HIGH VOLUME TRANSPORT APPLIED RESEARCH

a company of Royal HaskoningDHV

Computer Vision for Transport Data Collection

Development of software for low-cost transport data collection

September 2024

Project 103S: Computer Vision for Public Transport

T-TRIID: HVT055-SE043

DT Global

UK International Development

Partnership | Progress | Prosperity

This research was funded by UKAID through the UK Foreign, Commonwealth & Development Office under the High Volume Transport Applied Research Programme, managed by DT Global.

The views expressed in this report do not necessarily reflect the UK government's official policies.

Reference No.	HVT055-SE043
Lead Organisation/ Consultant	Integrated Transport Planning
Partner Organisation(s)/ Consultant(s)	N/A
Title	Computer Vision for Public Transport
Type of document	Project Report
Theme	Urban transport
Sub-theme	Low carbon transport, Policy Regulation, Technology and Innovation
Author(s)	Giles Lipscombe, Dr. Mark Dimond
Lead contact	Giles Lipscombe
Geographical Location(s)	Sierra Leone, Mozambique, Tajikistan

Abstract
The availability of transport data in low and middle-income countries (LMICs) is often poor and the budgets available for development projects are often unable to support significant primary data collection. We present a computer-vision-based data collection tool making use of dashcam video or crowdsourced street-level imagery to help fill these data gaps at a low cost. The tool outputs high-level data: the relative prevalence of different transport modes across each part of the study area, which is intended for use in work such as the development of city-wide transport strategies or Sustainable Urban Mobility Plans (SUMPs). During development we tested the tool in the cities of Bo (Sierra Leone), Dushanbe (Tajikistan) and Maputo (Mozambique).

Keywords	Computer vision, Data collection, Crowdsourcing, Street-level imagery, Urban transport
Funding	T-TRIID
Acknowledgements	The project team gratefully acknowledge the support of UKAid from the British people, through T-TRIID, part of the High Volume Transport Programme.

Issue	Status	Author(s)	Reviewed By	Approved By	Issue Date
1	Draft	GL	MD	MD	01/08/2023
2	Draft	GL	MD	MD	29/09/2023
3	Draft	GL	MD	MD	30/11/2023
4	Draft	GL	MD	MD	15/12/2023
5	Final	GL	MD	MD	26/06/2024

Contents

Appendices

Figures

Abbreviations/Acronyms

CPU	Central Processing Unit (of a computer)
CV	Computer Vision
FCDO	Foreign, Commonwealth & Development Office
GPU	Graphics Processing Unit (of a computer)
HIC	High Income Country
HVT	High Volume Transport
LMIC	Lower Middle-Income Country
SUMP	Sustainable Urban Mobility Plan
UK	United Kingdom of Great Britain and Northern Ireland
UMIC	Upper Middle-Income Country
WP	Work Package
YOLO	“You Only Look Once”, an object detection algorithm published by Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi in 2015

Executive Summary

Background

Cities in low and middle-income countries (LMICs) often have inadequate transport systems which fail to enable safe, affordable and efficient travel. At the same time it is common for local transport authorities to lack the resources and capacity to plan and regulate the sector. The resulting transport system often does not serve travellers (poor safety standards, predatory fares, fill-and-go operation), transport workers (low pay, long hours, job insecurity), or the city (congestion, pollution)[1].

Good availability of relevant information contributes to the planning and implementation of successful development programmes, but it is common that transport authorities lack the resources required for this level of data collection. This presents a vicious cycle in which a data deficit makes it difficult to plan interventions and secure additional funding, but without this funding they do not have the capacity to collect the data.

In this report we present a computer-vision-based data collection tool that seeks to address this issue by reducing the cost and complexity of data collection, with a focus on data that can be used in the development of city-wide transport strategies or Sustainable Urban Mobility Plans (SUMPs).

Description of the computer vision data collection tool

The data collection tool takes street-level geotagged images, uses an object detector to count the number of each object type in each image, and then aggregates these counts using a grid over the study area.

The tool is "source agnostic" regarding the input imagery – any source can be used provided the image quality is sufficiently high and each image is accompanied by an accurate pair of coordinates. For lowest costs and greatest ease of use we developed the tool around the use of crowdsourced data from Mapillary.com, which can be accessed through an API, but the drawback of this approach is that the quality and availability of data can be limited in some areas, particularly in LMICs (in comparison to HICs). Alternatively, the tool has been made to also include the ability to process video with a synchronised GPS trace. Although the collection of this data will greatly increase the cost of using the tool, by a few hundred to a few thousand US dollars depending on location, the use of bespoke video allows for control over the time of day, and the days of the week that are surveyed, allowing for higher quality results to be produced.

During development we used the *Grounded SAM* object detector, an 'open set' detector capable of recognising objects it has not specifically been trained to detect. *Grounded SAM* returned good performance when detecting objects such as people, motorcycles and cars, and would enable flexible and complex usage of the results due to its ability to detect other object types. For example, in Bo we may be able to distinguish motorcycle taxis (okada) from private motorcycles carrying passengers by looking for the overlap of two people with one motorcycle and one helmet, based on the observation that okada drivers appear to commonly wear helmets whereas their passengers, and private riders, do not. Grounded SAM unfortunately did not have good performance for three wheelers and heavy trucks, limiting its utility in regions such as Sub-Saharan Africa and South Asia, where these modes are prevalent.

The object detection component of the model has been made to be relatively standalone, and so can be easily modified or replaced to accommodate the rapid pace of development of this technology. The use of a different object detector such as YOLOv8[2] may provide better performance in some areas, such as accuracy for some objects and speed. This may come with some trade-offs, such as the inability to recognise three wheelers, as YOLO is a 'closed set' object detector. This shortcoming could be addressed by training the YOLOv8 model to recognise the required objects, however.

Once the objects in each image have been counted, the tool then aggregates the counts using a grid system. As the input images are typically captured in sequences, with subsequent images taken only a few seconds apart, we found that we needed to ensure we did not double count objects that appear in multiple images. For example, the same car may be shown in many pictures if the camera vehicle is stuck behind it in traffic).

[1] Ajay Kumar, Sam Zimmerman, and Fatima Arroyo-Arroyo. 2021. *"Myths and Realities of 'Informal' Public Transport in Developing Countries: Approaches for Improving the Sector."* https://www.ssatp.org/publication/myths-and-realities-informal-public-transport-developing-countries-approaches-improving

[2] YOLOv8 is available from Ultralytics (https://github.com/ultralytics/ultralytics) as of April 2024, however it may not be included in future releases as subsequent versions are released.

To account for this we take the maximum count seen for each object type, for each sequence, from the images in each grid square, as illustrated in Figure 24. This avoids the issue of counting an object multiple times while also mitigating issues associated with random variability in the quality of each image. For example, if one image in a sequence shows no pedestrians due to sun glare affecting the camera, the pedestrian count assigned to this sequence will not be affected because it uses the maximum count for pedestrians, which will have been taken from an image not affected by the glare.

The drawback of this approach is that we do not present an absolute count of the number of each transport mode in each grid square – instead, we provide the maximum number of objects of each type seen in each grid square. In practice we present these results as being on a scale of “less prevalent” to “more prevalent” to avoid giving the viewer the impression that the numeric values are absolute counts.

Examples of the outputs of the data collection tool can be seen in Figure 1-Figure 3. Although we do not report the average counts as absolute values, the results can still be used to show both the spatial distribution of each transport mode and the relative prevalence of different modes. We primarily intend for this information to be used in projects requiring high-level data with wide coverage, such as transport strategy development, pre-SUMPs or SUMPs. We note that we do not intend for this data to be used to replace traditional traffic counts surveying a single location in detail. While such surveys lend themselves well to the use of computer vision, this would require imagery from static cameras, not the vehicle-mounted moving cameras as presented in this report.

Figure 1: Data collection tool output for “person” object detection in Bo, Sierra Leone

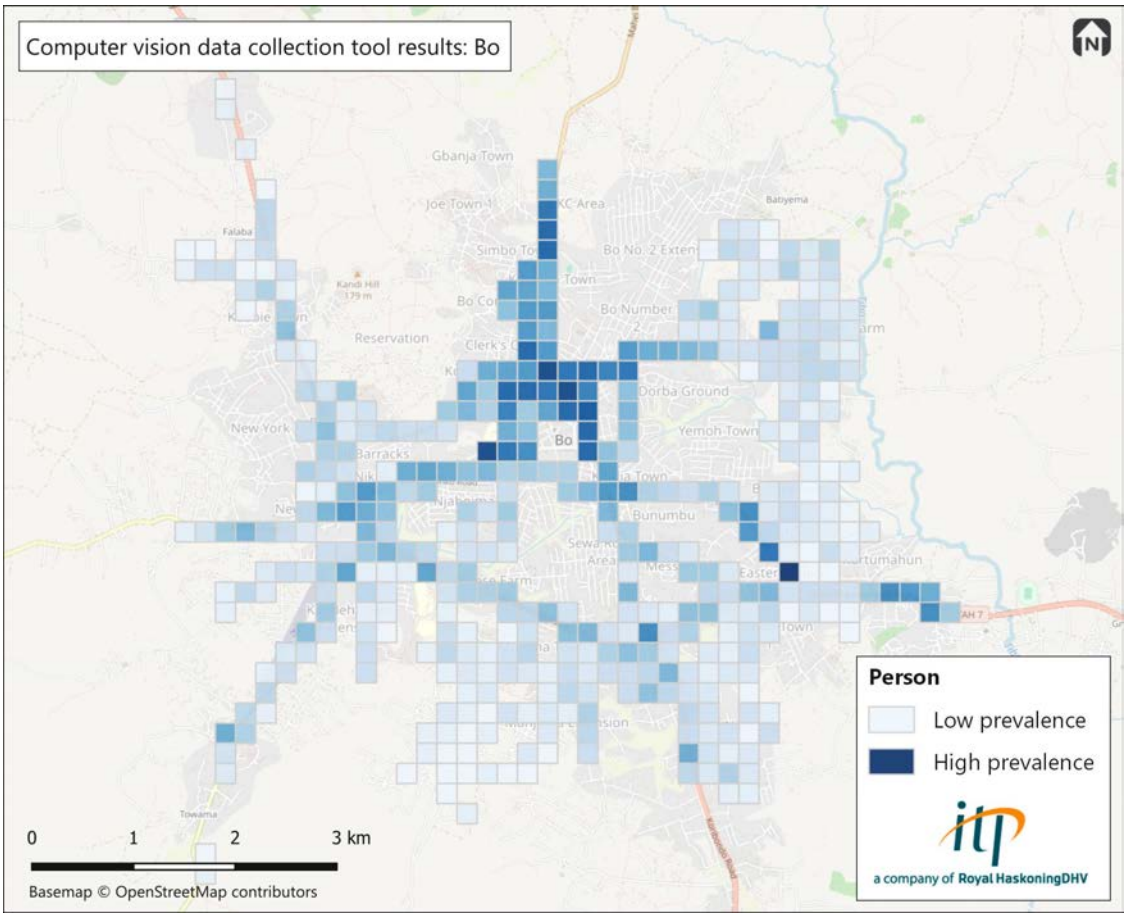


Figure 2: Data collection tool output for “motorcycle” object detection in Bo, Sierra Leone

