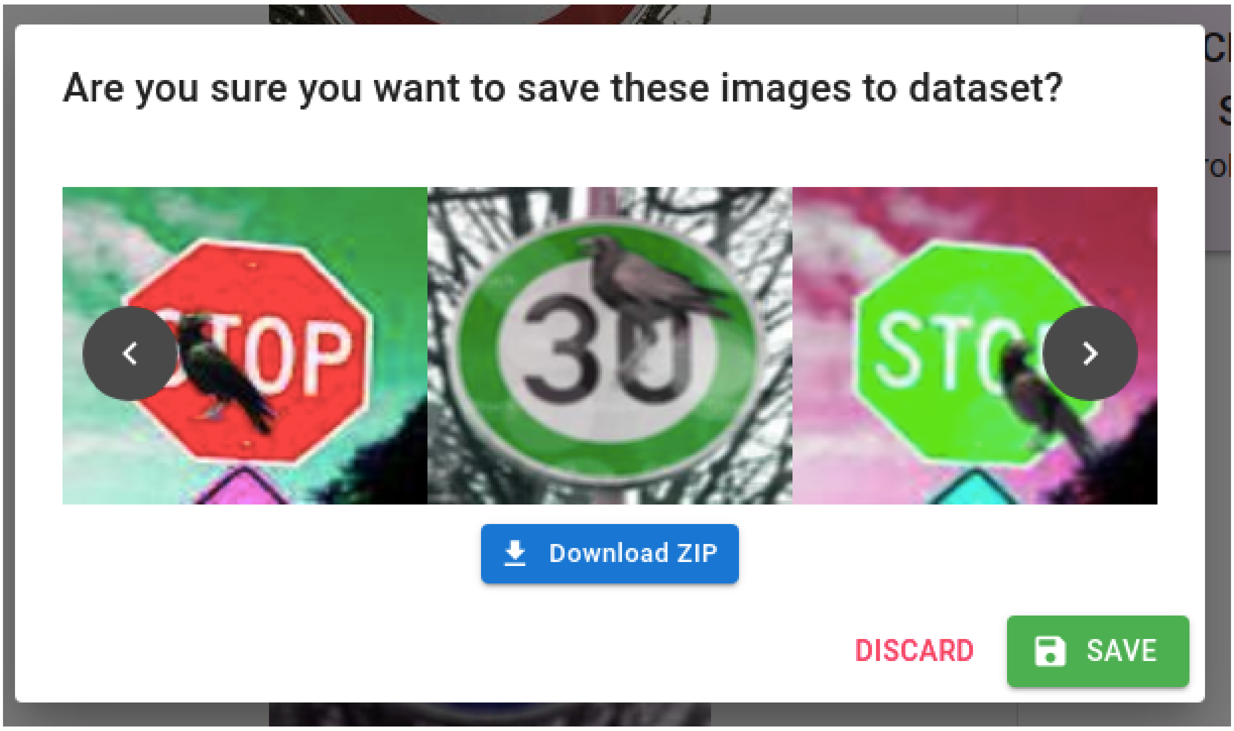


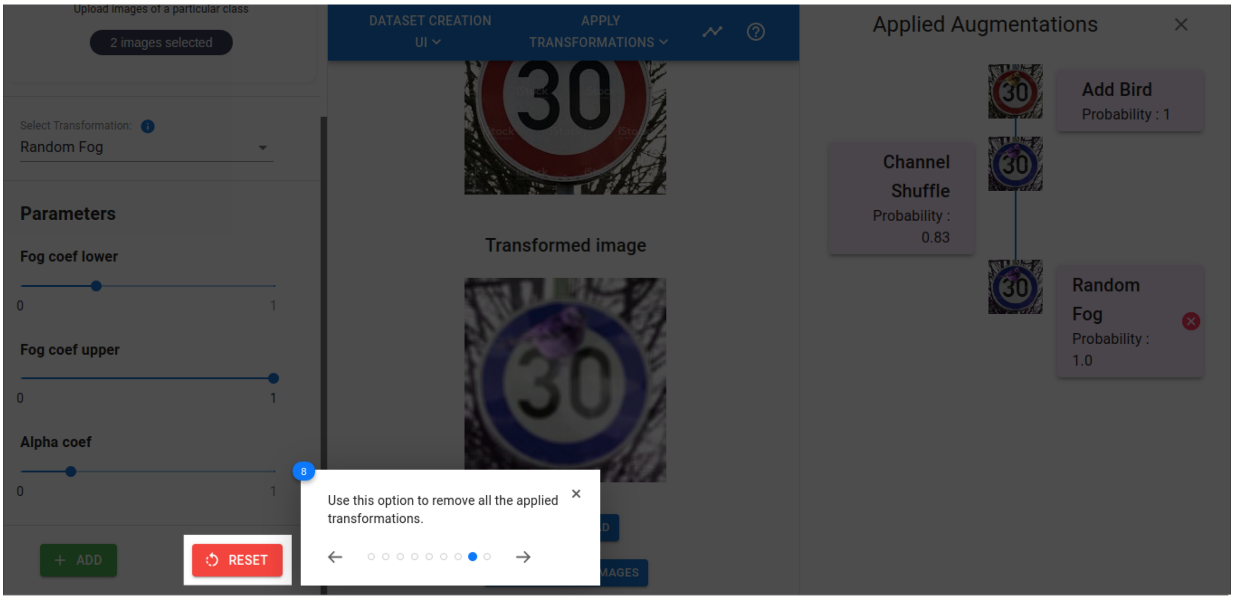
DATASET CREATION UI
APPLY TRANSFORMATIONS
Applied Augmentations
Upload images of a particular class
2 images selected
Select Transformation:
Add Bird
Parameters
Frac
0
1
Probability
0
Transformed image
No augmentations applied yet.
Apply this transformation to save it and apply more transformations.
ADD

