

Patch Perfect: Using AI to Fix Roads

In partnership with Motus and Standard Bank CIB Digital, we are thrilled to bring you this hackathon in the important landscape of road safety!

South Africa has the 11th largest road network in the world and the largest in Africa. Our roads are critical to the movement of people and goods across the country.

Potholes have long been an issue on many road segments across the country from residential areas to municipal and national roads. These potholes are safety hazards which can lead to accidents, traffic, punctures, and car damage. As such, they have a big impact on our lives.

It is important to fill potholes promptly and effectively. Potholes vary in size and must be filled with asphalt to be long-lasting. Additionally, the pothole to be filled should be dry, reasonably sized, and must not be surrounded by significant road damage.

Motus has been running an initiative called Patch Mzansi to fill potholes and hope to scale their operations. However, as with any venture, one must be able to understand their costs and expected material requirements. Therefore, having an accurate picture of how much asphalt is required to fill potholes is extremely important. Patch Mzansi have been collecting images of every pothole filled and how many bags of asphalt were required to fill them. We are interested in knowing how much asphalt is required to fill potholes purely based on an image of that pothole. This will help with resource allocation and ensure Patch Mzansi can sustainably fill potholes and improve road safety.

Welcome to the Challenge!

The hackathon aims to explore the vast potential of Machine Learning and Computer Vision to fill potholes quickly, correctly, and cost-effectively.

Challenge Question:

Can you predict the amount of asphalt required to fill a pothole given just an image of that pothole?

Bonus question (Not-compulsory):

Now that you have built a model to predict how much asphalt is required to fill a pothole based on just images, how might you commercialise this model? How can your model be used practically in industry?

Data

The dataset can be found at the following resource group: [2024 Hackathon Resources](#)

The dataset comprises of annotated images of potholes, where annotations include bounding boxes for the pothole itself and two classes representing measurements from a stick (L1 and L2). Additionally, a CSV file accompanies the images, containing unique identifiers for each pothole and the number of bags used to fill them. This combined dataset enables participants to develop predictive models to estimate the amount of asphalt required for filling potholes. We encourage the use of additional data sources to aid in your solution, but this is not required. The image annotations are stored in .txt files and follow the YOLO format which is as follows:

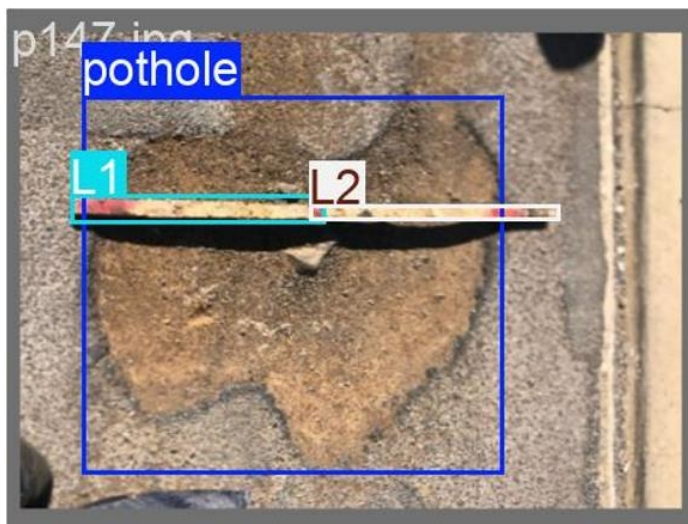
```
<object-class> <x> <y> <width> <height>
```

Where object-class is the label of a rectangular bounding box, x and y are the co-ordinates of the centroid of the bounding box, width is the total width of the box, and height is the total height of the bounding box. There are three classes present within the dataset. These include the following:

- 0 – Bounding box outlining the pothole itself.
- 1 – Bounding box representing L1 (one side of a measurement stick). **The true length for L1 is 500mm.**
- 2 – Bounding box representing L2 (the opposite side of the measurement stick).

Example:

p147.jpg



p147.txt

```
<object-class> <x> <y> <width> <height>
0 0.412894 0.530017 0.637298 0.792453
1 0.268949 0.371355 0.383855 0.060034
2 0.626967 0.377358 0.376473 0.034305
```

train.csv

pothole_id	bags_used
147	1

Files:

- **train_images.zip** – contains pothole .jpeg files where each file is named using the pothole_id to be used for training
- **test_images.zip** – contains pothole .jpeg files where each file is named using the pothole_id to be used for testing and submissions to the Kaggle leaderboard.
- **train_annotations.zip** – contains .txt files storing bounding box classes and co-ordinates in the YOLO format for training. For each image in train_images.zip there is a corresponding .txt file in train_annotations.zip.
- **train_labels.csv** – contains the pothole_id and corresponding number of bags for each pothole in the training set
- **test_labels.csv** – contains the pothole_id for each pothole in the test set. This dataset purposefully does not contain the bags used for filling as these should be predicted and submitted to the leaderboard.

Hints, Tips, and Tricks

Explore pre-trained models and utilize pixel-to-millimetre conversions to enhance accuracy when predicting asphalt bags required for pothole filling. Experiment with different machine learning approaches and leverage community resources for additional support.