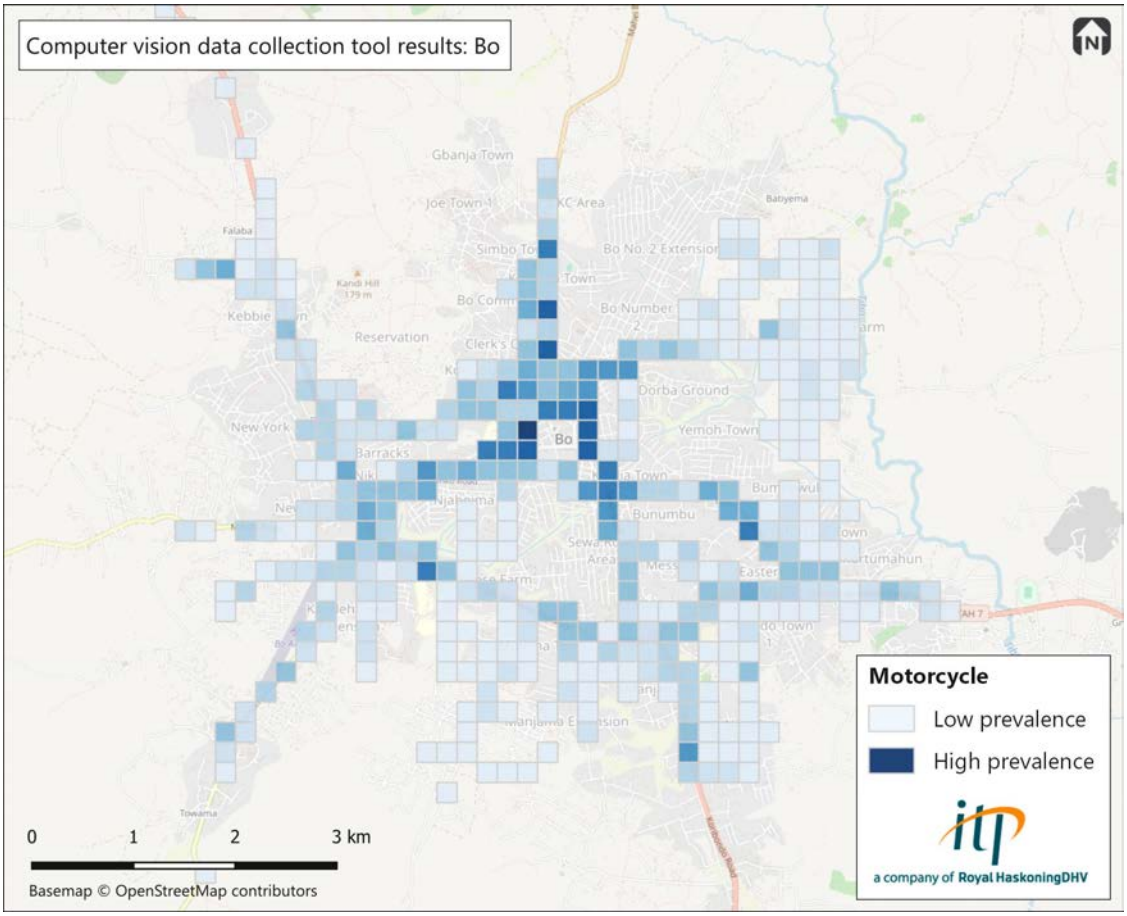
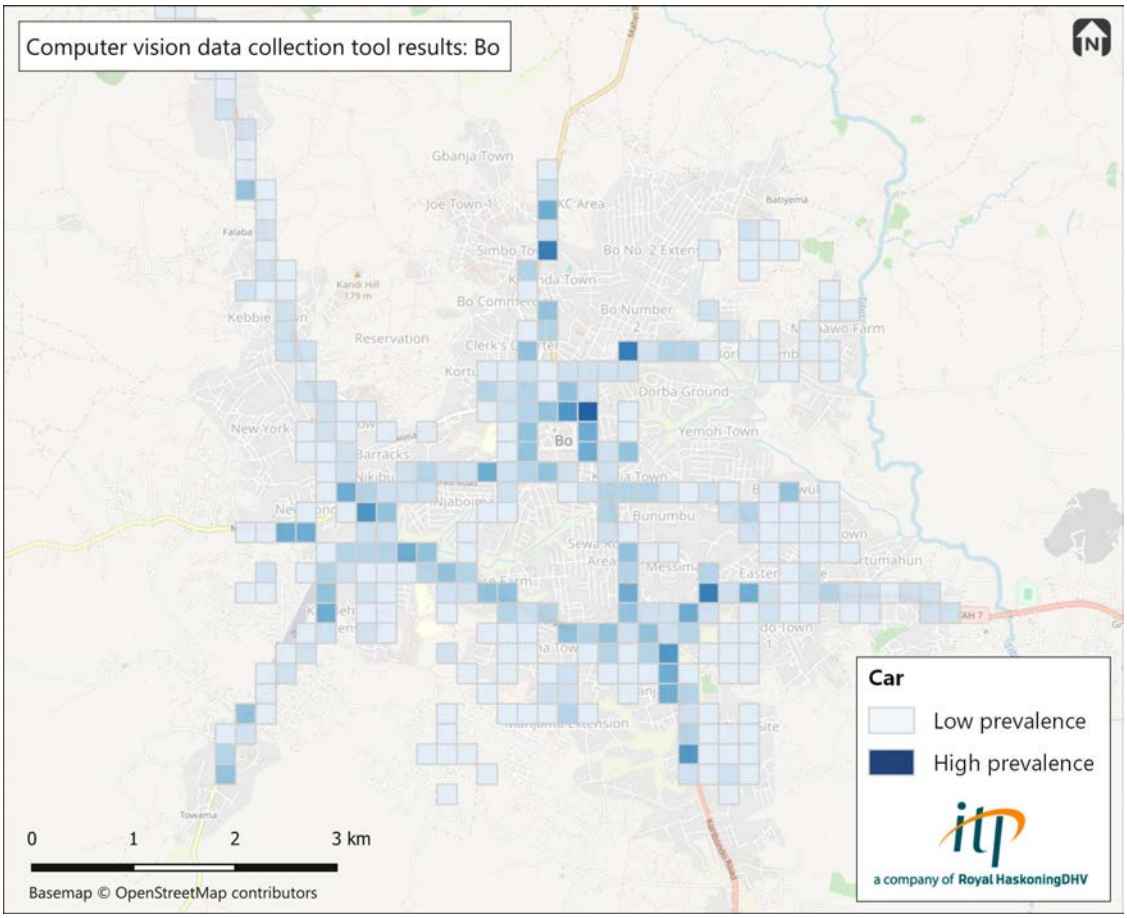


Figure 3: Data collection tool output for “car” object detection in Bo, Sierra Leone



Testing and case studies

We tested the tool in three cities in LMICs: Bo (Sierra Leone), Dushanbe (Tajikistan) and Maputo (Mozambique). Testing in Bo used a combination of crowdsourced data and video collected by bespoke dashcam surveys, as might be encountered when working on a higher-budget project or in a larger city with a relatively well-resourced transport department. Testing in Dushanbe and Maputo both used crowdsourced data only, which we expect to be the more common use case in real-world applications. Each city provided different insights into the availability of input data, performance of the computer vision component of the tool, and potential use cases of its outputs.

In Bo the availability of crowdsourced data initially appeared good but, once time and day filters were applied, only the southern part of the city retained acceptable coverage. “Dashcam surveys”, video recorded with an in-vehicle camera and GPS tracker, were therefore required to top up the input data. This case study allowed us to develop a workflow and practical experience in specifying these surveys and processing the results before their use in the tool.

Although the object detector worked well in Bo for certain transport modes (pedestrians, motorcycles and cars), it struggled with three wheelers due to these vehicles not being included in the most common datasets used to train computer vision models. Addressing this issue by training our own models will therefore be required for effective use of the tool in Sub-Saharan Africa.

The availability of crowdsourced imagery in Dushanbe is good, most likely because use of dashcams in private cars is relatively common, as seen in other post-Soviet countries. As three wheelers are not common in Dushanbe, the object detector worked well for most common transport modes. Identification of shared taxis, a common form of transport in Tajikistan, was not possible, however, which may limit more detailed analyses. We do not see a viable solution to this problem, as even humans have trouble distinguishing shared taxis from private cars - in Dushanbe most do not carry any markings. Despite this, Tajikistan and other LMICs in Central Asia appear to be well suited to deployment of this data collection tool, as the availability of input data is relatively good and detection of common transport modes is acceptable.

Maputo presented an interesting case study because the local transport authority, AMT, has independently collected and uploaded a large number of 360° images to Mapillary. Computer vision would allow for maximum value to be obtained from these, and future surveys at much lower costs than if the process was to be carried out manually. A discussion with a transport professional working in Maputo highlighted an additional possible use case of the data collection tool: the evaluation of intervention schemes by comparing pre- and post-implementation imagery, for example to examine the extent to which paving and widening roads affects the number of different types of user (pedestrians, street traders, vehicles).

Conclusions and future directions for development

In this T-TRIID project we have developed a prototype data collection tool to provide data that allows the characterisation of a city’s transport system, using computer vision technology with street-level imagery. We have demonstrated the tool’s operation in three cities: Bo (Sierra Leone), Dushanbe (Tajikistan) and Maputo (Mozambique), and produced heatmaps of the relative prevalence of people, cars, motorcycles, three wheelers, buses and trucks across each city.

We do not intend for this tool to replace detailed traffic counts, but instead provide higher-level data for the entire study area, suitable for transport consultancy projects such as transport strategy development. Such projects often have relatively small budgets with little capacity for city-wide data collection, so a key component for successful deployment is low cost of operation. In its current form the tool can produce outputs with only a few hours of active staff time and, though using crowdsource data, no monetary expenses.

