Invited Paper of IEEE CISOSE 2023

Athens, Greece, 17th July 2023

A Scenario-Based Functional Testing Approach to Improving DNN Performance

Hong Zhu, Thi Minh Tam Tran, Aduen Benjumea and Andrew Bradley

School of Engineering, Computing and Mathematics
Oxford Brookes University
Oxford OX33 1HX, UK
Email: hzhu@brookes.ac.uk

Outline

1. **Background**
2. **Related Work**
3. **Proposed Methodology**
4. **Case Study**

Scenario-based Testing (SBT)

OXFORD BROOKES UNIVERSITY

- Machine learning (ML), especially deep neural networks (DNNs), has been increasingly employed by critical applications:
  - Security protection
  - Autonomous vehicles
  - Communication and cloud/edge computing,
  - Medical research, etc.
- Necessary to perform extensive testing covering the complete variety of possible ***scenarios*** that the system may encounter
- Scenario-based testing (SBT):
  - To split test cases into a number of subsets so that each subset tests the system on one possible operation scenario
  - To assess and evaluate the performance of the system on one scenario at a time
- In traditional software engineering
  - Widely used
  - Proven to be efficient and effective
  - Required by standards for safety critical systems (e.g. ISO 26262)

SBT for ML Applications

- Current state of art:
  - Active in the research on testing autonomous vehicles
  - Limited to verification and validation testing of functional safety

  } It is required by ISO 26262

  - Focused on test cased generation: e.g. on traffic conditions, road layout, etc.
  - Effectiveness and efficiency: unknown
  - Very limited in testing other ML applications
- The challenges:
  - Traditional software: Test –> Detect Bugs –> Debug –> Quality Improved
  - ML models (DNN):
    - Models cannot be “debugged” via manual editing the model
    - Model retraining with error-inducing data
      - In sufficient number of error-inducing data
      - Error-inducing data may not indicate real weakness of the model
      - Improving performance on one scenario cause performance decrease on others

Proposed Solution

OXFORD BROOKES UNIVERSITY

- Methodology level
  - A process based on the principle of exploratory testing methodology
  - Achieving test efficiency through datamorphic testing methodology
- Technical level
  - Employment of transferred learning / continuous learning techniques to improve ML model performance
  - Uses of data augmentation techniques to improve test efficiency and effectiveness

Related Work 1: Methods and Processes

OXFORD BROOKES UNIVERSITY

- Scenario-based testing methods and processes
  - Use-cased driven OO methodology
    - Scenario is a linear sequence of interactions with the SUT
    - Scenarios are modelled in use case diagram, activity diagrams, scenario descriptions (e.g. user stories), behavioral descriptions, etc.
    - Test cases are derived from scenario descriptions/models
  - Testing autonomous vehicles
    - ISO 26262:
      - Scenario plays the central role of testing functional safety
      - Waterfall process model
      - *Earlier versions*: based on traditional software engineering methodology
      - *Current version*: inherited the conceptual model of its earlier versions
    - Menzel, Bagschik and Maurer [2018]:
      - *Functional scenario*: natural language description by domain experts
      - *Logic scenario*: formally specified by a set of state variables and their value ranges
      - *Concrete scenario*: each state variable is assigned a specific value -> test cases
    - Much work reported on deriving scenarios for testing AV (See, e.g. [Ding 2023] for a survey)
    - Zhu et al. [2018, 2019]: Datamorphic testing: a scenario-based approach to develop test systems

Related Work 2: Key Techniques

OXFORD BROOKES UNIVERSITY

- Test oracle problem
  - Metamorphic testing
  - Datamorphic approach (mutational metamorphic relations [Zhu 2015, Zhu 2018])
- Test case generation
  - Data augmentations to generate synthetic test data from real data, e.g.
    - Tian et al.: DeepTest [2018], image editing + Fog, rain effect, used photoshop
    - Zhang et al. [2018], DeepRoad, used GAN for testing AV
    - Zhu et al. [2018, 2019], used AttGAN to generate test cases for face recognition
    - Hasirlioglu and Riener [2020], developed a digital augmentation algorithm for rain effect
    - Musat et al. [2021], simulated the combinations of multiple weather conditions
- Test adequacy and coverage
  - Scenario coverage:
    - Coverage of all scenarios (all functional safety requirements) widely used in testing AV
    - Zhu et al. [2018, 2019] : 1'st order mutant coverage, high order mutant coverage, mutant combination coverage, etc.: different levels of combination of operation conditions.
  - Neural network structural coverage:
    - Pei et al. [2017] Neuron coverage; Ma et al. [2018] Neuron output coverage; Sun et al. [2019] Neuron decision coverage; Xie et al. [2022] Neuron path coverage, etc.