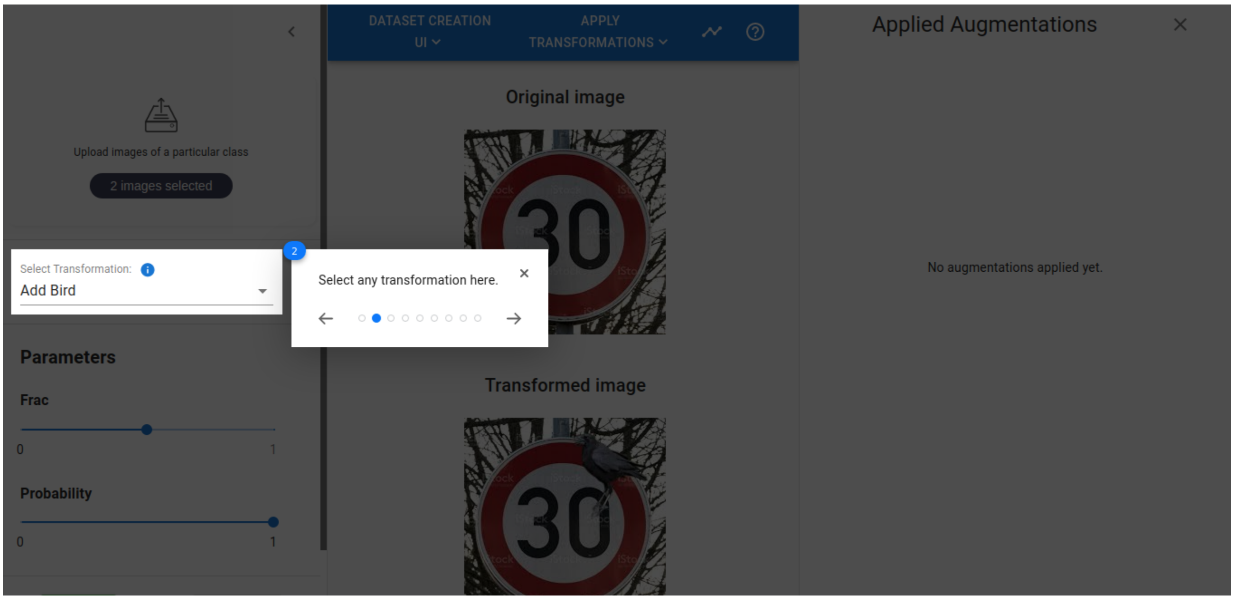

Upload images of a particular class
2 images selected
DATASET CREATION
UI
All the applied transformations, along with their previews in sequence can be viewed here. You can also remove some transformations or restore a previous stage.
Select Transformation:
Random Fog
Parameters
Fog coef lower
0
1
Fog coef upper
0
1
Alpha coef
0
1
ADD
RESET
Transformed image
Applied Augmentations
Add Bird
Probability : 1
Channel Shuffle
Probability : 0.83
Random Fog
Probability : 1.0


Upload images of a particular class
2 images selected
DATASET CREATION
UI
APPLY TRANSFORMATIONS
Applied Augmentations
Select Transformation:
Random Fog
Parameters
Fog coef lower
0
1
Fog coef upper
0
1
Alpha coef
0
1
Transformed image
Add Bird
Probability : 1
Channel Shuffle
Probability : 0.83
Random Fog
Probability : 1.0
Select this option to perform multiple iterations of the applied transformations and preview all the generated images. You can also download these images in a .zip folder or directly add to the dataset present on server.
DOWNLOAD
PREVIEW ALL IMAGES
ADD
RESET

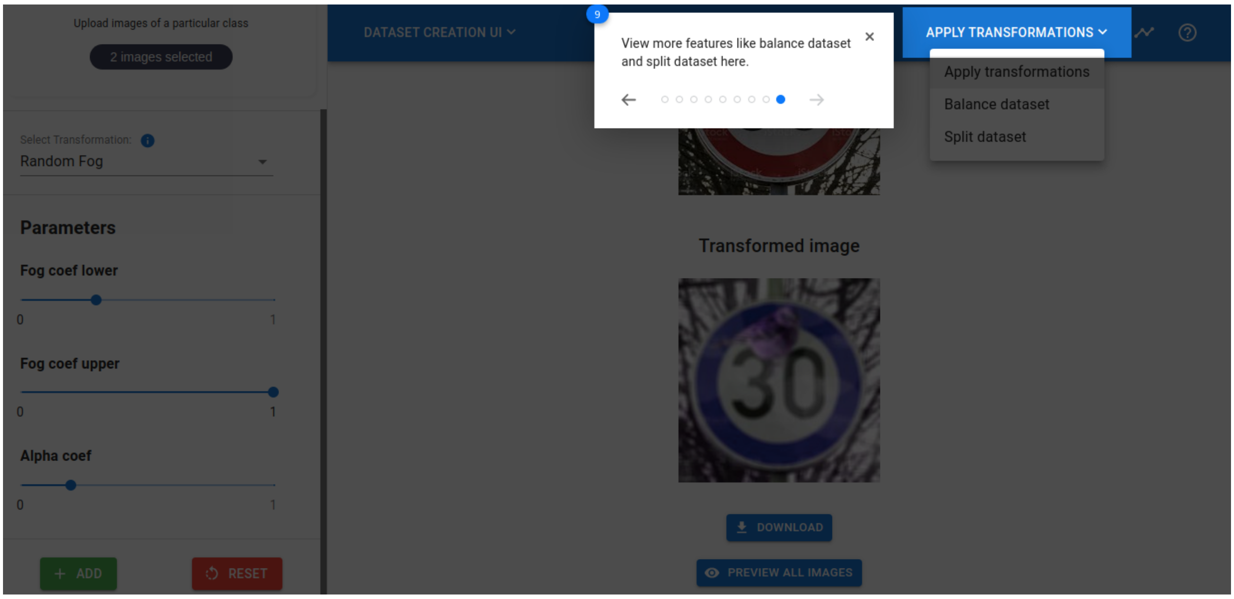

Preview
Class ID:
Speed limit (30km/h)
Number of Iterations *
7
Enter a positive number.
CANCEL
PREVIEW


Are you sure you want to save these images to dataset?
Download ZIP
DISCARD
SAVE


Upload images of a particular class
2 images selected
DATASET CREATION UI
APPLY TRANSFORMATIONS
Applied Augmentations
Select Transformation:
Random Fog
Parameters
Fog coef lower
0
1
Fog coef upper
0
1
Alpha coef
0
1
ADD
RESET
Use this option to remove all the applied transformations.
Transformed image
Add Bird
Probability : 1
Channel Shuffle
Probability : 0.83
Random Fog
Probability : 1.0



DATASET CREATION UI
APPLY TRANSFORMATIONS
Applied Augmentations
Upload images of a particular class
2 images selected
Original image
Select Transformation:
Add Bird
Select any transformation here.
No augmentations applied yet.
Parameters
Frac
0
1
Transformed image
Probability
0
1



Upload images of a particular class
2 images selected
DATASET CREATION UI
View more features like balance dataset and split dataset here.
APPLY TRANSFORMATIONS
Apply transformations
Balance dataset
Split dataset
Select Transformation:
Random Fog
Parameters
Transformed image
Fog coef lower
0
1
Fog coef upper
0
1
Alpha coef
0
1
DOWNLOAD
PREVIEW ALL IMAGES
ADD
RESET

DATASET CREATION UI
BALANCE DATASET
Before balancing
Frequency
140
120
100
80
60
40
20
0
After balancing
Frequency
140
120
100
80
60
40
20
0

Post Evaluation UI :

This tool lets users test your trained model on the test dataset and analyze various metrics. and tweak network and dataset based on the suggestions and features recommended by this feature, and decide what the next experiment can be.