In real-world deployment, however, a more effective solution may be to use free crowdsourced data as much as possible, with targeted dashcam surveys used to ‘top up’ any spatial or temporal gaps. Although this would incur larger costs than the all-crowdsourced use case, it should still prove cheaper whole-city data than traditional staffed surveys.

The next stage of technical development should focus on improvements to the object detector, most importantly by gaining the ability to reliably recognise common transport modes in LMICs such as three wheelers. This may be through finding a publicly available object detector which already supports this, or by training a model ourselves with our own training data. Although the former would be easier and more cost effective, the latter option would offer more flexibility and the ability to tailor the tool for better

performance in a given study area. Improving the detection speed would also be desirable, which could be done by enabling GPU processing instead of CPU or through the use of a different object detector, although this is not as high a priority as the cost of longer processing is minimal, and so does not significantly increase the in-service cost of the tool's use.

The next steps for maximising the utility of the tool focus on promoting greater awareness of the tool amongst our own colleagues in ITP, project partners in LMICs and clients in international development funding organisations. For project partners and clients in particular, we must pay attention to the perception of the tool and its outputs, and work to give them reassurance that the results can be trusted in real-world use. It will also be necessary to begin using the tool in real-world projects, not only to build a proven track record of its deployment, but also to provide more opportunities for learning and development. LMICs in Central Asia (Uzbekistan, Tajikistan and Kyrgyzstan) may be more suitable for this early stage of deployment than Sub-Saharan Africa (SSA), as they will require fewer improvements to the object detector. We will still target deployment in SSA in the short to medium term, however, as we believe the tool could be extremely useful in transport development work where alternative data sources are not available.

1. Introduction

This report describes the development of a computer-vision-based data collection tool which could be used to quickly and cheaply collect transport data in lower and middle-income countries (LMICs). An object detection model is used to recognise and count the transport elements seen in each input image, and these outputs are aggregated and displayed on a map. The tool can use imagery from a range of data sources to suit the local environment, such as crowdsourced data, commercial street-level imagery such as Google Streetview, or specially commissioned video surveys.

This project is a 'small project' in the Transport-Technology Research and Innovation for International Development (T-TRIID) competition, part of the High Volume Transport Applied Research Programme (HVT), funded by UK Aid from the UK Foreign, Commonwealth & Development Office.

This report is structured as follows:

- Section 2 contains the problem statement and a description of how our data collection tool attempts to address this.
- Section 3 provides a detailed description of the work undertaken during the development of the data collection tool.
- Section 4 presents the outputs of the data collection tool in our three test locations: Bo (Sierra Leone), Dushanbe (Tajikistan) and Maputo (Mozambique).
- Section 5 describes the dissemination activities undertaken as part of this project,
- Section 6 contains a summary of the previous sections and a discussion on the role our data collection tool can play in improving transport in LMICs.
- Section 7 contains a deployment plan for our data collection tool, describing the considerations and next steps that should be undertaken for successful roll-out of the tool in real-world transport projects.

2. Background

2.1 Problem statement

Cities in low and middle-income countries (LMICs) often have inadequate transport systems which fail to enable safe, affordable and efficient travel. It is common for local government bodies responsible for transport to lack the capacity to plan and regulate the sector, resulting in a transport system shaped by the individual and uncoordinated actions of private operators. Such under-planned and under-regulated transport systems often do not serve travellers well (poor safety standards, predatory fares, fill-and-go operation), do not serve transport workers well (low pay, long hours, job insecurity), and do not serve the city well (congestion, pollution)[3].

With insufficient capacity to carry out core planning and regulatory tasks, the resource-constrained transport authorities also tend to be unable to collect data, such as information on the number of informal transport vehicles on the road, routes of formal transport services or traveller flows between major origins and destination areas. This lack of information is a barrier to the regulation of the existing network, for example licencing of transport operators, and makes the targeting of transport interventions more difficult, if not impossible.

We, the authors of this report, encounter this problem through our work as transport consultants engaged by governments or international financial institutions (IFIs) to carry out development projects in LMICs. Although we very often work with local experts who know the city well, gaining an objective and detailed understanding of how an entire city moves is often not possible. In an ideal situation we would carry out surveys to collect the information we need, such as in-vehicle boarding and alighting counts, GPS route surveys and traveller interviews, however in reality many of the projects we work on have low budgets and there is little scope to do this. As a result, the projects we work on often rely entirely on pre-existing

[3] Ajay Kumar, Sam Zimmerman, and Fatima Arroyo-Arroyo. 2021. *"Myths and Realities of 'Informal' Public Transport in Developing Countries: Approaches for Improving the Sector."* https://www.ssatp.org/publication/myths-and-realities-informal-public-transport-developing-countries-approaches-improving

secondary data, which is variable in quality and availability, or the personal expertise and experience of our local partners.

A representative consultancy project in a city in an LMIC would be a transport strategy development project, with a budget in the range of a few tens of thousands to a few hundred thousand GBP. Here the consultant would work with the city government to identify objectives, policies and actions for the transport system in the future. While it might be possible to use best practice, case studies and experience for some of this, specific actions that are tailored to the city's specific circumstances can only be developed with a good understanding of the current transport environment.

In summary, the problem we seek to address with the data collection tool described in this report is as follows:

Many cities in LMICs do not have the capacity to collect transport data which can be used to plan transport improvements. Depending on the project, the data does not need to be extremely detailed – for example a citywide transport strategy may not need detailed traffic counts on all roads in the city. However, the data should be cheap to collect and process (both in monetary terms and in staff time) as the projects that would use them have limited budgets.

2.2 Our solution

We have developed a software tool that uses computer vision to detect and count transport-related objects in street-level imagery, aggregate them using a grid system, and plot the results on a map.

The outputs of the tool, an example of which is displayed in

Figure 4, are a set of choropleth maps showing a count for each object type in each grid square. Considering the limitations of input imagery used, these should not be considered actual counts of the number of each transport object. Instead, they should be viewed as a measure of the prevalence of the object type, relative to the rest of the city – with a high count being interpreted as “more prevalent” and a low count “less prevalent”.

These outputs are intended to provide high-level intelligence produced for a wide area at a low cost, rather than detailed analysis of specific locations. This makes the tool more suited to early-stage planning and strategy development projects, or for gaining a working knowledge of how people travel in different areas, for example when working in a new city for the first time. Although the outputs will not be as detailed as a two-way classified traffic count, they cover the entire city and allow a more complete picture to be built – a key requirement of data for city-wide planning work.

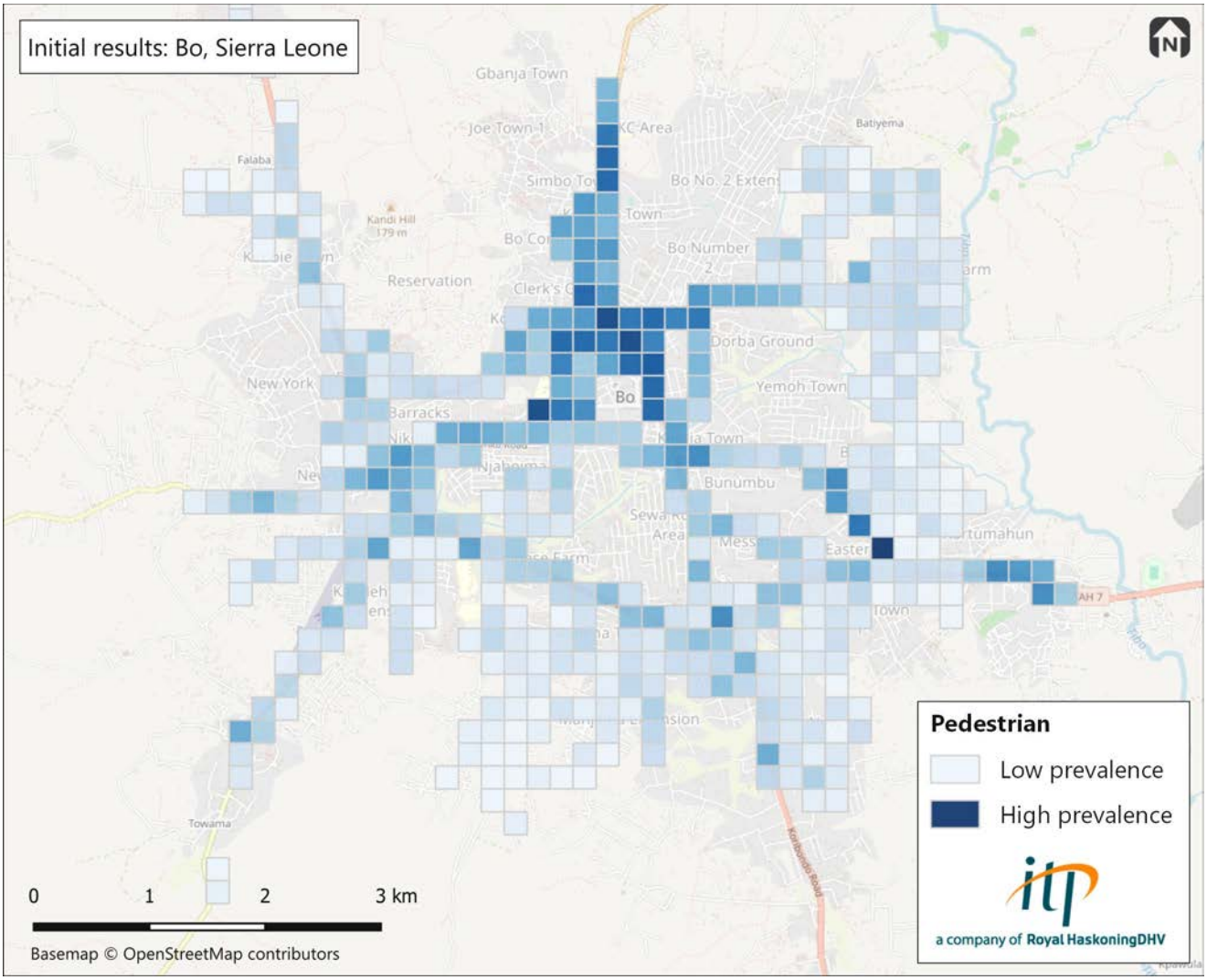
The tool has been built with the aim of being very cheap to use, requiring only a few hours of active staff time from start to finish. This will allow for city-wide data to be made available for low budget projects which would struggle to accommodate even a small number of static traffic counts. Although the outputs of the tool are not extremely detailed and may be based on a relatively small number of observations, they are preferable to no data at all, which is often the reality of transport consultancy work in LMICs.