Proposed Methodology

OXFORD BROOKES UNIVERSITY

The proposed testing process is an iterative cycle of the following

- *Assumption*:

  Identifying scenarios in which system's performance needs improvement

- *Diagnosis*:

  Confirming the assumptions via further testing ML model on the suspected scenarios
  - Analysing ML model's performance statistically on suspected scenarios
  - Confirming the assumption based on test results

- *Treatment*:

  Retraining the ML model with additional training data that target the diagnosed weak scenarios

- *Evaluation*:
  - Testing the treated ML model after retraining
  - Evaluating the effect of the treatment

Key Technical Problems

OXFORD BROOKES UNIVERSITY

- How to treat a ML model on a specific weak scenario without causing side-effects (e.g. performance decrease on other scenarios)
  - *Transfer learning*:
    To apply a transfer learning technique, i.e., to use the existing model as the base for retraining
  - *Targeted training*:
    To use a set of training data that represent the scenarios to be treated
  - *Prevention of side-effect*:
    To include a subset of the original training data in the retraining dataset to prevent the so-called *forget* effect
- How to conduct test efficiently (especially the problem of lack of data)?
  - Uses of data augmentations to generate synthetic data for both training and testing

Case Study: Autonomous Racing Car

OXFORD BROOKES UNIVERSITY

- Formula SAE Competition
  - Designing and building a fully autonomous vehicle from the ground up
    - a base vehicle provided by the competition
    - a custom array of sensors
    - software to control the vehicle
  - The driving tasks:
    - straight-line acceleration
    - a single lap of a track
    - figure-of-eight
    - a 10-lap timed event

The course is marked out by coloured traffic cones demarcating the circuit boundaries

