Carro x AWS Hackathon 2022

Secret Garage Group

Lee Hua, Philip, Vibin, Wai Yeong, Yong Hao

01

License Plate Extraction

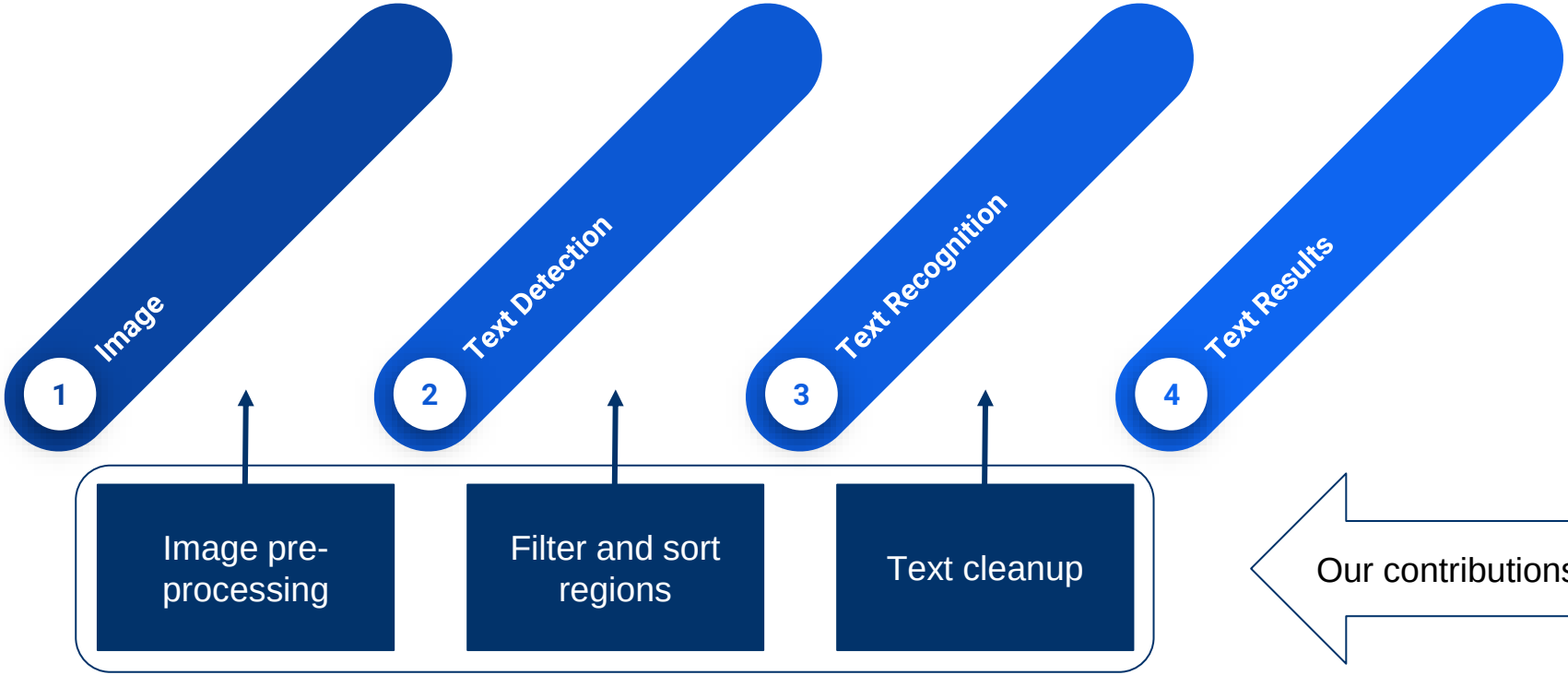
Computer Vision: Problem Statement 1

Motivation

Problem: Extract text from images of license plates

- Automated barrier free car park
- Surveillance
- Inventory management

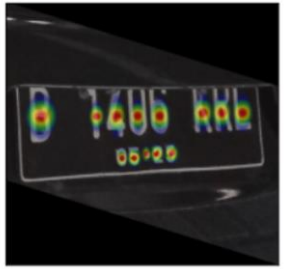
Pipeline


1
Image
2
Text Detection
3
Text Recognition
4
Text Results
Image pre-processing
Filter and sort regions
Text cleanup
Our contributions