Costs are minimised by using freely available crowdsourced imagery and open-source software, meaning the only costs required are staff time. The drawback of using pre-existing data is that the user has no control over the conditions of image capture, such as the time of day or day of the week.

We have designed the tool to be very flexible with regards to the source of the input imagery – any source can be used provided the images are of sufficient quality, have timestamps indicating when they were captured, and have precise location metadata to show when the image was taken. It is therefore possible to overcome a poor crowdsourced sample by supplementing it with additional imagery. The most promising data sources for this are video recorded specially for this purpose and commercially available street-level imagery from platforms such as Google Maps, Bing Streetside and Yandex Panoramas. Please see section 3.2 for more detail on the advantages and drawbacks of these data sources.

Finally, the tool is flexible in the geographic areas in which it could be deployed. The main requirements are geotagged street-level images, which are technically simple to produce, so in theory the tool could be used in any country. In reality, LMICs tend to have poorer data availability than HICs and this extends to crowdsourced imagery, so supplementary data sources may be required for wide scale deployment. Working with local governments and donor organisations in this area can help address this, however. For example, the local transport agency in Maputo, AMT, has taken the initiative to capture over 12,000 360° panoramas, which we used during our testing of the tool (please see section 4.3 of this report).

Figure 4: An example output from the development of the data collection tool in Bo, Sierra Leone. This choropleth map shows the relative prevalence of pedestrians in different parts of the city. A separate map is produced for each object type and counts can be combined to perform more advanced outputs – for example, identifying areas with high foot and vehicle traffic to highlight areas that may have an increased risk of traffic incidents.



2.3 Aim of this project

The focus of our work on this project is to develop a methodology for using street-level imagery to generate useful intelligence for transport consultancy projects and create a basic working implementation of this workflow (the “data collection tool”). This initial implementation is intended to be the basis of further refinement and development, building on our present work to create a market-ready product. We would like to note that, although we will produce some useful transport data during this project, these are not intended to be the primary output of this work and we cannot guarantee their accuracy and applicability if used for transport interventions.

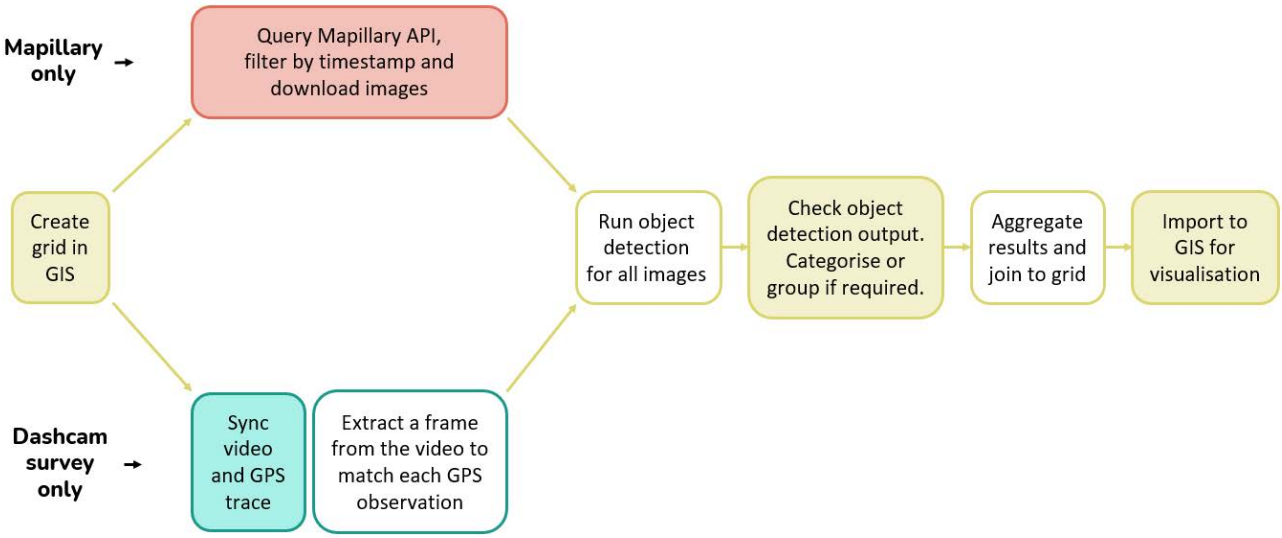
3. Implementation description

This section of the report describes the methodology of the data collection tool in detail, and highlights the key features and limitations identified during development.

3.1 Overview of data collection tool workflow

The overall workflow of the tool is shown in the
Figure 5.

Figure 5: Diagram of data collection tool workflow. Steps requiring more than minimal staff-time are denoted by filled boxes.



The first stage in the workflow is to create a square grid. This grid serves two purposes: it is used to query data from the Mapillary API in manageable chunks, and it is later used to aggregate the results.

The next stage is to source the input data: street-level images with timestamps and geotags. The exact workflow in this stage depends on the specific data source. During development we used two data sources: Mapillary[4] and a dashcam video survey we commissioned in Bo, Sierra Leone. Regardless of the data source, the key steps in this stage are the extracting or downloading of the images, and filtering by timestamp and/or location to ensure that the resulting set of images are likely to be suitable for comparison against each other. For example we may only wish to consider images taken between 9 am and 10 am on a working day or, if the available sample is limited, we may relax our criteria and allow any image captured in daylight hours on a weekday.

Once a suitable sample of images has been compiled, they are input into an object detection model, the "computer vision" component of the tool. Each image is checked against a list of user-input object classes (e.g. "person", "car", "bus") and the object detector counts the number of objects of each class identified in each image. The output of this stage is a comma-separated variable (CSV) file which lists, for each image, the image ID, the classes identified, and the number of detections for each class.

With our current object detector, we found that sometimes an object was assigned multiple classes, for example "bus truck". To correct for this, we manually check a subset of images to identify which class each of these multi-class detections should be assigned to, and manually classify them as such. This stage of the workflow is only necessary due to the characteristics of the object detector used, and could be removed or reduced with further development. The object detection component is largely independent of the rest of the data collection tool, so switching to a different object detector can be achieved with little to no changes to the other components.

Once the objects in each image have been counted, the output CSV is input into a post-processing and aggregation script, which produces the final output. A detailed description of this stage is presented in section 3.5 but, in summary, a single representative count is calculated for each object class in each grid square. These counts are joined to the grid shapefile created in the first stage of the methodology.

The final step in the data collection tool workflow is visualisation, which is done manually by importing the joined grid shapefile into GIS software. The styling used can vary depending on the outputs obtained, the size of the city surveyed, and the context of the project, but for demonstration purposes during development we created a series of choropleth maps using the grid polygons. One map is created for each object type, and each grid square is coloured according to the final count for the object type in that square. An example output can be seen in

[4] https://www.mapillary.com/

Figure 4 above.

The input images are downloaded or extracted as frames from a video and assigned to a grid square. This requires the images to be geotagged with precise locations, which is common for online imagery sources such as Google Streetview or Mapillary, but not video sources. If video is collected specially for use in the data collection tool, it must therefore be accompanied by a GPS traces.

Once assigned to a grid square, a person-in-the-loop filtering system helps the user define a time of day and day of week filter, to find an acceptable medium between the similarity of the sample and the geographic coverage of the data.

The filtered image files are then input into an object detection model, which identifies and tags specified transport elements when they are detected in an image. For example, an image may contain three people, two cars and one bus. These results are saved against the image ID and location tag so the results can be processed in different ways without having to re-run the object detection, which is slow without powerful computer hardware.

The object detection results are aggregated to the grid described earlier, to allow for easier interpretation of the results and to account for some of the intrinsic limitations of the data as collected. The output of this process is that each grid square is assigned a count for each object type seen in the city.

Finally, the counts for each grid square are visualised using a choropleth map in GIS, allowing for custom styling that suits the specific purpose required for the data.

3.2 Input data sources

The data collection tool can use any images that have timestamps, to allow filtering by day and hour, and precise location tags, to allow for the results to be plotted on a map. Almost any source of imagery meeting these requirements could be used, requiring only small changes to the image downloading and processing scripts (described below in section 3.3).

During development we trialled both crowdsourced imagery and bespoke dashcam surveys in Bo, Sierra Leone to develop workflows for both data sources. We expect pre-existing imagery to be the most common data source in real-world use due to its very low cost, with dashcam video only used to top up the crowdsourced data where required, as it significantly more expensive to collect.