Features:

- ➔ Metrics
- ➔ Class Wise accuracy on Test Data
- ➔ Top 5 Classes
- ➔ Wrongly predicted images
- ➔ Confusion Matrix
- ➔ Most K confused classes
- ➔ Precision Recall Curves
- ➔ ROC Curve
- ➔ Model Interpretation by Gradient Based Attribution

- *Metrics*:
    1. Accuracy score
    2. Precision score
    3. Recall score
    4. F1_score_macro
    5. F1_score_weighted

    Good F1 score means the model is robust and is able to handle imbalance in the dataset properly.

If train accuracy **>>** test accuracy, we get the intuition of our model getting overfitted.
To solve overfitting problems

- we can decrease the depth of the model
- simply increase the dropout probability
- we can use the model with lesser parameters
- use regularizations technique.

If train accuracy and test accuracy are both very low, the model may be underfit.
To solve the problem of underfitting

- we can increase the depth of the model
- Can train model for more epochs
- Can remove noise from the images by applying

The dataset may be imbalanced if F1 score is not as good as the accuracy. This can be handled by

- balancing the dataset by oversampling or undersampling the images.
- Penalizing misclassifying class with less images in the loss function.

● *Class Wise accuracy on Test Data:*

This feature plots accuracy, in increasing order, against corresponding class.
This plot gives us the intuition on which class our model is not able to generalize i.e. giving the wrong result. Improving their performance will improve the overall metrics.
This can be done by

- identify general patterns by visual inspection in images of these classes, which is causing the failure.
- add more images of that class in training by surveying or by augmentation.
- Penalizing misclassifying these classes in the loss function.

● *Top 5 Classes:*

Lists out all the top 5 classes that have been predicted with the highest confidence
For each example, the model outputs a series of numbers called the confidence that communicates how strongly it associates each label with that example. If the number is high, the model has high confidence that the label should be applied to that example.

● *Wrongly predicted images:*

Lists out all the images in the dataset that have been wrongly classified.

This can be an effective visual tool for the data scientist for pointing out the reasons behind the images being classified wrongly due to unrepresentative data samples like:

- Unseen augmentation
- ‘New’ images of the class not present in the training set
- Adversarial attacks

Sometimes a visual inspection can identify patterns that you can then correct by adding more training data or modifying existing training data.

- *Confusion Matrix :*

  A Confusion matrix is plotted which is an N x N matrix where N is the number of target classes. The matrix compares the actual target values with those predicted by the machine learning model. This gives us a holistic view of how well our classification model is performing and what kinds of errors it is making. It is a visual way to inspect the performance of a classification model.

- *Most K confused classes -*

  It gives the top k most confused pair of classes and the percentage of samples of class A that are misclassified as class B.HIgher the number of misclassified samples means that model is considering Class A as Class B.

  For the pairs where this problem is severe, the following can be done to improve the metrics:

  - Increasing number of training Image of Class A by surveying or by augmentation.
  - Penalizing misclassifying class A image in the loss function.
  - Using test-time augmentations.

- *Precision Recall Curves:*

  Precision and Recall Curve: This feature plots the Precision and Recall of the model at various thresholds for a chosen class.
  Precision Vs Recall Curve: This feature plots Precision versus Recall of the model calculated at different thresholds for a chosen class.

  A high area under the Precision vs Recall curve represents both high recall and high precision. High scores for both show that the classifier is returning accurate results (high precision), as well as returning a majority of all positive results (high recall).
  An ideal system with high precision and high recall will return many results, with all results labeled correctly.

- *ROC Curve:*

  This function plots the True Positive rate (TPR) Vs False Positive Rate (FPR) of the model for various thresholds for a chosen class.

  It helps us to assess the discriminative capability of the model independent of the classification threshold. The worst case possibility is when the curve is along the diagonal from 0 to 1 which occurs when the model classifies all samples into either class with equal probability (complete confusion). The best case scenario is when the TPR is 1 for all values of FPR which gives the maximum area under the curve.