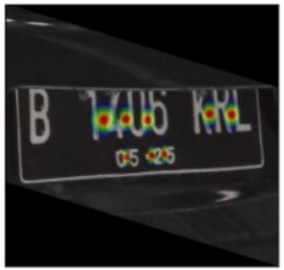
Text Detection

CRAFT: Character-Region Awareness For Text detection

Region Map



Affinity Map



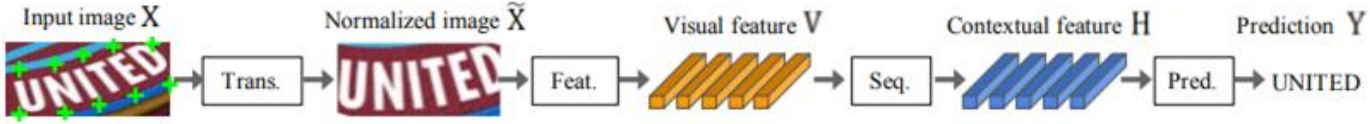
Minimum Bounding Rectangles

Text Recognition

(1) Deep Text (2) Easy OCR (3) Ensemble

DeepText model: TPS-ResNet-BiLSTM-Attn

Source: https://arxiv.org/abs/1904.01906



TPS : thin-plate spline transformation to improve geometric invariance
ResNet : CNN backbone for feature extraction
BiLSTM : sequence modeling, contextual information for robust prediction
Attn : attention-based sequence prediction

Idea

Projective transform:
Document scanner

Available in market
- Eg. Office Lens

Idea

Projective transform:
License plate

Corners obtained from license plate detector model

Exploratory Data Analysis

Challenges

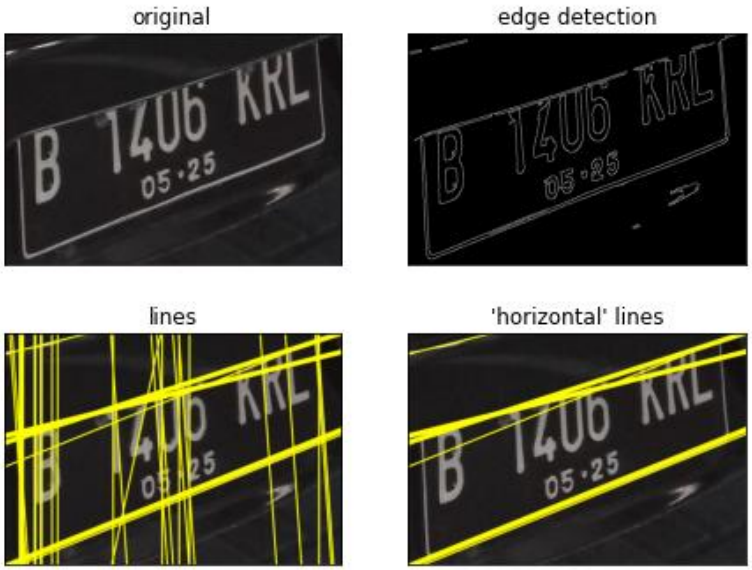
Our focus was on **pre-processing**! Since the actual text detection and recognition are quite good