1. Consider leveraging Google Colab or Kaggle notebooks for GPU resources, enabling faster and more robust model training.
2. Utilize Pre-trained Models: If you're new to computer vision tasks or struggling with detecting bounding boxes for different classes (like potholes, L1, L2), consider leveraging pre-trained models like YOLOv8. These models are well-suited for object detection tasks and can be adapted to your specific classes with fine-tuning. There are plenty of available tutorials online to aid in the object detection task. Other pre-trained computer vision models can be easily found on HuggingFace.
3. Implement Pixel-to-Millimetre Conversion: Since you have annotations with measurements from a stick (L1, L2), utilize this known length to calculate the pixels per millimetre ratio in your images. This conversion factor will help you accurately measure the size of potholes in millimetres, which is crucial for predicting the amount of asphalt needed.
4. Feature Engineering: Apart from pothole size, explore other potentially predictive features such as the shape (aspect ratio), depth (if discernible from images), and other features. Incorporating these features can enhance the accuracy of your predictive models.
5. Model Selection and Evaluation: Experiment with different machine learning models such as regression, random forests, or neural networks to predict the number of asphalt bags required. Evaluate your models using appropriate metrics (e.g., Mean Squared Error, R-squared) to ensure robust performance and generalize well to new data.
6. Visualization and Interpretation: Visualize your results to understand how well your model is predicting asphalt bag requirements. Interpret the importance of different features and assess if additional data or features could further improve predictions.
7. Community and Resources: Don't hesitate to seek help from the community or refer to tutorials and forums online. Many resources are available to assist with specific challenges in image annotation, model training, and predictive modelling.

By combining these strategies, you can streamline your approach to effectively predict the number of asphalt bags required to fill potholes, leveraging both the image data with annotations and the historical filling data provided in the CSV file.

Team details

Teams can consist of 1 - 5 individuals. Approximately 5 teams will be shortlisted and then invited to present to a judging panel on Thursday 22 August, 17:30 at an in-person event at the Stellenbosch University Museum. We fully encourage women in data science and there will be additional prizes for eligible teams.

Prizes:

Prizes will be award to the top three teams selected at the finals. The top-placed women's team will be eligible for an additional prize. The prizes are as follows:

1st Place Team – R22 000.00

2nd Place Team – R12 000.00

3rd Place Team – R6 000.00

Prize for top performing team consisting of at least 80% women: R6 000.00 (this can be received in addition to first, second or third prizes).

Submission and Judging:

The Hackathon will launch virtually on Thursday 8 August (17:30). **Submissions will be accepted from Thursday 8 August until Sunday 18 August (23:59).**

Your submission is broken into two parts:

1. Submit your predictions on the test set to the Kaggle Competition Leaderboard.
 - a. Please ensure your submissions on the leaderboard are show with your team's name being the same name chosen upon registration.
 - b. Please provide predictions for the potholes in test_images.zip and test_labels.csv
 - c. Kaggle Leaderboard Link:
<https://www.kaggle.com/t/434b2e475d3c4e88911c7518856ca844>
2. Submit your methodology and code to the supplied link:
 - A pdf document of no longer than one page explaining your solution.
 - A PowerPoint presentation describing the problem, your approach, and your solution.Finalists will be given an opportunity to polish their slides before the final presentations, without adding any content.

- A link to code. It can be a GitHub repository, a google drive link, a Kaggle Notebook, or a Google Colab Notebook. Make sure that permissions are set so that we can download your code.
- All submitted documents must include your Team Name and Team Members
- **Please note this should match up with your final submission on the Kaggle Leaderboard.** We will verify that Kaggle leaderboard predictions line-up with your submitted code and methodology.
- Submissions Link: <https://forms.office.com/r/ymq63AxcZ1>

Make use of the submission form to submit your solution. The form will close on **Sunday 18 August at 23:59**.

There will be two rounds of judging. In Round 1, solutions will be scored and approximately 5 finalists will be selected to present their solutions at an in-person event on **Thursday 22 August**. Finalists will be emailed on Tuesday 20 August; however, all participants are encouraged to attend the in-person event.



Evaluation of solutions

Judges:

Round 1 Judges: Dr. Sunday Oladejo, Dr. Gray Manicom, Murali Veeraragoo, Miren Bitcham, John Mukomberanwa

Finals Judges: Sunday Oladejo (Stellenbosch University), Ofentse Motaung (Motus), John Mukomberanwa (Standard Bank), Heidi Taylor (Standard Bank), Musa Malwandla (Differential Capital)

Hackathon Finals Master of Ceremonies: Dr. Itai Makone (Stellenbosch University)

Evaluation Criteria:

Participants will be evaluated based on two primary scores to determine their final standing in the hackathon:

1. **Leaderboard Rank (20%):** The Kaggle leaderboard rank reflects the performance of participants' predictive models compared to others in the competition. It accounts for 20% of the final score and measures the accuracy and effectiveness of their predictions in estimating the number of asphalt bags required to fill potholes.
2. **Methodology, Approach, and Code (80%):** The remaining 80% of the final score is based on the methodology, approach, and code used by participants to develop their models. This is described further in the below rubric.

Round 1 Judging Rubric:

Category	Description
Leaderboard (20)	
Rank based on R^2 (20)	Score will be assigned based on the relative position on the Kaggle Leaderboard.
Methodology and Presentation (80)	
Problem interpretation (8)	Is the problem interpreted in a suitable manner? Is data used to drive modelling decisions?
Model design (15)	Are model design decisions well motivated?
Evaluation protocol (13)	Are metrics and evaluation procedures appropriately chosen?
Creativity and insights (10)	How creative is the solution approach? What ideas are presented as an extension to the base problem?
Interpretable Results (10)	Are the results easily interpretable and understandable for business users.
Relevance of solution to a context (12)	What impact could the solution have?
Presentation and pitch (12)	Clarity and structure of the presentation.