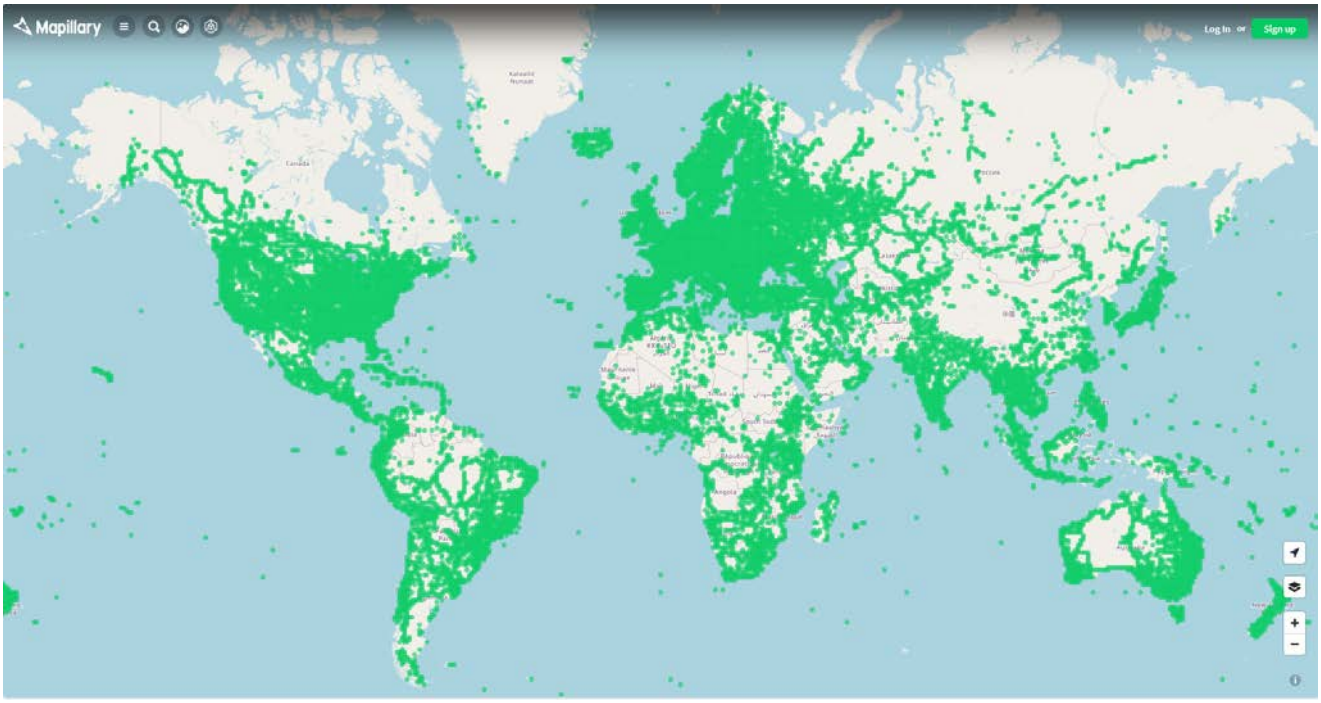
Pre-existing imagery	
+ Low cost Crowdsourced data is available for free and is often high quality, meeting all the requirements for use in the data collection tool. Using free imagery means the only costs of the data collection tool are a few hours of staff time.	**– Large variability in geographic coverage** As data availability is determined by the local surveying community, not all areas have good imagery coverage, even large cities. Where a city does have data, there may not be even coverage across the entire area. For example, it may only cover major roads, or surveying may be limited to a few neighbourhoods. In general, availability of street-level imagery is lower in LMICs than HICs. See Figure 6 for a comparison of selected sources. Projects in LMICs may therefore require “topping up” of data with other sources, such as dashcam surveys.
+ Quick and simple to obtain With API (application programming interface) access, data can be queried and filtered in minutes.	**– Potential for greater sampling bias** For example, crowdsourcing contributors typically use vehicle-mounted cameras, so wealthier neighbourhoods may be surveyed more often.

Thousands of images can be downloaded in a matter of hours.	
+ Required metadata is already present Imagery platforms typically include timestamps and geotags, simplifying the filtering and aggregation stages of our workflow.	**– No control over image capture conditions** Images may be collected on unfavoured days or times (e.g. weekends) due to the nature of crowdsourcing, whereas for transport data use it would be best to have all images captured at similar times of day and day of the week.
+ Commercial data sources are also available e.g. Google Street View, Bing Streetside, Yandex Panoramas. Commercial sources tend to have more even and widespread coverage in the locations they offer, compared to crowdsourced data. Commercial sources are cheap compared to bespoke dashcam surveys. Google Street View Static API currently costs 0.007 USD per image[5].	**– Largest commercial sources have restrictive licences** Google Maps terms of service apply to the Street View Static API, and prohibit both downloading of images and the creation of content from the data provided. Other platforms may have more favourable licences, however. For example, the primary data source we used during development, Mapillary, is a platform providing use to user-uploaded data. In uploading data to the platform, these users grant Mapillary a licence to *"use, reproduce, publish, create derivative works from, distribute, publicly perform and display"* the content[6]. Mapillary then provide access to this data under the CC BY-SA licence[7], which allows commercial use. It should be noted that the Mapillary Terms of Use[8] apply additional terms for commercial use – specifically, users must prohibit and have safeguards against reidentification or unblurring of the imagery and must ensure that Mapillary is notified of any breaches of these clauses. We do not believe these licences or terms of use will present any issues in the development or future use of the data collection tool.

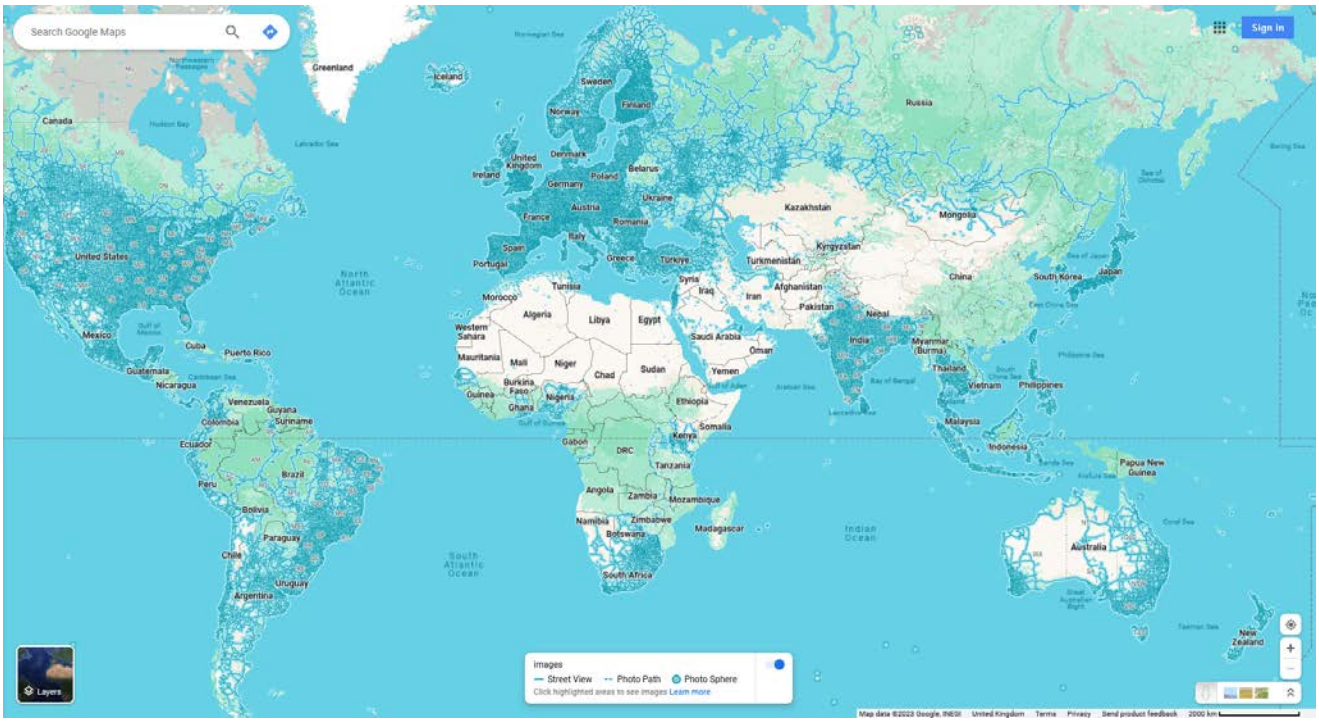
[5] https://developers.google.com/maps/documentation/streetview/usage-and-billing
[6] https://www.mapillary.com/terms, accessed: 2024-06-25
[7] https://creativecommons.org/licenses/by-sa/4.0/, accessed: 2024-06-25
[8] https://www.mapillary.com/terms, accessed: 2024-06-25

Figure 6: Comparison of global coverage of selected data sources

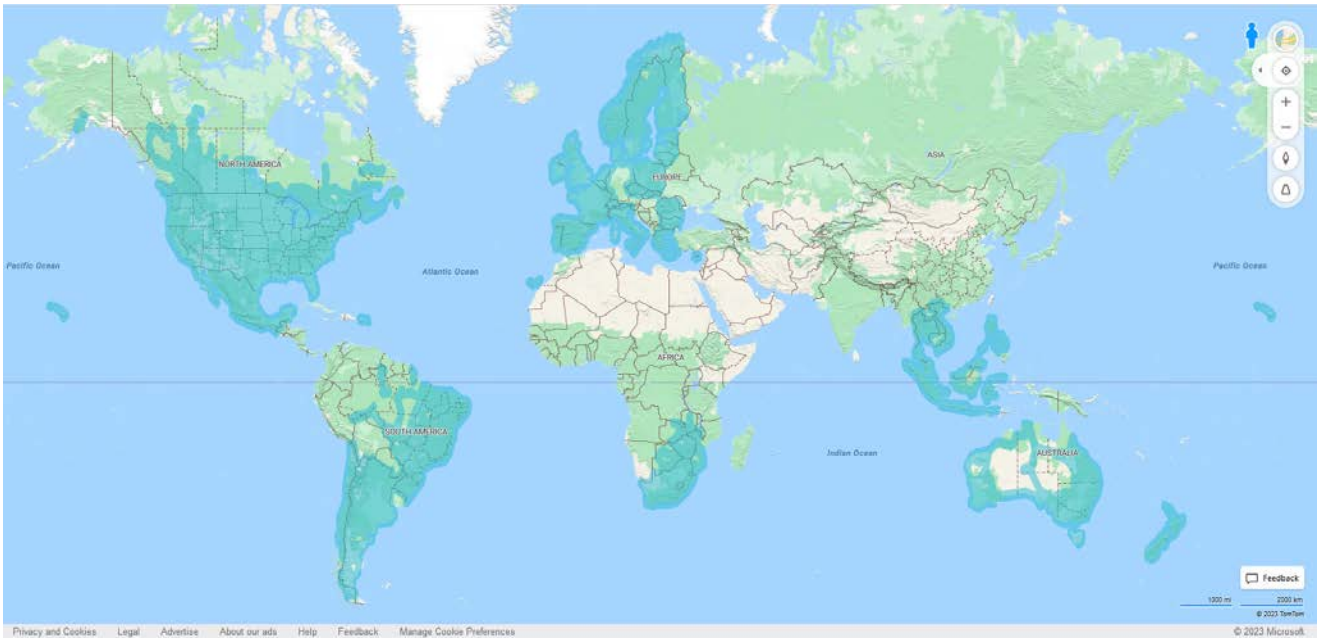
Mapillary: mapillary.com



Google Street View: google.com/maps



Bing Streetside: bing.com/maps



Bespoke dashcam + GPS survey	
+ Allows data collection conditions to be controlled The survey can be specified so data is only collected at certain times of day, and on certain days of the week. Other factors that may affect the data can also be avoided, for example public holidays or poor weather.	**– Expensive** As a dashcam survey requires staff to spend significant amounts of time travelling around the city, the cost of a dashcam survey can be as high as tens of thousands of GBP, for a larger city. In addition, procuring the supplier, specifying the survey, and monitoring and checking the results requires a not insignificant amount of staff time, adding to the monetary cost of the survey.
+ Coverage can be controlled to reduce bias The survey can be specified to provide good coverage across all parts of the study area, including areas which might not be surveyed often by crowdsourced data sources, e.g. less wealthy neighbourhoods, rural areas.	**– Technical aspects of the survey can be difficult to convey to surveyors** The output of the survey requires not only a video but also a GPS trace collected at the same time, which can be synchronised to the video. Not all videographers will be familiar with collecting GPS data, and we encountered this issue in the early stages of our GPS trace in Sierra Leone. Although we were able to work with the videographer to resolve it, it delayed the survey slightly and required more staff time than anticipated.
+ High image quality A high-resolution camera can be chosen, which will improve the performance of the object detection model.	**– Surveying can take a long time** It may take days, weeks or months for the data to be collected, which could delay project deliverables in real-world deployment. We anticipate the data collection tool being most useful at the start of the project, and it may not be possible to programme length data collection periods before work commences.