Cropped license plate region

Rectangle 'borders' not clear

Strategy #1

Image pre-processing - Horizontal image alignment



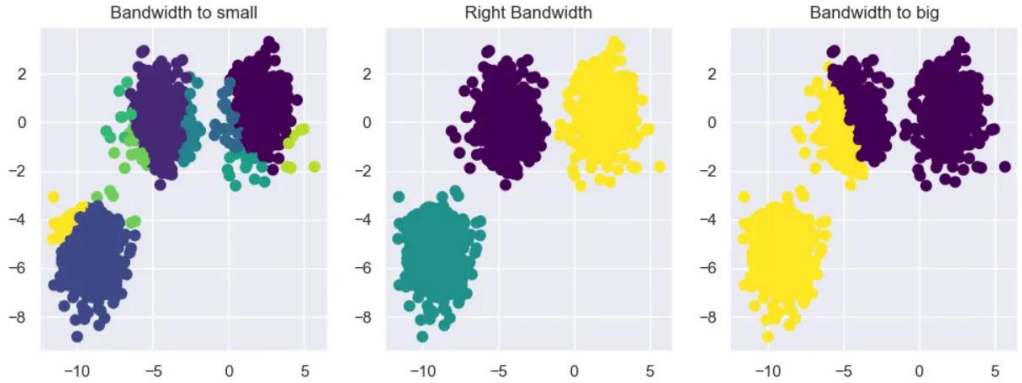
Also serves as data augmentation

Text Detection

Mean Shift Clustering

Unsupervised technique for clustering

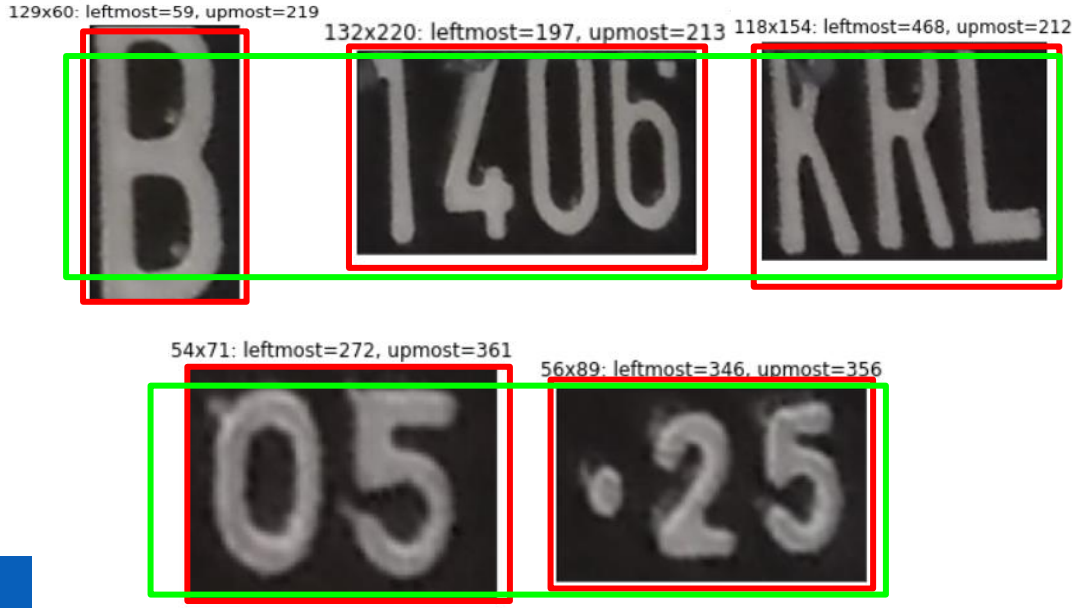
- # clusters do not need to be given
- Only 1 parameter: bandwidth
- img_diag → good indicator



Bandwidth selection: img_diag/20

Strategy #2

Text sorting - Mean shift clustering



1-D feature:
Upmost y-coordinate

Strategy #3

Text filtering - Mean shift clustering

Cluster with largest heights

1-D feature: Cropped height

Text Recognition

'B' : 0.9932
'14Ub' : 0.3722
'KKL' : 0.7105

Prediction: B14UbKKL

Strategy #4

Text cleanup - Capitalisation, Regular Expressions

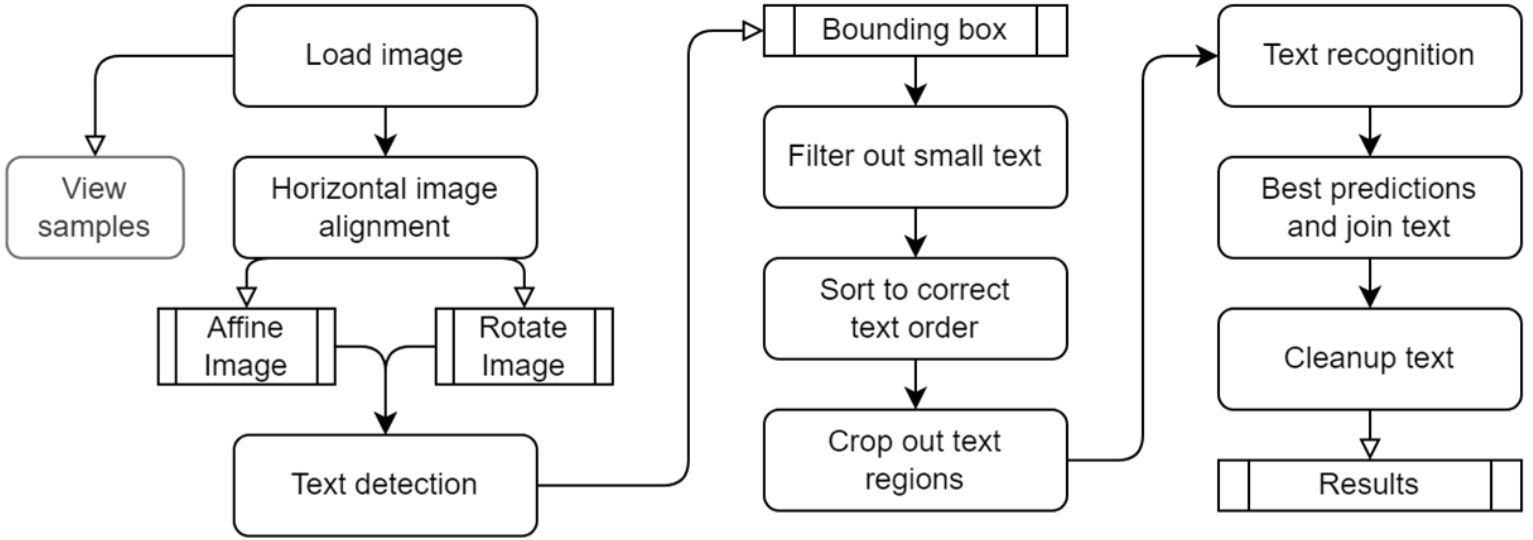
License plate only has alphanumeric capital characters.

```
re.sub('[^A-Z0-9]', '', text.upper())
```