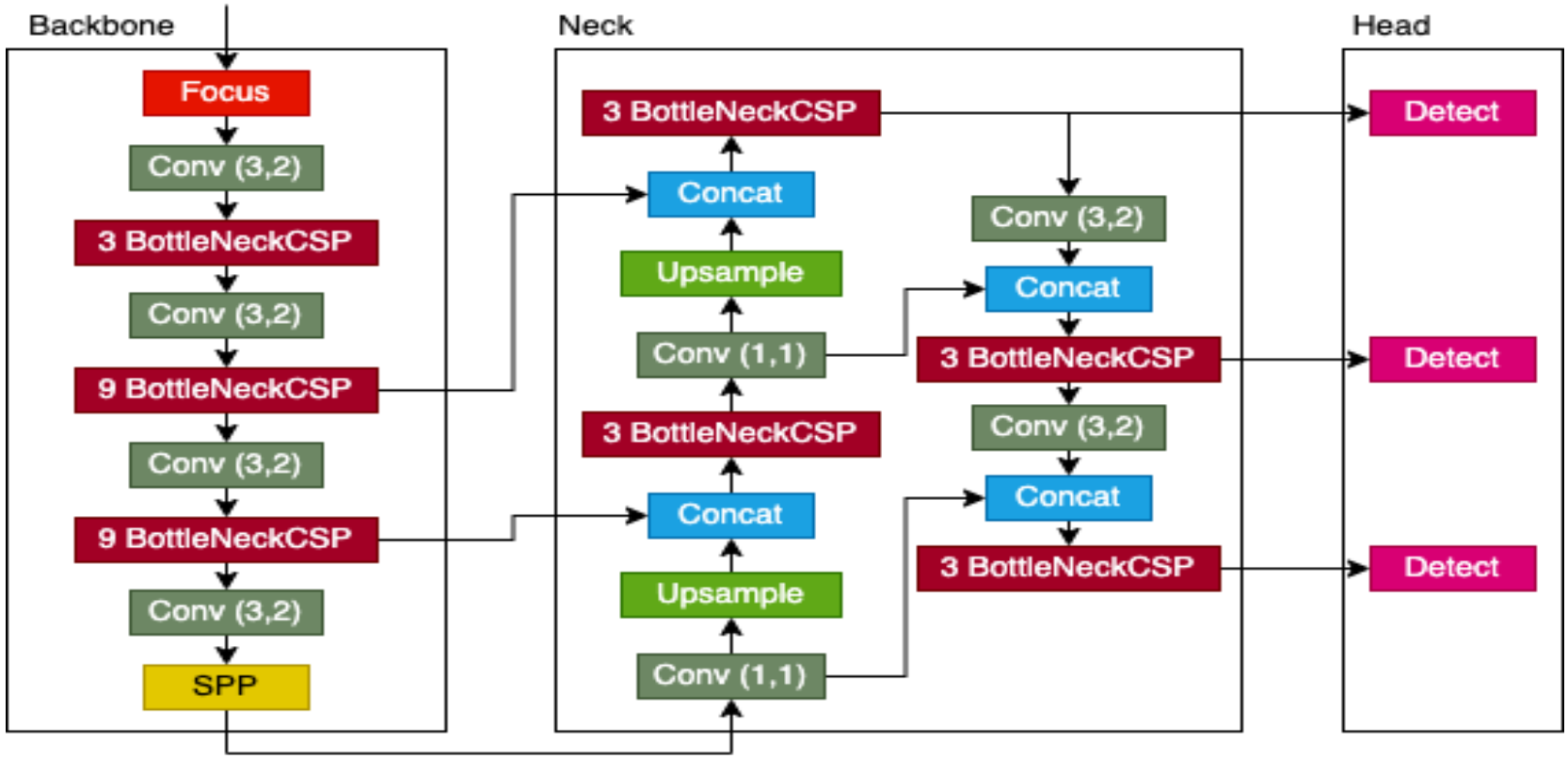
- yellow and blue cones on left and right, respectively
- orange cones to dictate other features (*e.g.*, start/finish lines, stopping areas etc).

- The perception system
  - **Function**: detect and identify the location and colour of each cone
  - **Consequences of error**:
    - a slow lap
    - hitting a cone: a time penalty, and the misplaced or knocked-over cone makes a subsequent lap more challenging
    - the vehicle veering off course, failure to complete the event

The Perception System

OXFORD BROOKES UNIVERSITY

- A custom trained YOLOv5 instance of “S” scale (training dataset: 644 images)
- Developed by OBR Autonomous Team



Testing Model M_0

OXFORD BROOKES UNIVERSITY

- ***Test dataset***: 100 Images from a setup track on the campus with some dark images, objects in shadow, sun beam, etc.
- ***Performance***:
  - Precision: 93.01%
  - Recall: 92.12%
  - mAP@50: 90.13%

- To further improve the performance, the errors made by the model were manually inspected.
- The problem is identified that many of the errors were in the situation when the picture was taken in a poor lighting condition and weather conditions.

Definitions of the Performance Metrics

Metric	Definition
Precision	The percentage of recognised cones that are correct with IoU = 0.5
Recall	The percentage of cones that are recognised with IoU = 0.5
mAP @50	The average precisions over all images, which the area under the precision-recall curve with IoU = 0.5

Suspected Weak Scenarios

OXFORD BROOKES UNIVERSITY

- *Bright*. The input image to the perception system is in a very bright lighting condition.
- *Dark*. The input image is in dark lighting condition.
- *Flare*. The input image is of flaring lights.
- *Rain*. The image is in a raining weather condition.
- *Fog*. The image is in a fog weather condition.
- *Water*. The image is in the situation when water is splashed on the camera lens.
- *Speed*. The input image is when the vehicle is moving fast.

These operation conditions were insufficiently represented in the training and testing datasets due to difficulty in the collection of such data and the expense in labelling.