	– May require additional data processing Depending on the GPS collection method used by the videographer, the video and GPS trace may need to be synchronised before they can be used. In our Sierra Leone dashcam survey, the videographer recorded the GPS trace using a mobile phone, while the video was recorded separately. We therefore had to manually synchronise the two data feeds, which was time consuming and may not always result in good matching accuracy.

3.3 Image selection and processing

The purpose of this part of the data collection tool is to compile a set of geotagged images, representing broadly similar transport conditions, that can be input into the computer vision model. Our implementation of this takes the form of two Python scripts – one for the Mapillary data source, which queries the Mapillary API, filters the results it finds, and downloads the images, and one for the dashcam survey, which synchronises the video and GPS trace, then extracts a frame from the video for every GPS observation point.

Regardless of the specific data source and workflow requirements, the output of this component of the tool is a sample of images that have all been determined to represent sufficiently comparable transport conditions for the desired application. This sample is accompanied by a CSV containing metadata for each image (for example: IDs, timestamps, coordinates), allowing the outputs of the object detection stage to be aggregated in alternate ways without re-processing them.

3.3.1 Mapillary data selection and processing

Querying of the Mapillary API, processing and filtering of the returned data and the downloading of the filtered images was implemented using a Python script.

Identifying the images

In development we used the “Create grid” function in QGIS[9] software to create a 200 x 200 metre square polygon grid in the appropriate projected UTM (Universal Transverse Mercator) zone, which was then converted to the WGS84 coordinate reference system, for compatibility with the Mapillary API.

Each grid square is used as a bounding box to query the Mapillary ‘tiles’ API endpoint and return metadata about the images found in this area. The endpoint can return a maximum 2000 items per request, so the relatively small size of the grid squares helps limit the number of items returned each time.

The metadata contained in the response includes image IDs, sequence IDs, image coordinates and timestamps. The grid square ID is also attached to the image, for use in subsequent post processing and aggregation. Sequence IDs are important for use in the aggregation of results in the later stages of the workflow. Mapillary contributors record a sequence of images in a single trip through the city. Each image is assigned a unique image ID, and all images from the same trip are assigned the same sequence ID. Two images from the same sequence, captured seconds apart, will likely show a similar scene – the same people and cars, but from a slightly different viewpoint. We cannot track these objects between images using computer vision, this can only be done with video, so we will need to find a way to avoid double counting objects seen in multiple frames. For more information on this issue, please see section 3.5.

Filtering

Ideally, we would use only images that capture broadly similar travel conditions. As we cannot know what these conditions are like on every hour of every day, we simplify this by aiming to use a sample captured at a similar time of day, on similar days of the week. This is achieved by filtering the timestamp metadata.

As images are filtered out of the sample, the spatial coverage will likely decrease so a suitable balance between sample similarity and geographic coverage must be achieved. It is difficult to automatically

[9] https://qgis.org

determine when this balance has been achieved, particularly as the balance will differ significantly between projects.

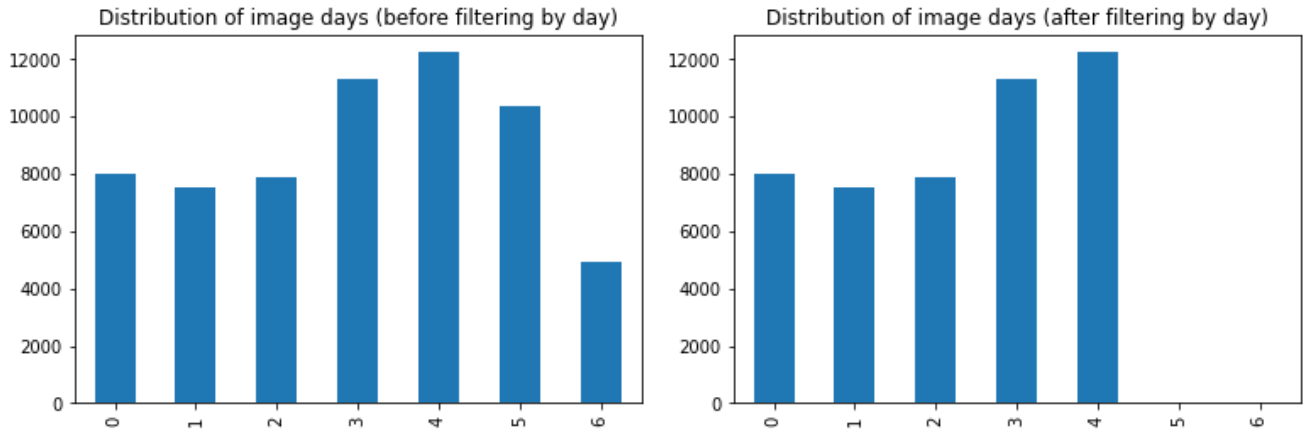
We therefore use an iterative “person-in-the-loop” filtering method where we display indicators before and after filtering, so the user can adjust the filters to return a suitable sample. Filters are applied using the timestamp to set which days of the week should be included, and which hours of the day. An example filter may therefore include only images captured on weekdays between 8 am and 9 am, to show morning commuter travel conditions.

The following indicators used to help the user determine a suitable filter set:

- A histogram showing the total number of images captured on each day of the week
- A histogram showing the total number of images captured by hour of the day
- A map showing the geographic spread of images across the study area, before and after filtering. This is the main way the user can tell if the filtered sample has sufficient spatial coverage for their needs.
- Two difference plots are produced to show the change in spatial coverage – one showing, for each grid square, the proportion of images remaining after filtering, and a second showing the same for image sequences.

Figure 7: Example of filter information histograms for Dushanbe, Tajikistan

Mapillary imagery in Dushanbe has good coverage from weekdays (where 0 in the graph denotes Monday) so we can remove weekends, which typically see quieter travel conditions than weekdays.



Having removed the weekends, the time profile of data shows two prominent peaks between 9-10 am and 2-3 pm. Assuming that Dushanbe has similar commuting times to other cities, a morning peak around 8-9 am and an evening peak around 5-6pm, from this profile we see we can create a sample with a large number of observations by taking the inter-peak period from 10 am – 4 pm.

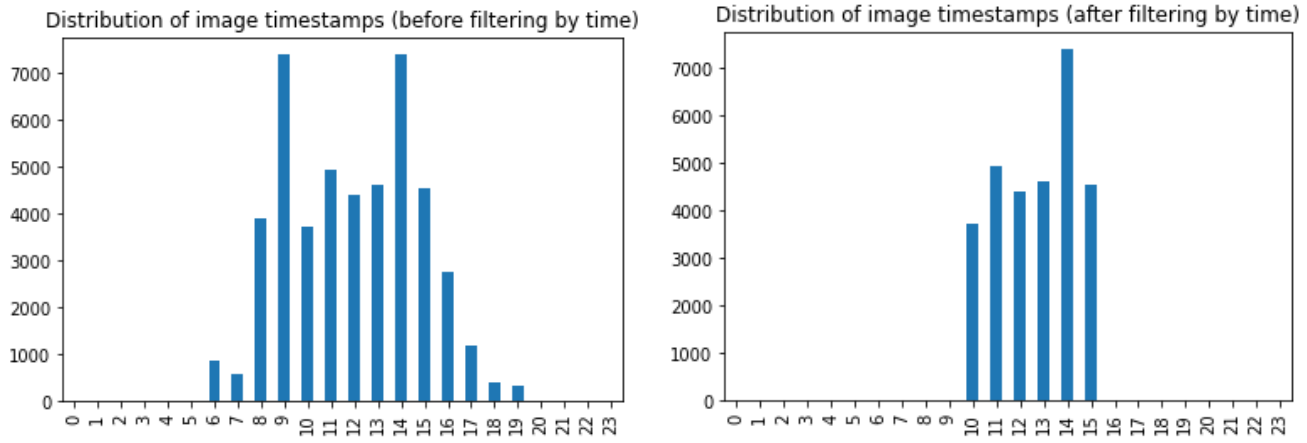
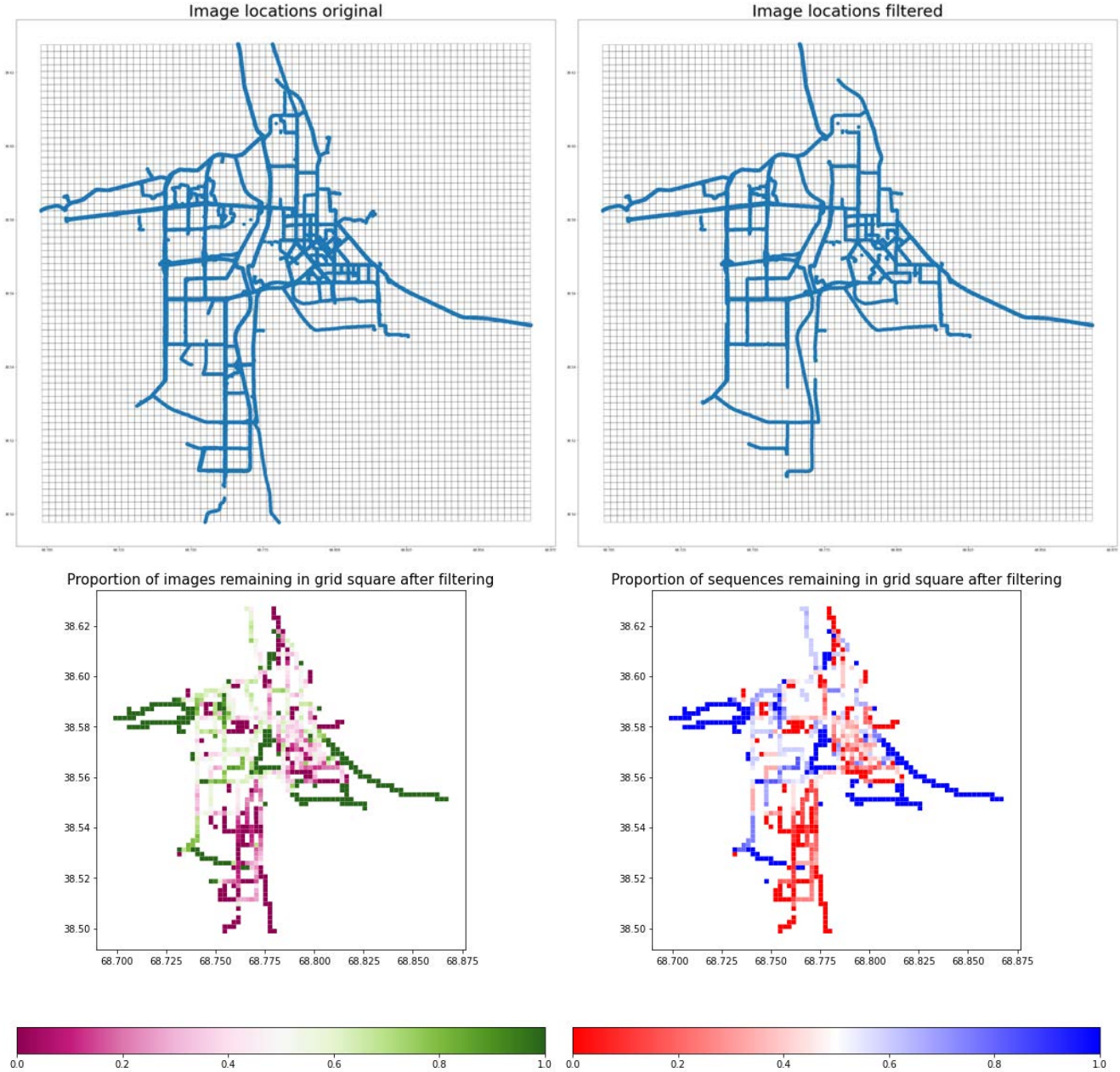