Eg. B34.52s! → B3452S

Implementation

`get_text_preds(img)`



RESULTS

Image	Mean Levenshtein distance		
	DeepText	EasyOCR	Ensemble: DeepText + EasyOCR
Original	0.8505	0.7712	0.8415
Affine	0.9518	0.8595	0.9422
Rotate	0.9433	0.8403	0.9335
Ensemble: Affine + Rotate	0.9511	0.8620	0.9404

Future Work

Thresholding and dilation to help with faint text (w.r.t background)

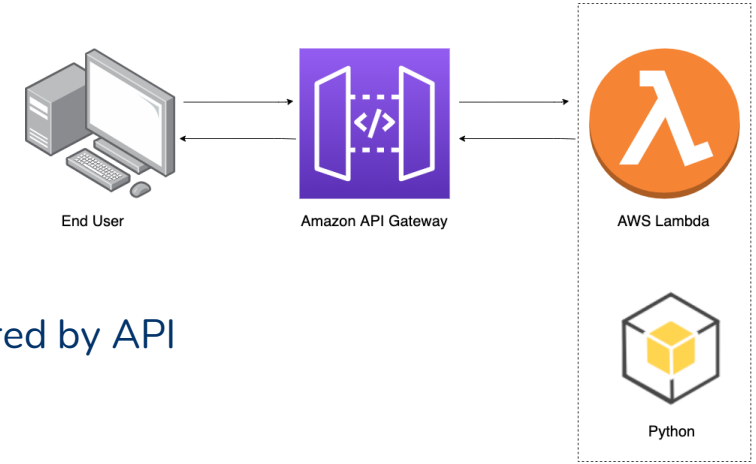
Future Work

Resize to fixed size input model

- Optimized for tflite implementation on CPU based server

Amazon Serverless Computing

- Stream images through Amazon API Gateway
- AWS Lambda runs script when triggered by API
- Sends results back to end user



02

Engine Issue Detection

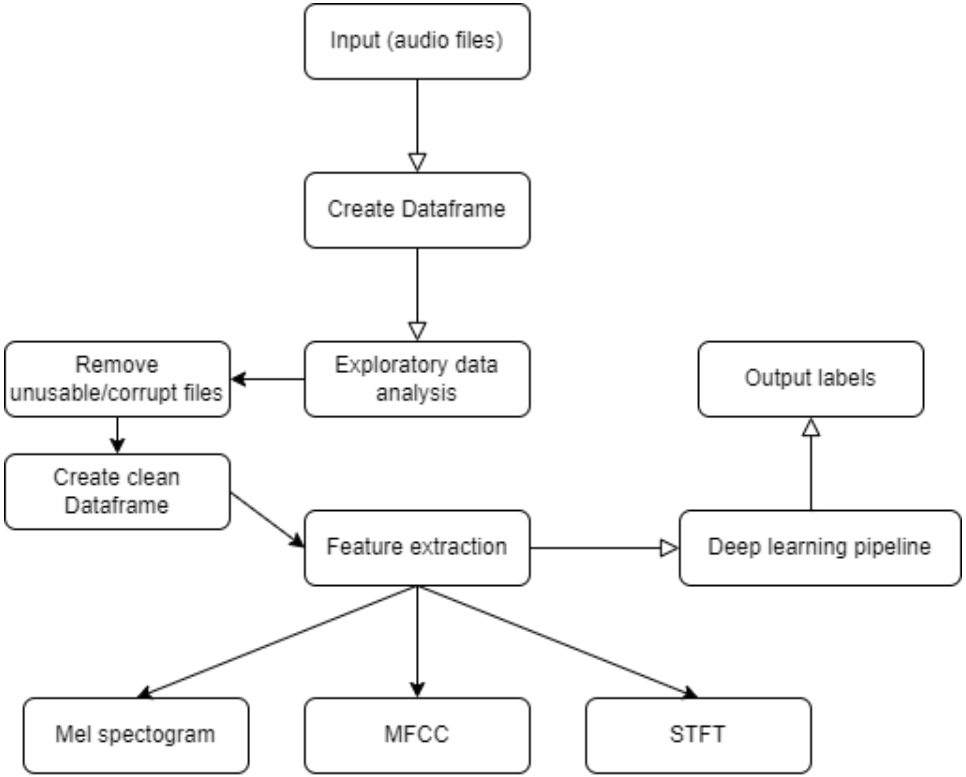
Acoustic Engineering: Problem Statement 2