Generation of Test Datasets

OXFORD BROOKES UNIVERSITY

- Seven datamorphisms are developed
  - by using the data augmentation library of the open-source projects Automold;
  - one for each scenario
- Each datamorphism is applied to 100 images in the test dataset

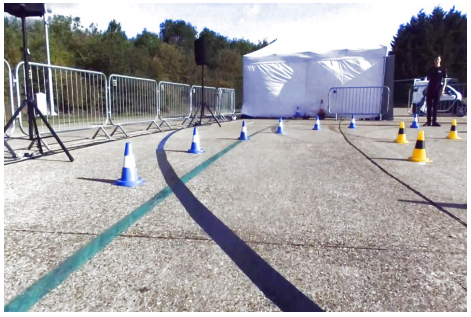
Datamorphisms for Testing M_0

Name	Specification	Implementation
Bright	Change the brightness of the image upwards	Set the bright coefficient = 0.9
Dark	Change the brightness of the image downwards	Set the darkness coefficient = 0.4
Flare	Add flare areas to the image	Add a flare layer
Fog	Add fog effect to the image	Set the fog coefficient = 0.4
Rain	Add rain effect to the image	Add raining effect
Speed	Blur the image as if shot when camera is moving	Add a speedy effect
Water	Add water splash effect to the image	Add blurry effect and a layer of water drops