- *Model Interpretation by Gradient Based Attribution :*

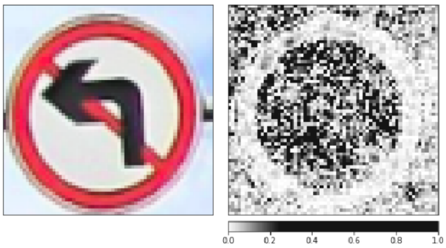


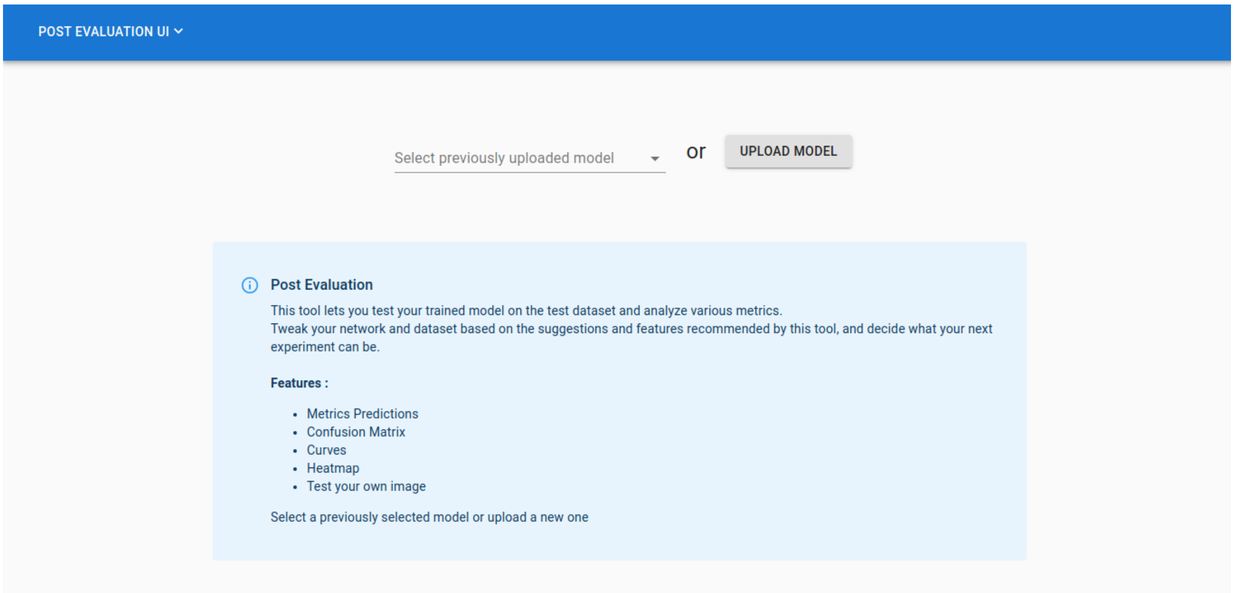
Fig: Image, Generated Result

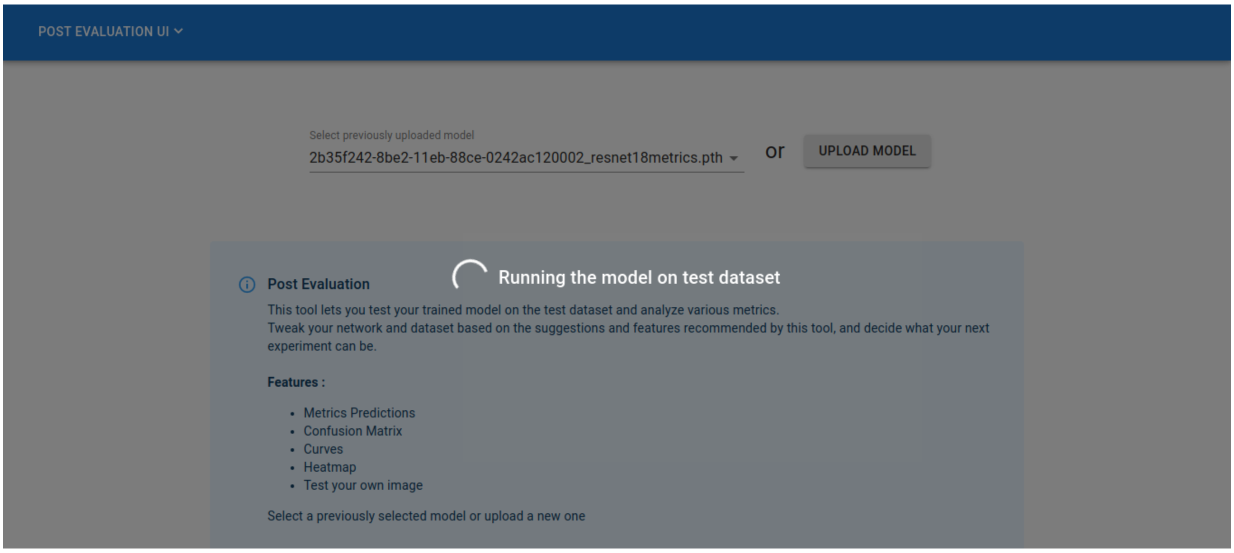
  This feature takes an input image and generates its gradient based attribution and the user can then visualize where the model focuses.

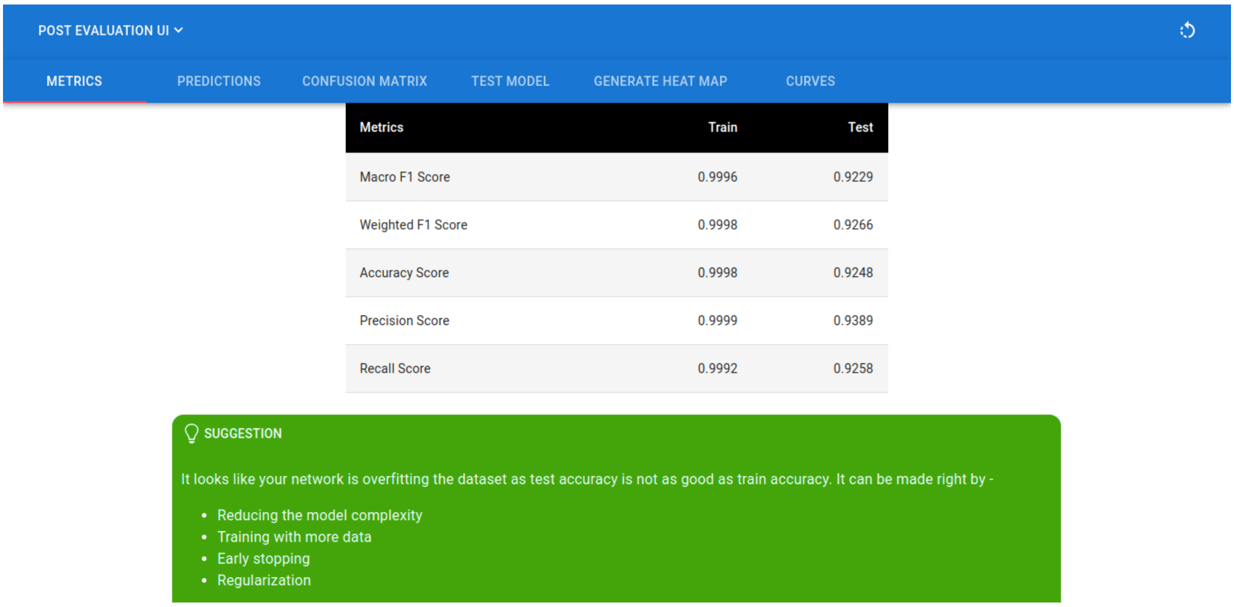
  Metrics only tell of a model's predictive decisions. Over time, the performance might change due to model concept drift caused by various factors in the environment. Hence, it is of paramount importance to understand what drives a model to take certain decisions. Model interpretation helps us to interpret how our model approaches the decisions, where it focuses on the input Images.

  This aims to answer the following questions :
  - What drives model predictions?
  - Why did the model take a certain decision?
  - How can we trust model predictions?