Motivation

Problem: Can we detect car engine issues based on only engine sounds?

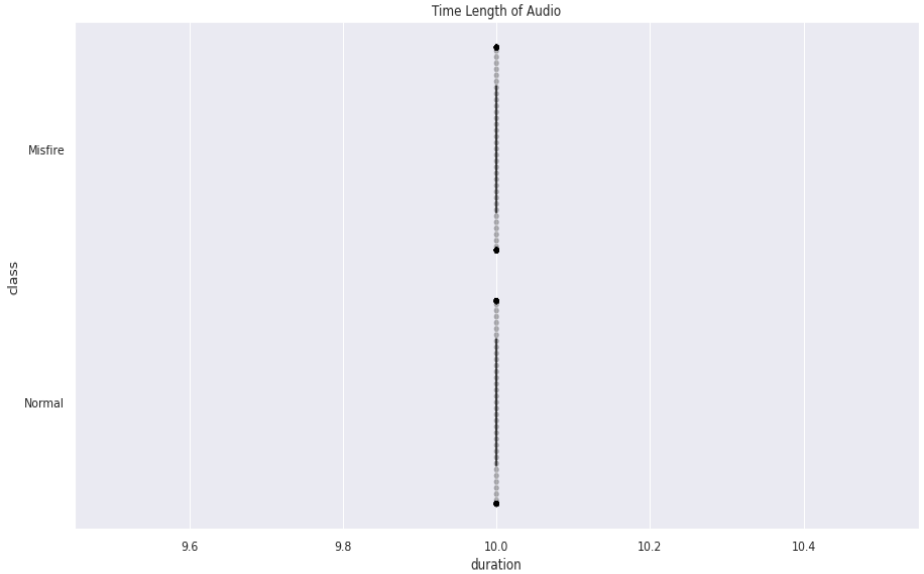
Task: Build a binary classification model to detect car engine issues

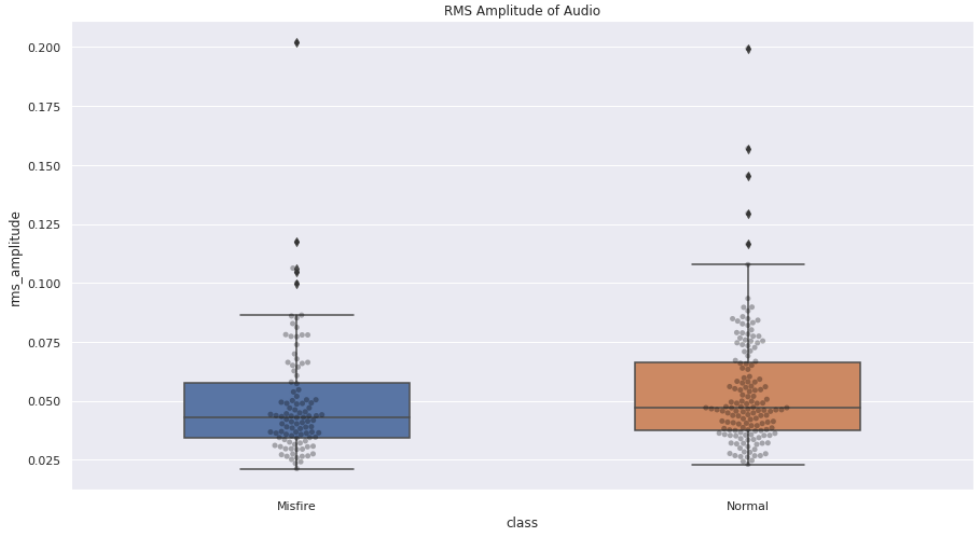
Pipeline


Input (audio files)
Create Dataframe
Exploratory data analysis
Remove unusable/corrupt files
Create clean Dataframe
Feature extraction
Output labels
Deep learning pipeline
Mel spectogram
MFCC
STFT