Effects of Data Augmentations

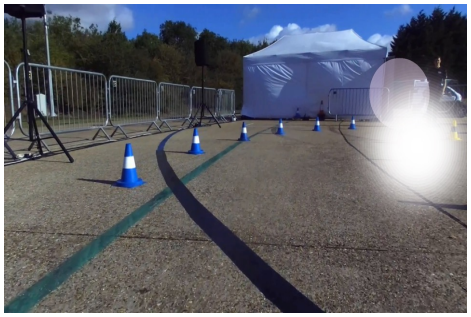
OXFORD BROOKES UNIVERSITY

(1) Original

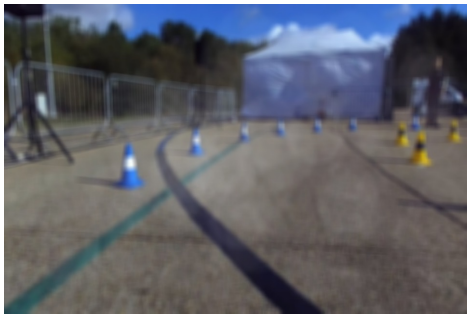


(2) Bright

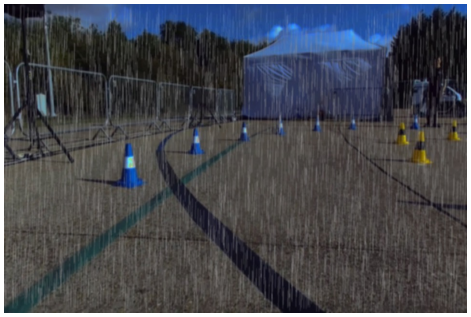
(3) Dark



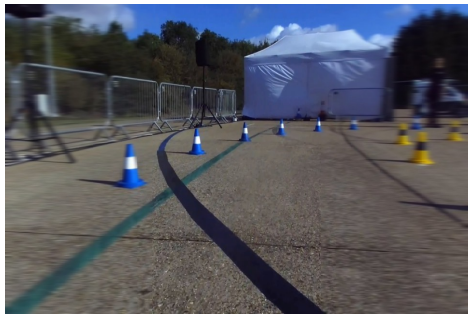
(4) Flare



(5) Fog



(6) Rain

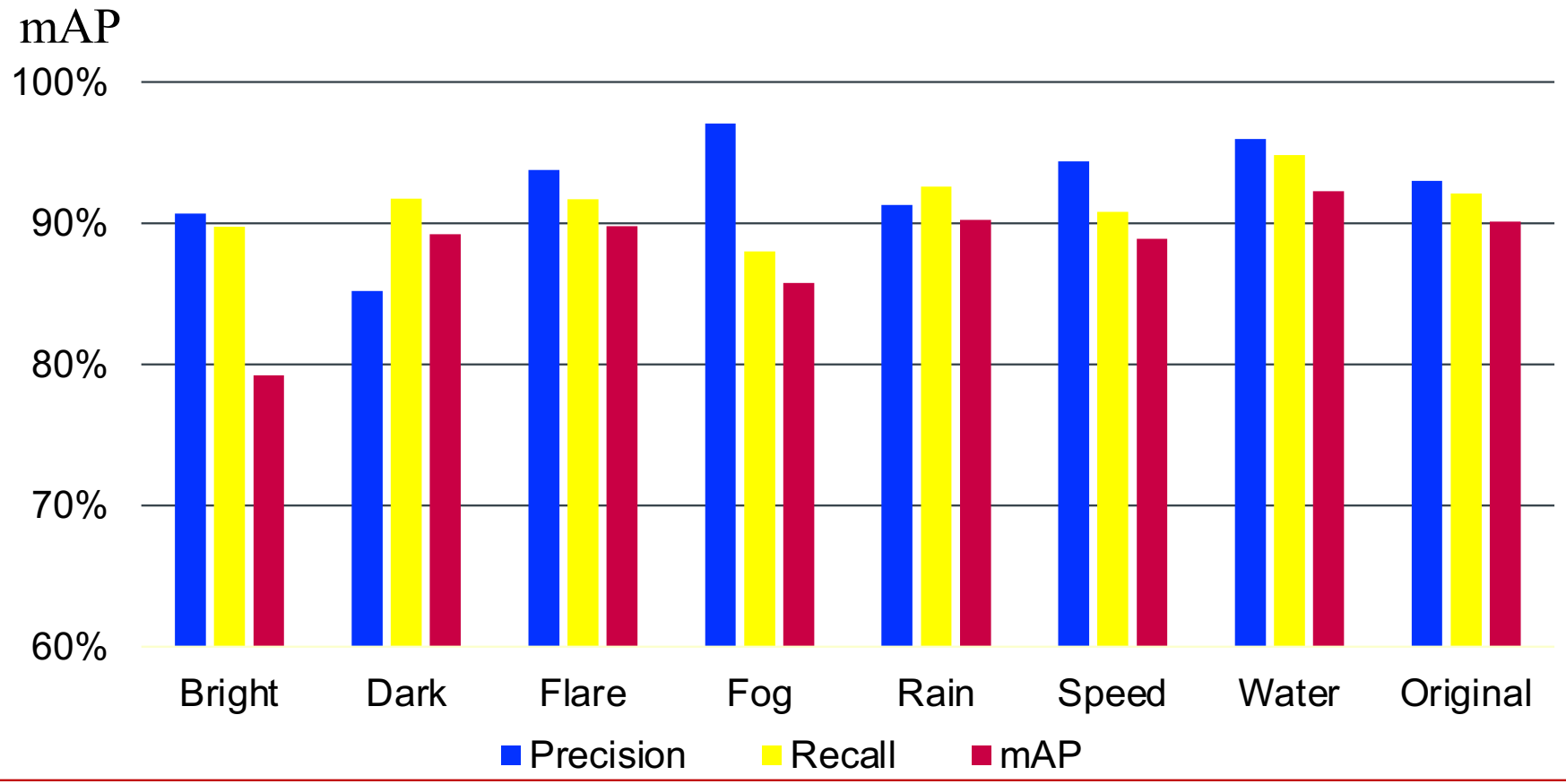


(7) Speed

(8) Water

Evaluation of M_0 on Scenarios

OXFORD BROOKES UNIVERSITY

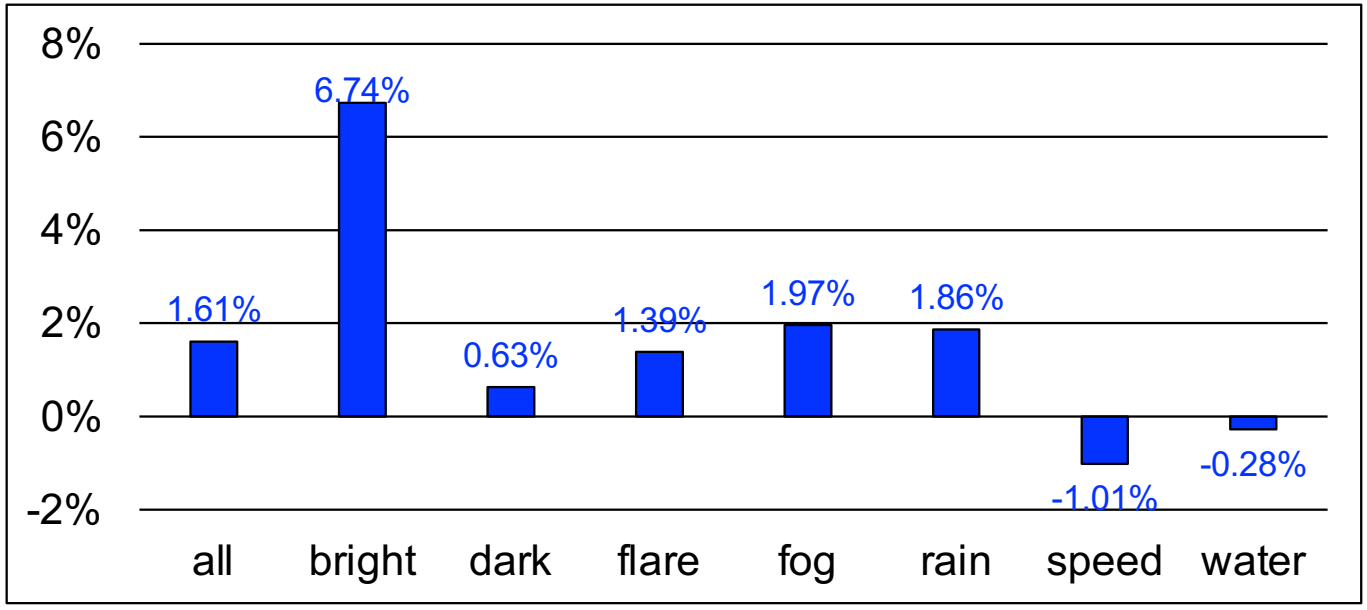