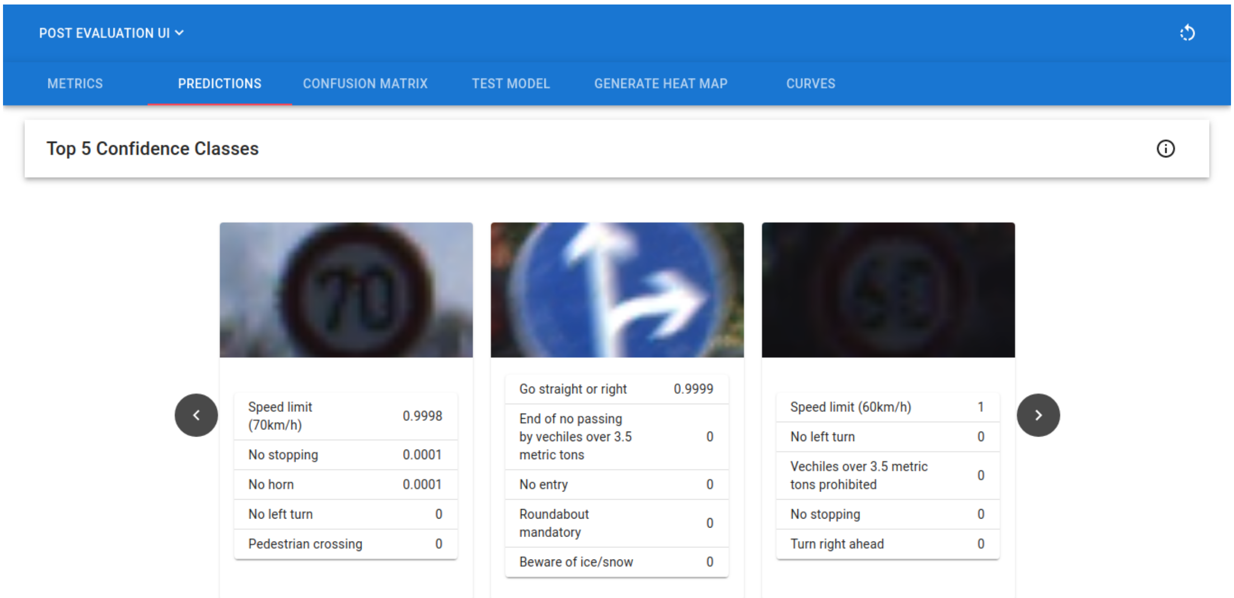
Screenshots:

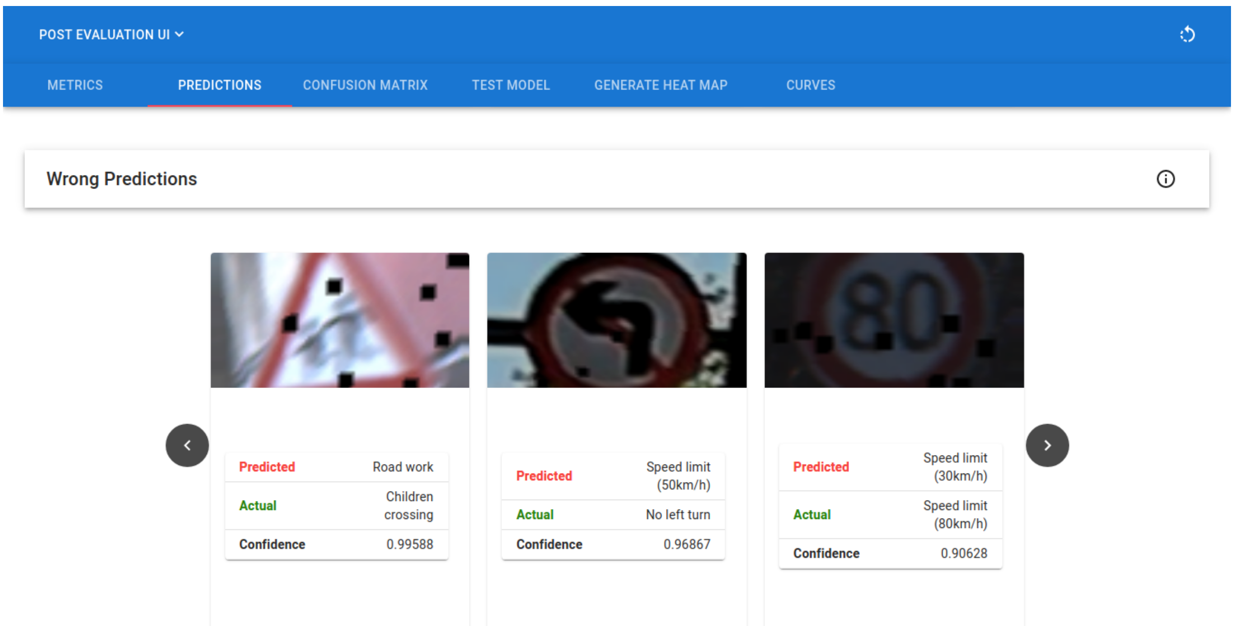

POST EVALUATION UI
Select previously uploaded model
or
UPLOAD MODEL
Post Evaluation
This tool lets you test your trained model on the test dataset and analyze various metrics.
Tweak your network and dataset based on the suggestions and features recommended by this tool, and decide what your next experiment can be.
Features :
Metrics Predictions
Confusion Matrix
Curves
Heatmap
Test your own image
Select a previously selected model or upload a new one


POST EVALUATION UI
Select previously uploaded model
2b35f242-8be2-11eb-88ce-0242ac120002_resnet18metrics.pth
or
UPLOAD MODEL
Post Evaluation
Running the model on test dataset
This tool lets you test your trained model on the test dataset and analyze various metrics.
Tweak your network and dataset based on the suggestions and features recommended by this tool, and decide what your next experiment can be.
Features :
Metrics Predictions
Confusion Matrix
Curves
Heatmap
Test your own image
Select a previously selected model or upload a new one


POST EVALUATION UI
METRICS
PREDICTIONS
CONFUSION MATRIX
TEST MODEL
GENERATE HEAT MAP
CURVES
Metrics Train Test
Macro F1 Score 0.9996 0.9229
Weighted F1 Score 0.9998 0.9266
Accuracy Score 0.9998 0.9248
Precision Score 0.9999 0.9389
Recall Score 0.9992 0.9258
SUGGESTION
It looks like your network is overfitting the dataset as test accuracy is not as good as train accuracy. It can be made right by -
Reducing the model complexity
Training with more data
Early stopping
Regularization


POST EVALUATION UI
METRICS
PREDICTIONS
CONFUSION MATRIX
TEST MODEL
GENERATE HEAT MAP
CURVES
Class-wise Accuracy on Test data
Accuracy
100
90
80
70
60
50
40
30
20
10
0
Speed limit (80km/h)
General caution
Traffic signals
No horn
No left turn
Turn right ahead
End of all speed and passing limits
Slippery road
Bumpy road
Right-of-way at the next intersection
Stop
Speed limit (70km/h)
Speed limit (100km/h)
Dangerous curve to the left
Children crossing
Yield
Go straight or left
Go straight or right
Ahead only
End of no passing
Roundabout mandatory
End of no passing by vechiles over 3.5 metric tons
Turn left ahead
Keep right
Wild animals crossing
Beware of ice/snow
Car
Bicycles crossing
Keep left
Speed limit (20km/h)
Road work
Speed limit (30km/h)
Speed limit (50km/h)
Speed limit (60km/h)
End of speed limit (80km/h)
Speed limit (120km/h)
No passing for vechiles over 3.5 metric tons
No passing
Priority road
Vechiles over 3.5 metric tons prohibited
No vechiles
No entry
Dangerous curve to the right
Double curve
Pedestrian crossing
Road narrows on the right
Pedestrians
No stopping
SUGGESTION
"Speed limit (80km/h)" and "General caution" are lowest in accuracy. Improving their performance will improve the overall metrics. This can be done by :
Identifying general patterns by visual inspection in images of these classes, which is causing the failure.
Adding more images of these classes in training by surveying or by augmentation.
Penalizing misclassifying these classes in the loss function