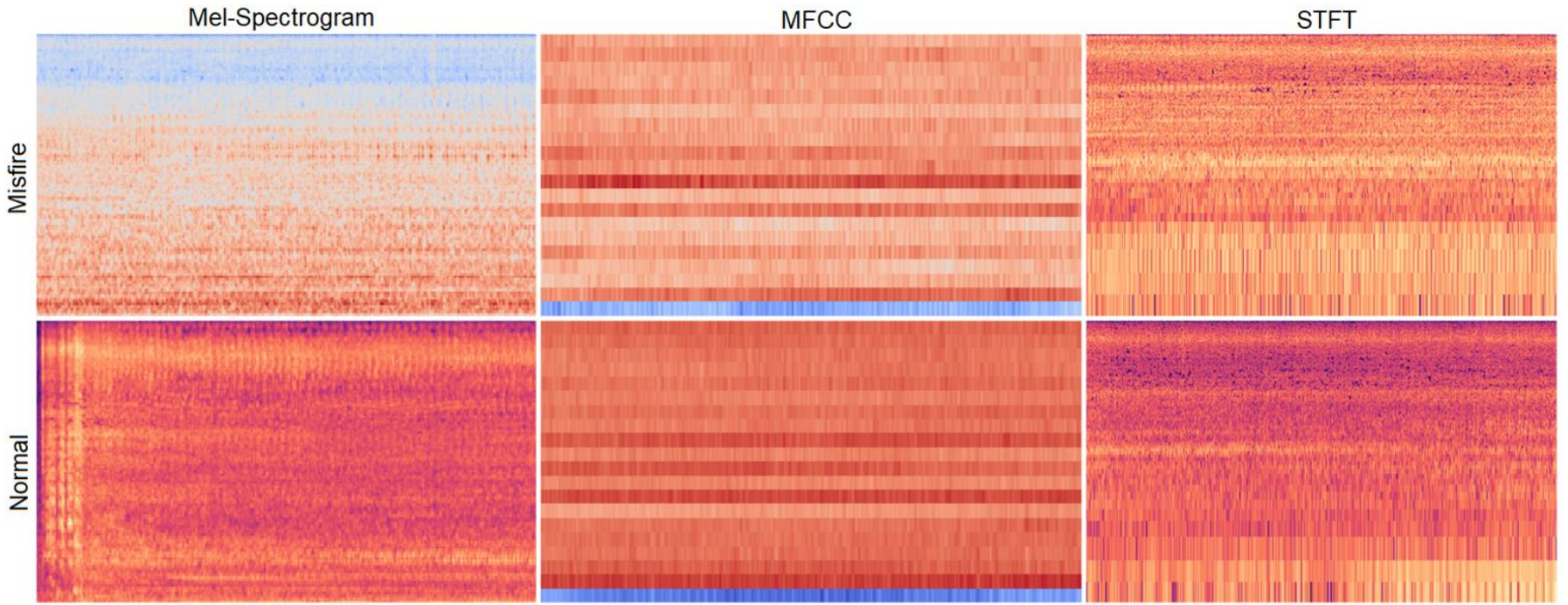
Exploratory Data Analysis

- Time Length
- RMS Amplitude





Feature Extraction



Feature Extraction

- STFT outperforms the MFCC and Mel-spectrogram, based on experimental data

	Model		
Features	LogReg(Baseline)	CNN (DL)	CNN-LSTM
MFCC	0.588	0.4706	0.588
MEL-SPEC	0.588	0.5294	0.588
STFT	0.6275	0.6275	0.588

