# Traffic Sign Detection

Manuel Scurti & Mingjie Ye

# Traffic Sign Recognition and Box Regressor Task

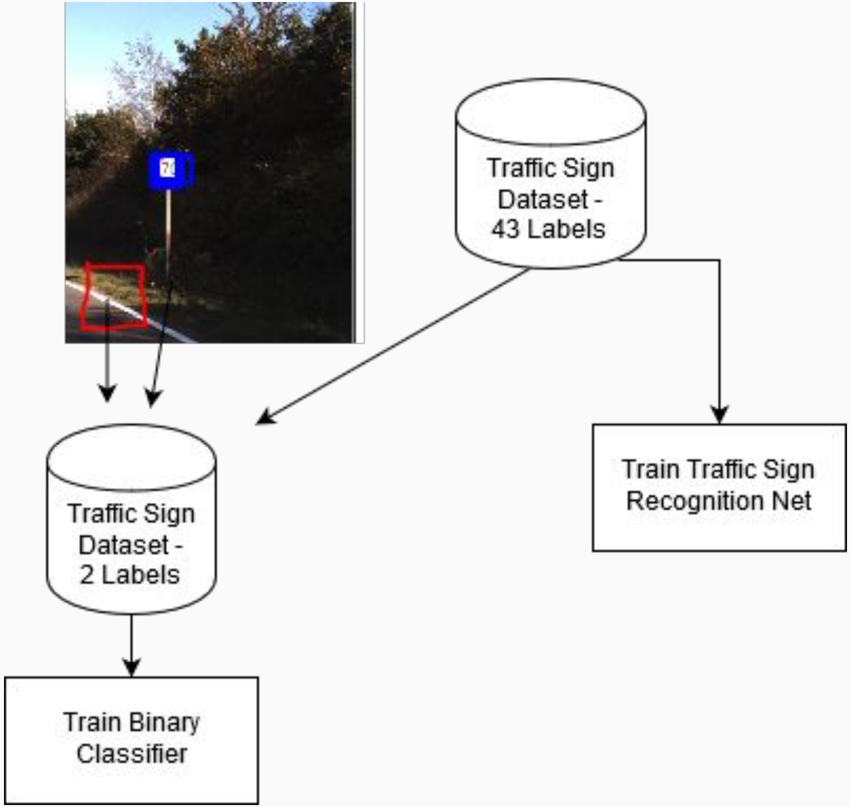
The german traffic sign benchmark (GTSRB) is a multi-class classification challenge. The benchmark has the following properties:

- Multi-class classification problem
- More than 40 classes
- More than 50000 images in total

Each traffic sign is provided with its own bounding box, so that it becomes a full traffic sign detection task.

# Implementation

- Two neural network involved
  - Binary classifier NN
  - InceptionV3 for traffic sign classification
- Binary classifier used to tell whether a box proposal is a traffic sign or not
- InceptionV3 used to determine the type of traffic sign


Traffic Sign Dataset - 43 Labels
Traffic Sign Dataset - 2 Labels
Train Traffic Sign Recognition Net
Train Binary Classifier

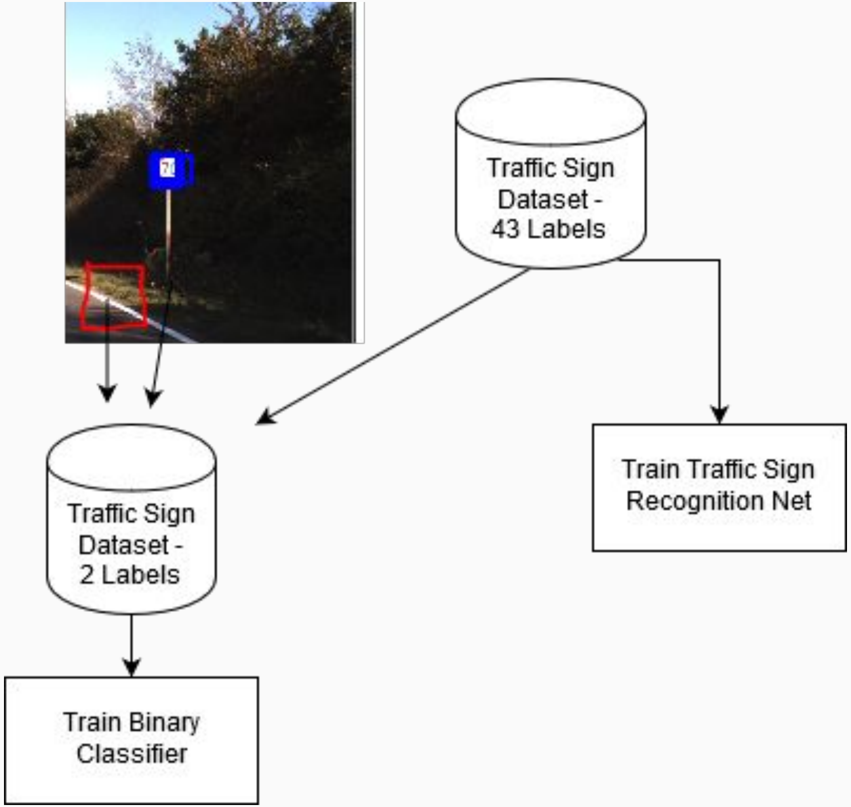
# Implementation - Creating the dataset for the binary classifier

For each original train image:

1. Take ground truth crop and label it as “traffic sign”
2. Take random crop and label it as “no traffic sign”

How to limit random crops that include the traffic sign?