POST EVALUATION UI
METRICS
PREDICTIONS
CONFUSION MATRIX
TEST MODEL
GENERATE HEAT MAP
CURVES
Top 5 Confidence Classes
Speed limit (70km/h) 0.9998
No stopping 0.0001
No horn 0.0001
No left turn 0
Pedestrian crossing 0
Go straight or right 0.9999
End of no passing by vechiles over 3.5 metric tons 0
No entry 0
Roundabout mandatory 0
Beware of ice/snow 0
Speed limit (60km/h) 1
No left turn 0
Vechiles over 3.5 metric tons prohibited 0
No stopping 0
Turn right ahead 0


POST EVALUATION UI
METRICS
PREDICTIONS
CONFUSION MATRIX
TEST MODEL
GENERATE HEAT MAP
CURVES
Wrong Predictions
Predicted Road work
Actual Children crossing
Confidence 0.99588
Predicted Speed limit (50km/h)
Actual No left turn
Confidence 0.96867
Predicted Speed limit (30km/h)
Actual Speed limit (80km/h)
Confidence 0.90628

POST EVALUATION UI

METRICS | PREDICTIONS | CONFUSION MATRIX | TEST MODEL | GENERATE HEAT MAP | CURVES

Most Confused Classes

SUGGESTION

"Speed limit (80km/h)" is classified as "Speed limit (30km/h)" the most number of times.The following can be done to remove the confusion:

- Increasing number of training images of "Speed limit (80km/h) " by surveying or by augmentation.
- Penalizing misclassifying "Speed limit (80km/h)" image in the loss function.

- 50.0% images of **Speed limit (80km/h)**are being predicted to be of **Speed limit (30km/h)**.

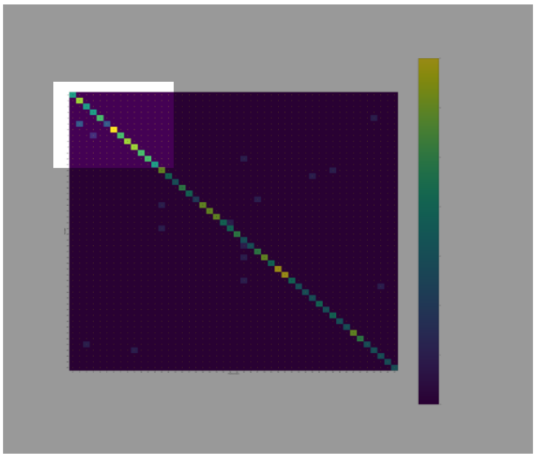
Speed limit (80km/h)

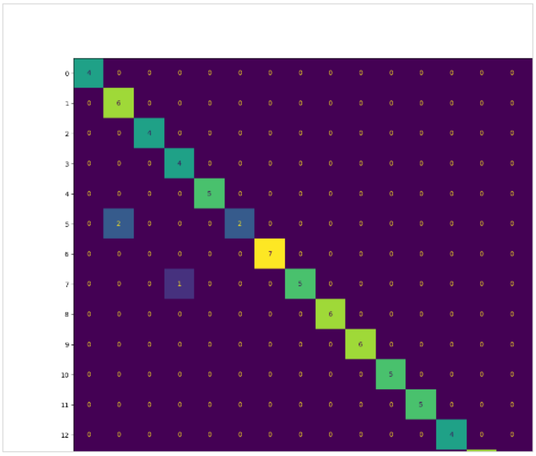
Speed limit (30km/h)