Figure 8: Spatial coverage resulting from filters specified in Figure 7 (Dushanbe, Tajikistan)

Having applied the filters described in Figure 7, the spatial coverage of the data in Dushanbe is largely retained. As can be seen from the plot in the lower left, areas in the far north, south, and city centre have seen significant amounts of data filtered out, but overall there is still sufficient coverage to draw conclusions about most parts of the city.



Downloading

No images are downloaded until the user sets the download flag to 'True'. Once a suitable filter has been applied, and the flag has been set, the script will use the Mapillary 'graph' API to download each image that remains.

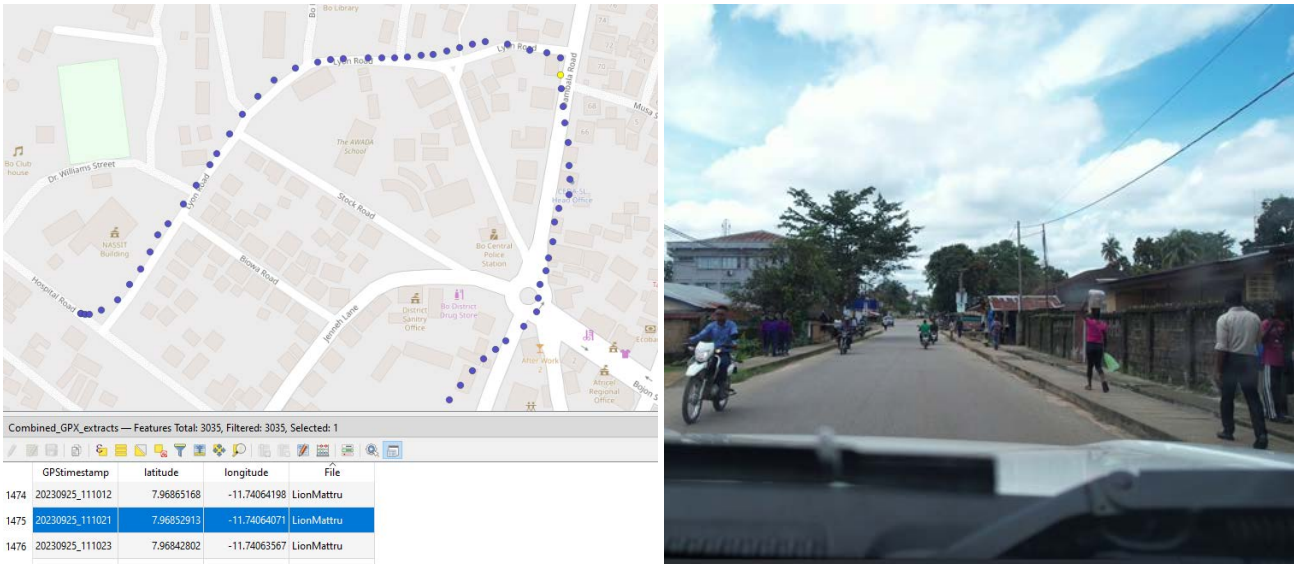
In addition to the images, an output CSV is also produced containing relevant metadata about each image – the imageID, sequenceID, coordinates, timestamp and the grid square it sits within. This CSV is used in the post processing and visualisation stage described in section 3.5.

3.3.2 Dashcam video processing

A dashcam survey carried out for use in the data collection tool will likely consist of multiple video files, with GPS traces either embedded in the video or provided as separate files. Ideally the GPS would be captured by the camera during recording and embedded in the video, in which case there is no need to synchronise the two data streams. During the survey we commissioned for the development of this tool, however, we received a separate GPX file recorded on a mobile phone at the same time as the video, which was captured by a separate camera.

We therefore synchronised each video with its corresponding GPS trace by finding a clearly identifiable location on the route taken by the camera vehicle, and noting the video timestamp at this point, and the GPS timestamp at this point. As the video and GPS were recorded by separate devices, this process had to be carried out manually.

Figure 9: Manual synchronisation of video and GPS



The sampling rate of GPS traces can vary but, for the purposes of this tool, greater observation frequency is desirable. Our dashcam survey in Sierra Leone sampled the location every second (1 Hz), which provided a reasonable number of observations.

A single frame was extracted from the synchronised video to match each GPS observation point using the OpenCV Python library. As the camera captures 30 frames per second but the GPS is sampled only once per second, multiple frames could reasonably be assigned to each point. To work around this we chose the first frame from the nearest second to the GPS observation, but it would also be possible to choose a random frame from any of the 30 frames captured in the same second.

Another alternative would be to extract multiple frames for each GPS observation, which would be desirable if there is a strong chance that a single frame would be blurry – for example if the camera was operated with a slow shutter speed. Having multiple frames for each GPS point would make it possible to average out bad frames or images that the object detector struggles with, at the cost of processing time during the computer vision component of the tool.

Once extracted, the video frames were tagged with a unique ID to enable matching of the image and its contents, and its coordinates taken from the GPS trace.

Unlike pre-existing data, the dashcam video should require no filtering based on time or date as good survey design would ensure that only relevant data is captured.

Figure 10: Example of frame extraction (please note this is for demonstration only – for use in the tool, one frame was extracted for every GPS observation, rather than every tenth observation as shown in this diagram)

3.4 Object detection

The core component of the computer vision data collection tool is the object detector, the software that uses computer vision to identify and count transport objects in each of the input images. The object detector can count the number of each object class identified in each image, as well as record the coordinates of each object within the image, allowing for more advanced processing do be done (for example, by checking the overlap between a detected person and a detected helmet, it is possible to determine if a motorcyclist is wearing a helmet).

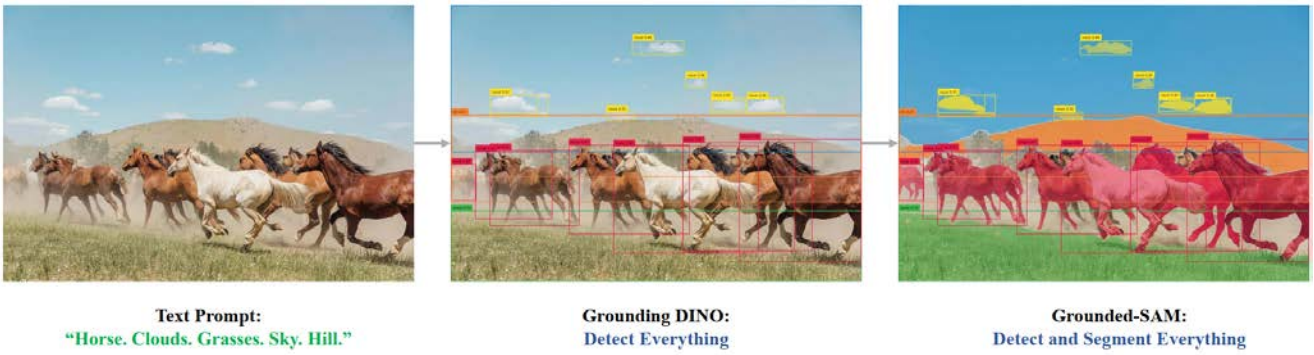
The following description applies to object detector we used for development, *Grounded SAM*[10], but any other object detector could be used provided it can output a similar CSV file listing the detection counts. The object detector is a standalone piece of software, so very little modification to the rest of the tool is required when it is changed.

[10] https://github.com/IDEA-Research/Grounded-Segment-Anything

3.4.1 Description object detection system

Grounded SAM is an object detection framework composed of the *Grounding DINO*[11] object detector and the *segment-anything*[12] image segmentation model, allowing the user to use text prompts to specific the objects they wish to detect in an image. *Grounding DINO* is an open-set object detector, meaning that it is able to detect objects it has not been trained to identify. S*egment-anything* divides the pixels of an image into groups ("segments"), defining the extents of each object and its position within the image.

Figure 11: Object detection and segmentation with Grounded SAM (Source: Grounded SAM Contributors, https://github.com/IDEA-Research/Grounded-Segment-Anything)



Combining these abilities in *Grounded SAM* provides many advantages for transport data collection, as it allows for more transport-relevant objects to be detected, even if they are not found in the training datasets on which the object detector was trained. We made use of this in our Sierra Leone test case to recognise three-wheelers, but this capability could be useful in other contexts, for example if a form of shared taxi is recognisable based on a distinguishing feature (such as yellow marshrutka in Kyiv, or red minibuses in Hong Kong).