The test results confirmed the suspection that the model M_0 performed less satisfactory on these scenarios

Improving Model M_0

OXFORD BROOKES UNIVERSITY

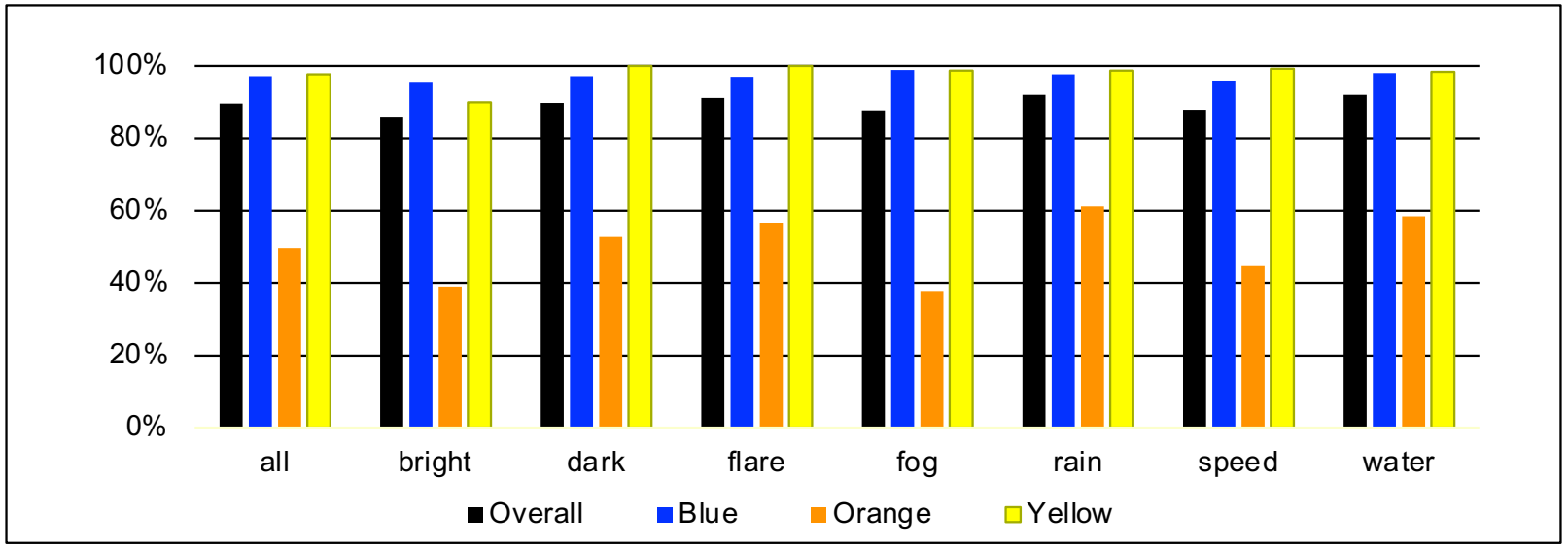
- Use M_0 as the base model
- Training dataset:
  - *Synthetic data*: Apply each datamorphism to 10% original training data selected at random
  - *Original training data*: 10% selected at random
- Result (M_1) improved on overall performance by 1.61 points on mAP