POST EVALUATION UI

METRICS | PREDICTIONS | CONFUSION MATRIX | TEST MODEL | GENERATE HEAT MAP | CURVES

Confusion Matrix



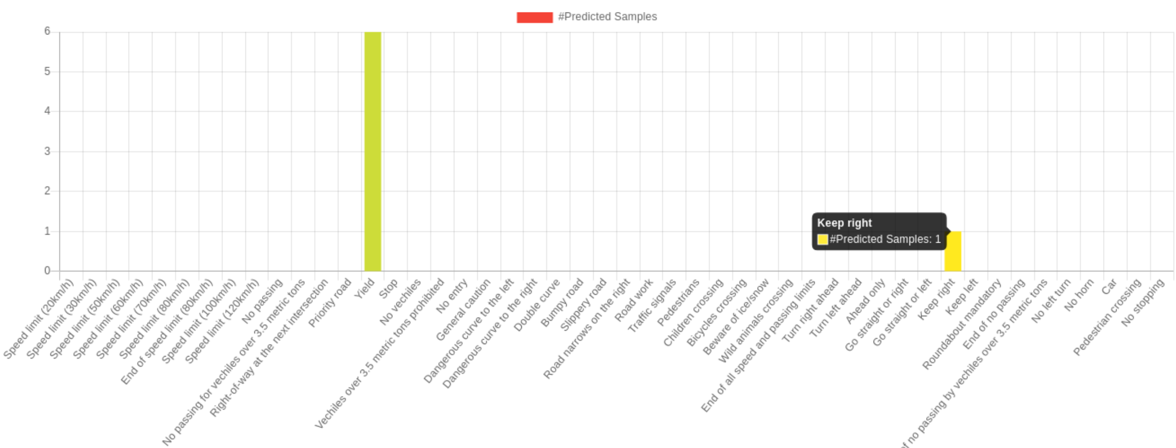


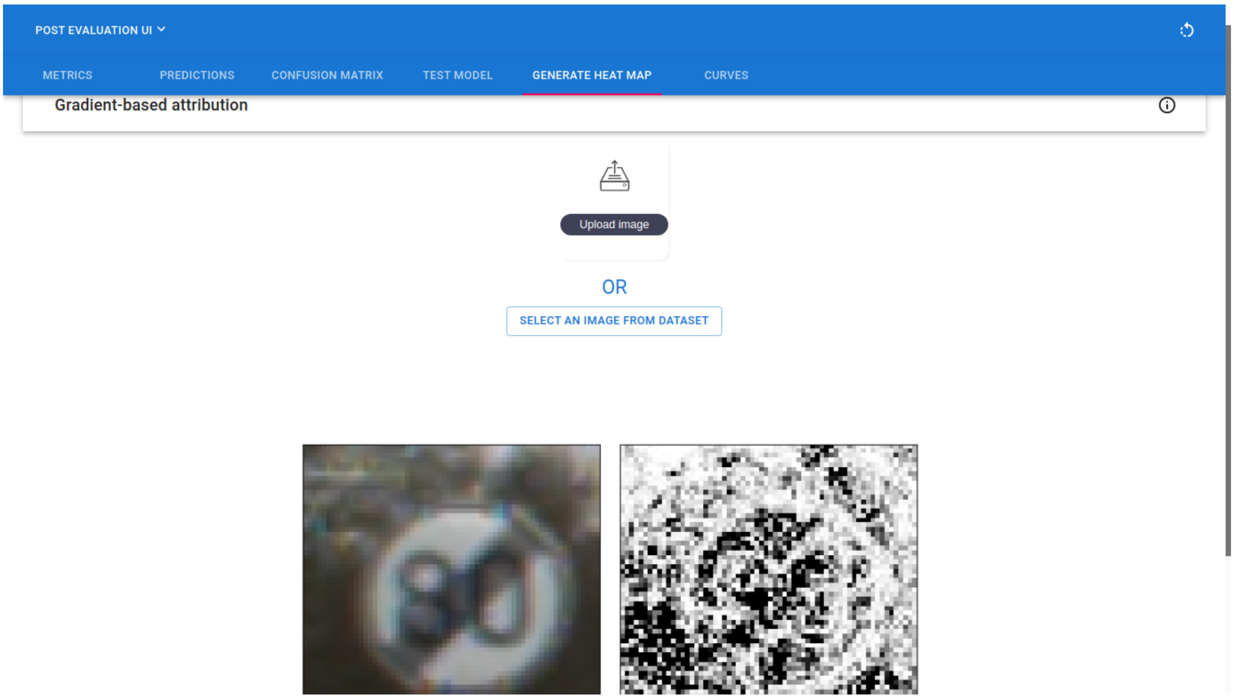
POST EVALUATION UI

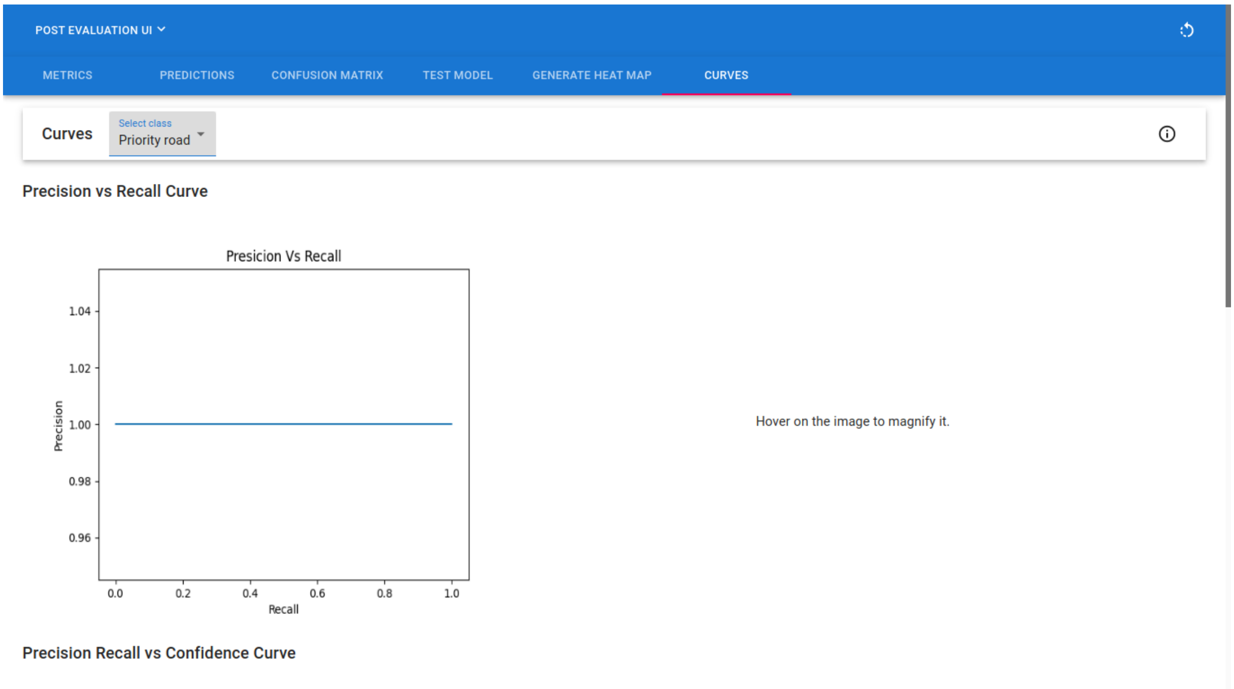
METRICS | PREDICTIONS | CONFUSION MATRIX | TEST MODEL | GENERATE HEAT MAP | CURVES

Confusion of selected class vs all classes

Select Class
Yield




POST EVALUATION UI
METRICS
PREDICTIONS
CONFUSION MATRIX
TEST MODEL
GENERATE HEAT MAP
CURVES
Gradient-based attribution
Upload image
OR
SELECT AN IMAGE FROM DATASET

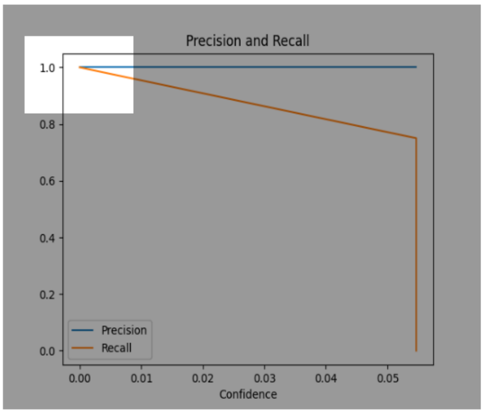

POST EVALUATION UI
METRICS
PREDICTIONS
CONFUSION MATRIX
TEST MODEL
GENERATE HEAT MAP
CURVES
Curves
Select class
Priority road
Precision vs Recall Curve
Presicion Vs Recall
1.04
1.02
1.00
0.98
0.96
Precision
0.0
0.2
0.4
0.6
0.8
1.0
Recall
Hover on the image to magnify it.
Precision Recall vs Confidence Curve