1. Divide the image in 4 regions
2. Determine in which region is the ground truth
3. Take the random crop from any of the remaining 3 regions



Traffic Sign Dataset - 43 Labels

Traffic Sign Dataset - 2 Labels

Train Traffic Sign Recognition Net

Train Binary Classifier

# Implementation - Training the binary classifier

- Structure of the binary classifier: CNN with 6 conv layers and 1 FC layer

| |
|---|
| Conv2D 3x3, 32 maps, stride 1x1 + BatchNorm + Leaky ReLU activation function |
| Conv2D 3x3, 32 maps, stride 1x1 + BatchNorm + Leaky ReLU activation function |
| MaxPooling 2x2 |
| Conv2D 3x3, 32 maps, stride 1x1 + BatchNorm + Leaky ReLU activation function |
| Conv2D 3x3, 32 maps, stride 1x1 + BatchNorm + Leaky ReLU activation function |
| MaxPooling 2x2 |
| Conv2D 3x3, 32 maps, stride 1x1 + BatchNorm + Leaky ReLU activation function |
| Conv2D 3x3, 32 maps, stride 1x1 + BatchNorm + Leaky ReLU activation function |
| Dense 512 + BatchNorm + LeakyReLU activation function |
| Dense 2 + Softmax activation function |

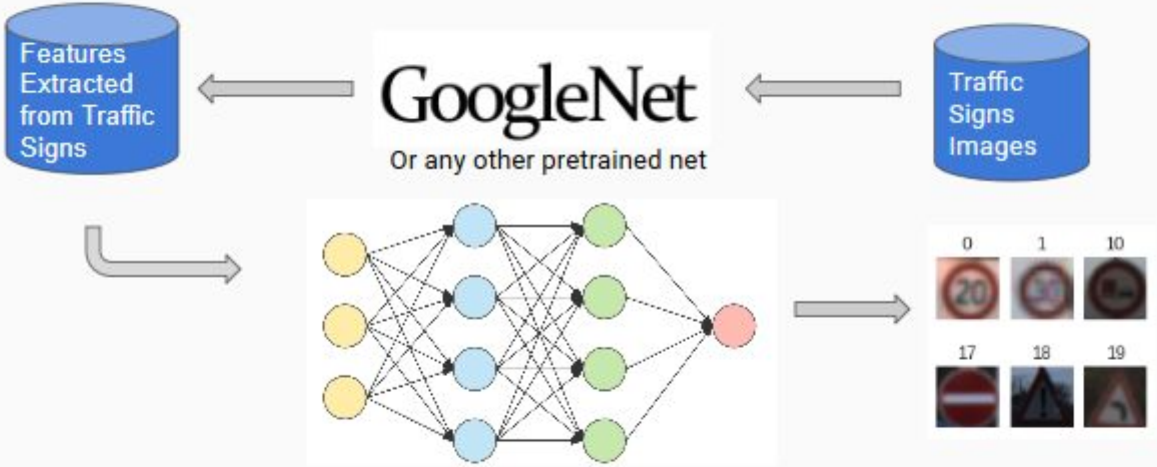
Results:
**10** epochs
**94%** accuracy
**0.13** inference time (seconds)

Train on 1003 samples, validate on 426 samples

Images scaled at 48x48

# Implementation - Using Transfer Learning from InceptionV3 to train the classifier

1. Take a pretrained model without fully connected layers
2. Train the pretrained model using traffic signs images for few epochs
3. Use it as a bottleneck features extractor
4. Train for few epochs a fully connected model from scratch using this dataset

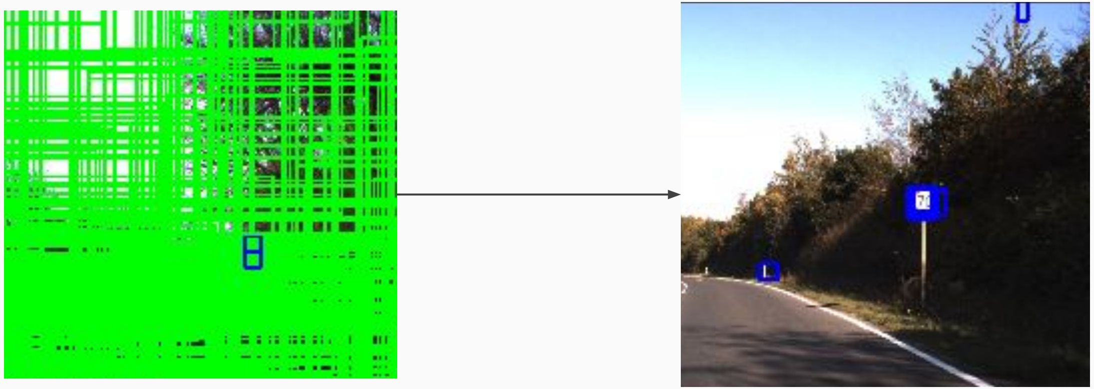

Features Extracted from Traffic Signs
GoogleNet
Or any other pretrained net
Traffic Signs Images
0
1
10
20
17
18
19

Results: **98%** accuracy, **1.31** inference time (seconds)

# Implementation - Processing images

For each box proposal:

1. Determine if the box proposal contains a traffic sign or not with the binary classifier
2. If yes, determine which kind of traffic sign is in the proposal using the Inception NN
3. Save the results



# Results

- Intermediate **mAP of 31.05%** achieved with an average of around 2 minutes of processing per image
- Process stopped for excessive use of RAM

Other approaches followed:

- Softmax threshold - FAIL - slow process, hard to find a reasonable threshold value
- Add a new class “no-traffic-sign” to the original classification model - FAIL

Thank you!

Questions?