Testing Model M_1

OXFORD BROOKES UNIVERSITY

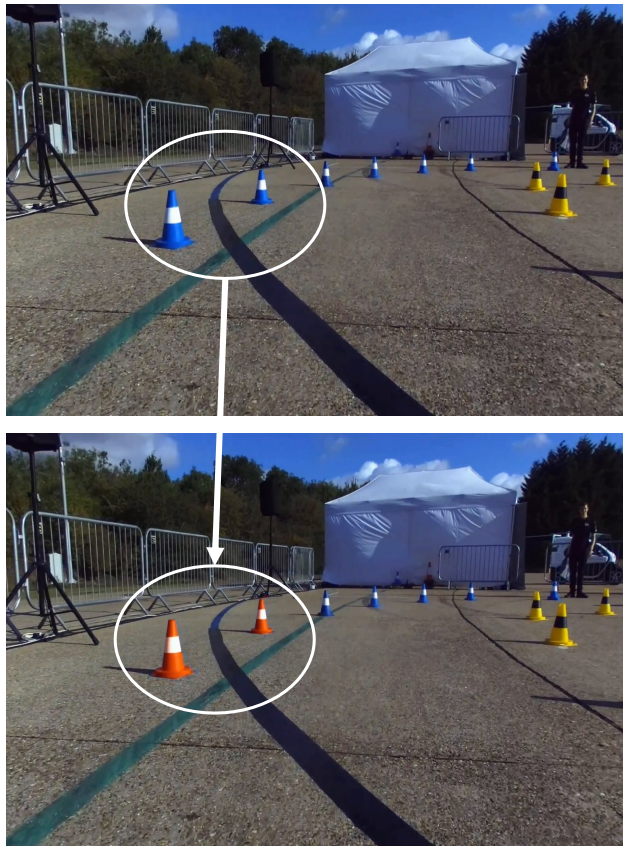
- The performance of model M_1 slightly decreased on the scenarios of *Speed* and *Water*.
- The test cases on which the model fails were inspected and problem identified
  - The models did not detect ***orange-coloured cones*** so well as other coloured cones
- The model is statistically analysed of its performance on detecting different coloured cones