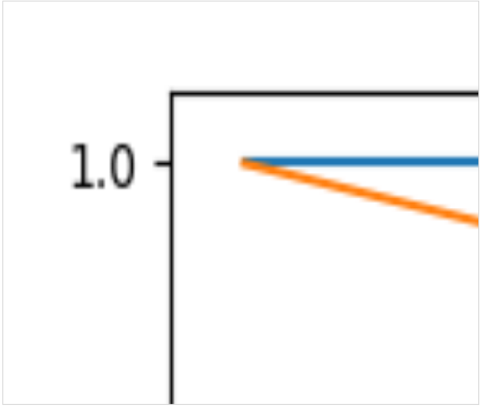
POST EVALUATION UI

METRICS | PREDICTIONS | CONFUSION MATRIX | TEST MODEL | GENERATE HEAT MAP | CURVES

Recall

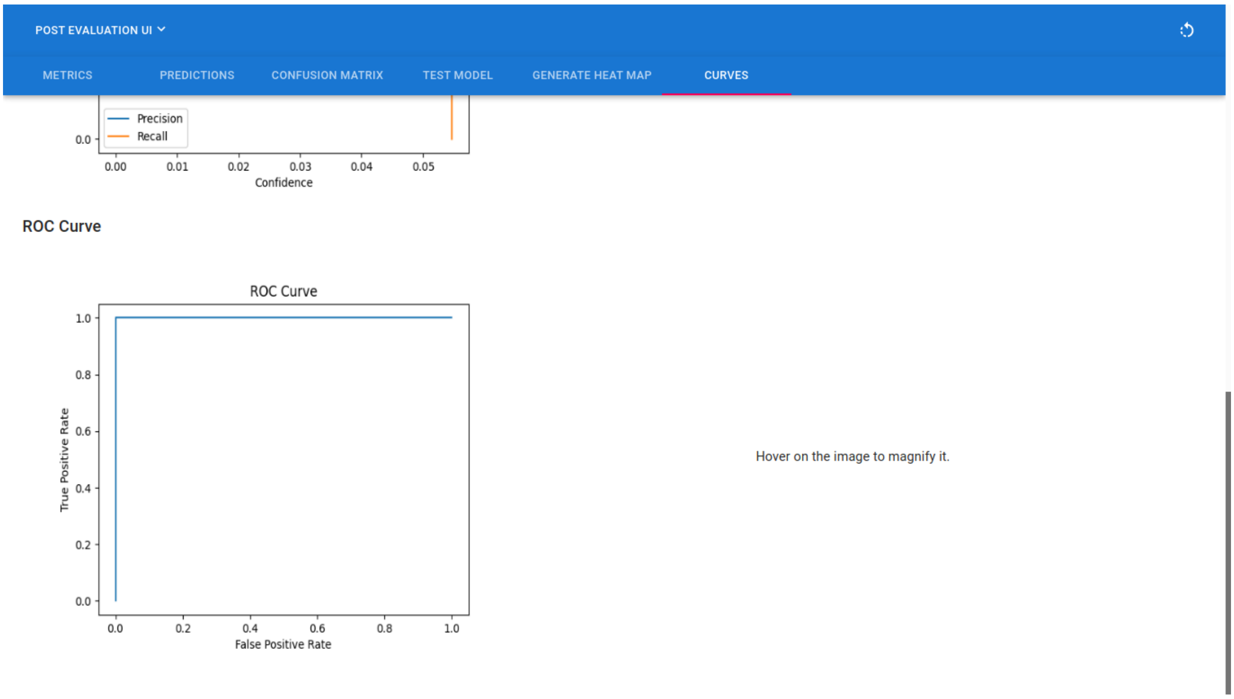
Precision Recall vs Confidence Curve

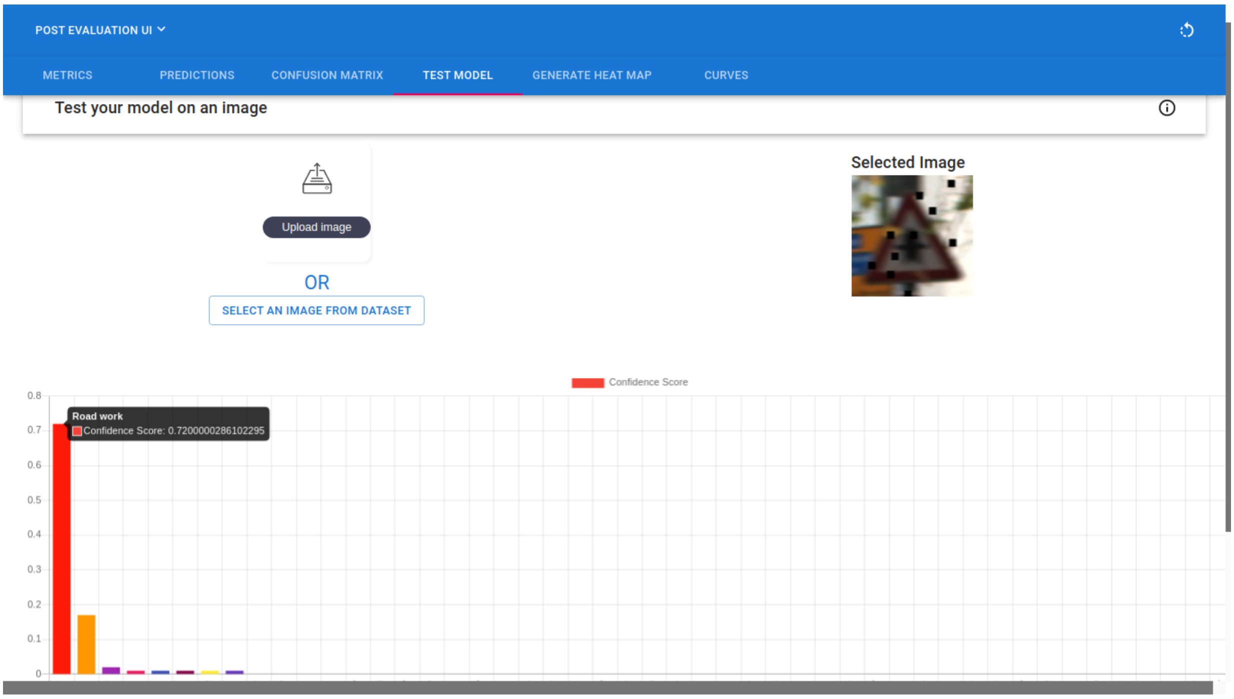




ROC Curve

ROC Curve


POST EVALUATION UI
METRICS
PREDICTIONS
CONFUSION MATRIX
TEST MODEL
GENERATE HEAT MAP
CURVES
Precision
Recall
0.0
0.00
0.01
0.02
0.03
0.04
0.05
Confidence
ROC Curve
ROC Curve
1.0
0.8
0.6
0.4
0.2
0.0
True Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
False Positive Rate
Hover on the image to magnify it.


POST EVALUATION UI
METRICS
PREDICTIONS
CONFUSION MATRIX
TEST MODEL
GENERATE HEAT MAP
CURVES
Test your model on an image
Upload image
OR
SELECT AN IMAGE FROM DATASET
Selected Image
Confidence Score
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
Road work
Confidence Score: 0.7200000286102295