Paper Reference

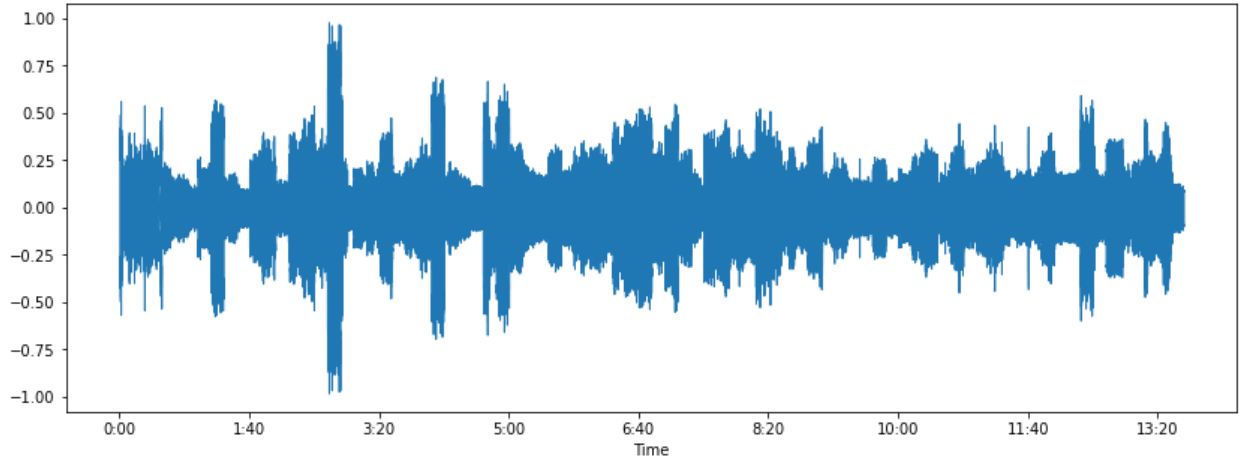
M. Lasseck, Audio-based Bird Species Identification with Deep Convolutional Neural Networks., in: CLEF (Working Notes), 2018.

- Concatenate audio signal in time domain by class labels
- Extract audio chunks from file with a duration of ca. 5 seconds
- Apply short-time Fourier transform

Strategy #1

Time concatenation of audio signals by label

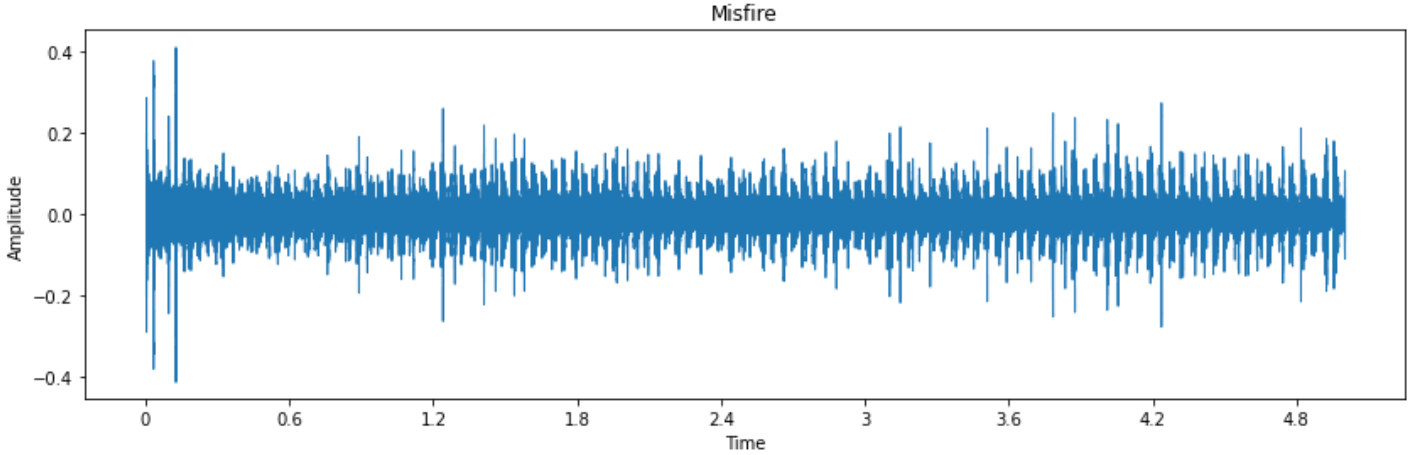
- Abrupt change in between subsequent audio samples ⇒ Noise injection
- More variation in training dataset