Improving Model M_1

OXFORD BROOKES UNIVERSITY

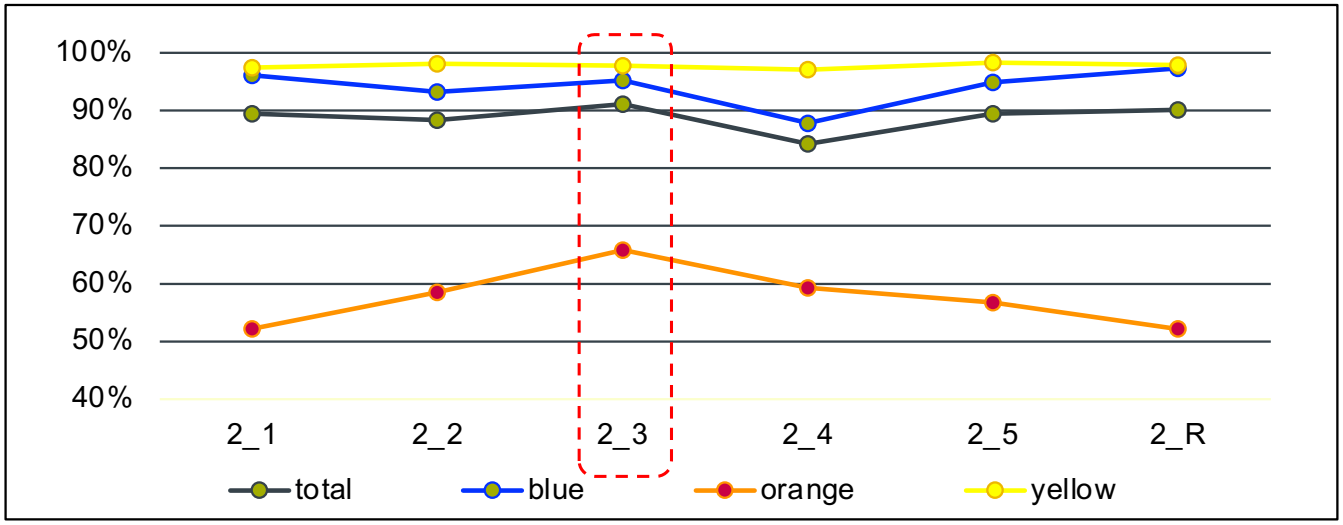
- Datamorphism *OrangeCone*
  - ***Function***: to transform blue cones in images into orange cones
  - ***Implementation***: Python code to manipulate images
- Models $M_{2.x}$
  - Base: M_1
  - Training data:
    - *Original training data*: 10% selected at random
    - *Synthetic data*: applying datamorphism *orangeCone* to 10%, 20%, …, 50% of original training data selected at random
- Model $M_{2.R}$
  - Base: M_1
  - Training data:
    - 145 images of real pictures of that contain orange cones
    - 10% of the original training data selected at random



Evaluation of Models $M_{2.x}$

OXFORD BROOKES UNIVERSITY

- The best model: $M_{2.3}$ (= M_3)



- The performance of detecting orange cones improved from 50% to 65.84% (by > 30%) on mAP.
- The performance of detecting other coloured cones remain the same.

Observations

OXFORD BROOKES UNIVERSITY

- The performance of DNN models can be improved effectively through SBT following the iterative process proposed in this paper
  - There is still space for M_3 to improve
  - ML model are inevitably imperfect
  - Is there a limit for performance improvement?
- Synthetic data generated by applying augmentation to train machine learning models can achieve a performance as good as natural data
  - Developing and applying datamorphisms are cost efficient to generate both training and testing data
  - Data augmentations that are not 100% preserving semantics can still be cost-efficient
- More training data does not necessarily imply better performance
  - Model $M_{2.3}$ is better than $M_{2.4}$ and $M_{2.5}$
- Transfer learning techniques (esp. rehearsal-based methods used in the case study) are useful to prevent "*catastrophic forgetting*"

Conclusion: Summary

OXFORD BROOKES UNIVERSITY

- Iterative process: ***Symptoms –> Diagnosis –> Treatment –> Evaluation***

- Scenario-based testing
  - A subset of test cases to represent one operation scenario
  - Testing on different scenarios separately and compare with each other statistically
    - *In diagnosis stage*: To confirm in which scenario the model is weak in performance
    - *In evaluation stage*: To check if performances are improved on treated scenario and not decreased on other scenarios
- Use of synthetic data
  - To generate scenario-based test datasets
  - To generate training datasets targeting weak scenarios
- Improving performance: Treatment on weak scenarios
  - Transfer learning with rehearsal

    New training data on the treated scenario + a subset of original data selected at random

Conclusion: Future Work

OXFORD BROOKES UNIVERSITY

- Further development of Morphy to provided more support to test automation
  - The case study is conducted with the Morphy testing tool
- Further case study to improve the performance of the perception system and other components of the autonomous racing car
  - Control component, path planning component, etc.
- How to improve ML model's performance based on test results is a big research question for software testing community
  - Transfer learning / continuous learning techniques seem promising
  - How to prevent *catastrophic forgetting* is the key

Thank You

Acknowledgement

The work reported in this paper is partly funded by the Oxford Brookes University's 2020 Research Excellence Award. The authors are grateful to the OBR Autonomous team for their engagement in the project and providing support in the case study.