Strategy #2

Extract 5-second audio chunks from concatenated signals

- Randomization in audio signals
- Improved test score from 0.63 to 0.72



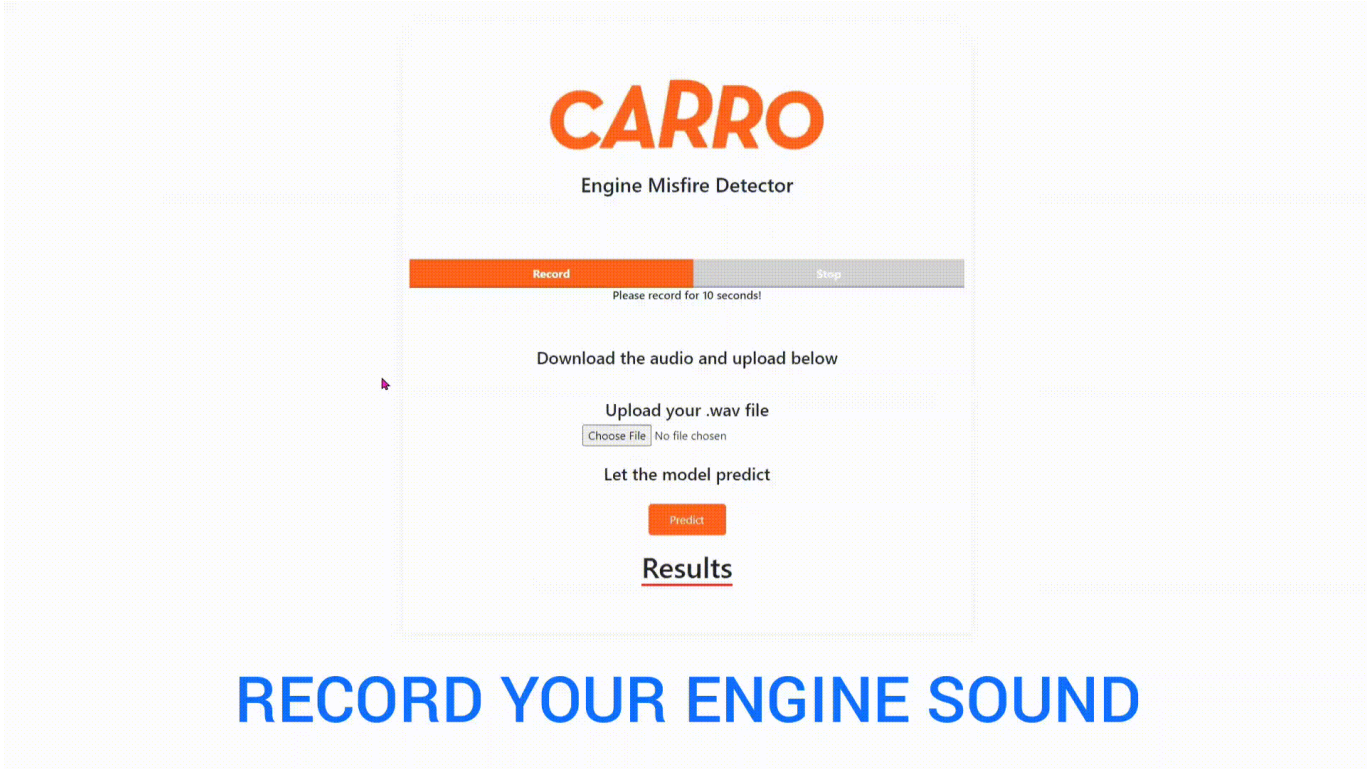
Strategy #3

Divide test audio into 5-second chunks and perform

- Mean-score voting
  - Take mean of all 6 votes
- Threshold voting
  - 2 out of 6 votes ⇒ Misfire

	Model
Method	Final Model
Normal	0.7255
Mean-score voting	0.7647
Threshold voting	0.7451

Deployment



RECORD YOUR ENGINE SOUND

Future Work

Preprocessing/Feature extraction

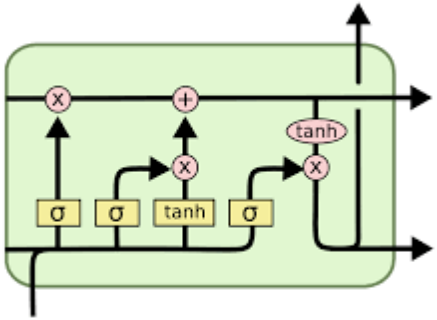
Compare and/or combine different data augmentation techniques

- Image augmentations e.g. reflection of spectrogram
- Audio signal augmentations e.g. pitch shift, addition of random noise

Deep Learning Pipeline

Compare and contrast more advanced models

- Faster R-CNN model
- LSTMs and GRUs



THANK YOU

Q&A Session