In practice, the workflow in the data collection tool for object detection is very simple. The user specifies the directory containing the images to be processed and enters a text prompt describing the objects to be detected (for example: "person . car . bus . truck". Full stops divide the classes, allowing for inputs containing spaces to be used - e.g. "yellow bus").

The object detector will then process each image in turn and output a CSV file containing a list of all image IDs, with the number of each detected object in that image. The output of an annotated version of the input image is optional. While not strictly necessary for the data collection tool, it is useful to spot check the detections in a few images.

[11] https://github.com/IDEA-Research/GroundingDINO, see Liu et al. (2023)
[12] https://github.com/facebookresearch/segment-anything, see Kirillov et al. (2023)

Figure 12: Detection of a bus

Figure 13: Detection of a keke (three wheeler) and a motorcycle

Figure 14: Detection of a person and a car

Figure 15: Detection of a heavy truck

Figure 16: Detection of truck trailers

3.4.2 Testing of identification of three wheelers

Three wheelers, called “keke” in our testing location, Sierra Leone, are a common mode of transport in LMICs around the world. They are not common in HICs, however, leading to the possibility that they are not well represented in the datasets used to train object detectors. For example, three wheelers by any common name were not found in the COCO[13], Objects365[14] or Open Images[15] datasets, which are commonly used to train and test object detection software (see e.g. Liu et al. (2023)).

We tested Grounded SAM’s ability to detect three wheelers with different prompts: “three wheeler”, “keke”, “auto rickshaw” and “tricycle”. For the test image, “three wheeler” was the only prompt which returned a correct detection – all others included the three nearby motorcycles.

Figure 17: Testing of identification of three wheelers

[13] https://cocodataset.org
[14] https://www.objects365.org
[15] https://storage.googleapis.com/openimages/web/index.html

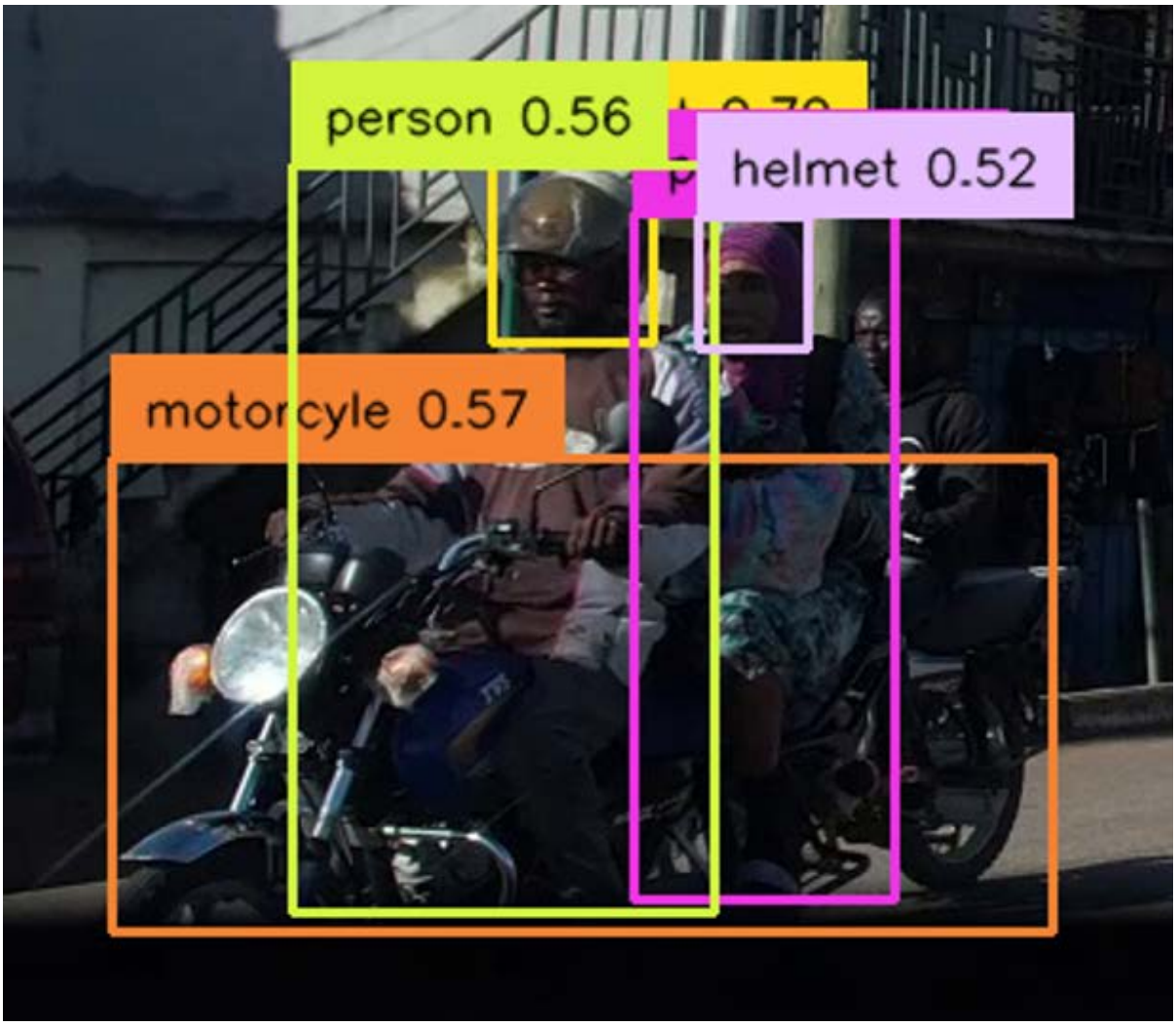
3.4.3 Methodology for identification of okada

Motorcycle taxis, known as “okada” in Sierra Leone, are a common informal transport mode in many countries around the world. Often un- or loosely regulated, it is difficult to tell okada apart from private motorcycles as they may not have any distinguishing markings.

We developed a methodology for distinguishing okada from private motorcycles based on our observations in Bo, Sierra Leone. It appears okada drivers often wear helmets and coats, whereas their passengers and private motorcyclists do not. On this basis we would assume that, in Bo, any motorcycle with a helmeted driver and one or more passengers is a motorcycle taxi.

Implementing this in the data collection tool requires the use of the coordinates and boundaries of the detected objects. A motorcycle taxi is therefore defined as a collection of objects where the “motorcycle” bounding box intersects with multiple “person” bounding boxes, and only one of those “person” detections intersects with a “helmet” detection.

Figure 18: An example of an okada in Bo, Sierra Leone, showing helmet use by the driver but not the passenger. Note that the detection of a helmet on the passenger is erroneous, likely due to the shadows present in the image.



3.4.4 Detection of parked cars

Car parking is an important consideration in urban transport, affecting not only traffic flows but also the effectiveness of other transport policies and interventions, for example parking restrictions in city centres can dissuade private car use and increase public transport patronage.

We attempted to distinguish parked cars from moving cars from the Mapillary and dashcam imagery we obtained but were ultimately unsuccessful. We tried using the location of the car in the frame, as illustrated in

Figure 19, but this was not a solution because the camera is also moving – in the distance the car is close to the centre, then it moves towards the edge of the frame as the camera car passes it. This is the same behaviour as seen when viewing a moving car so cannot be used as the basis for parking detection.

Grounded SAM can identify the roadway as an object, and its extents, so one option could be to look for cars that only partially overlap with the roadway. But this would also not be a reliable method – it depends on the car being parked at the side of the road, which is not always the case, and the road edge detection being accurate. This would require both an unobstructed view of the road edge, which will not always be possible, and it requires there to be a distinct road edge. Road quality varies significantly in LMICs, with kerbs and pavement not always present, and many roads entirely unpaved.

We believe the most reasonable solution to parking detection would be to use the full video along with object tracking. This may still not be viable, however, as object tracking requires persistent objects in the background to use as a reference for motion, which are not present due to the imagery being captured from a moving vehicle

.

Figure 19: The position of a parked car relative to the centre of the image changes as the camera moves closer. The same behaviour is seen in moving cars, so this method cannot be used to distinguish moving and parked cars.

3.4.5 Detection performance

We found the performance of the object detector to be variable when applied to both Mapillary and dashcam images. People are recognised well, as are passenger cars and motorcycles. Buses, trucks and vans, however, tend to only be clearly identified when relatively close to the camera – when further away it is common for the classification of the same vehicle to vary between frames.

Three wheelers have proved the most challenging to detect consistently – they can be detected from all angles in certain conditions but are prone to misclassification, and the “three wheeler” class can also be applied to other modes in error as well.

Some misclassifications are reasonable to human reviewers, for example SUVs, pickup trucks, vans and smaller heavy trucks can look similar and could be assigned to different classes depending on how they are used. Others errors, however, are obviously incorrect when reviewed by a human and can significantly affect the results returned by the tool.

Key issues identified are:

- The object detector struggles to identify objects in the distance, approximately 50 metres or more from the camera. We speculate that this is due to image quality and the small size of far-away objects in the image, which means they are represented by a relatively small number of pixels, increasing identification difficulty. This ‘short-sightedness’ affects the way detection results are aggregated, as described in section 3.5.
- The use of the tag “three wheeler” does not detect auto rickshaws consistently and often results in motorcycles and heavy trucks being misclassified as three wheelers. This may be due to a lack of three wheelers in training data. Improving detection may be possible by using a different object detector or by training the model ourselves, however this can require significant staff time to tag images and is often computationally expensive.

Further research and experimentation is required to determine if these issues could be addressed in the existing object detector by tweaking configuration settings or prompt combinations. If these actions are not effective, it is, by design, easy to modify the data collection tool’s workflow to use a different object detector. We present a deployment plan in Appendix A: , which notes these as priority tasks for successful roll-out of the tool.

Unfortunately these issues may persist to some degree, even with the most advanced and effective object detection technology. Lighting conditions, nearby objects blocking the view of other objects, and oblique viewing angles all decrease performance, but cannot be controlled in the real world. There is also a trade-off between magnification and field of view – for example it is possible to obtain a detailed image of a far-away object using a telescope, but the field of view would be low and so would not cover all the other objects visible from the camera’s position. There are also challenges arising from the use of imagery taken from a moving vehicle, for example difficulty in using object tracking. This requires a static background for use as a reference for movement, which is not present when the camera is moving as well.

Other object detectors were tested prior to the project and provided better detection of certain classes (for example: trucks and buses), but were inferior in other respects, including detections of three wheelers and novel or uncommon items. This lesser flexibility is a significant trade off that would need to be considered if the object detector were to be changed.

Figure 20: Example of a low quality detection of a keke (three wheeler)

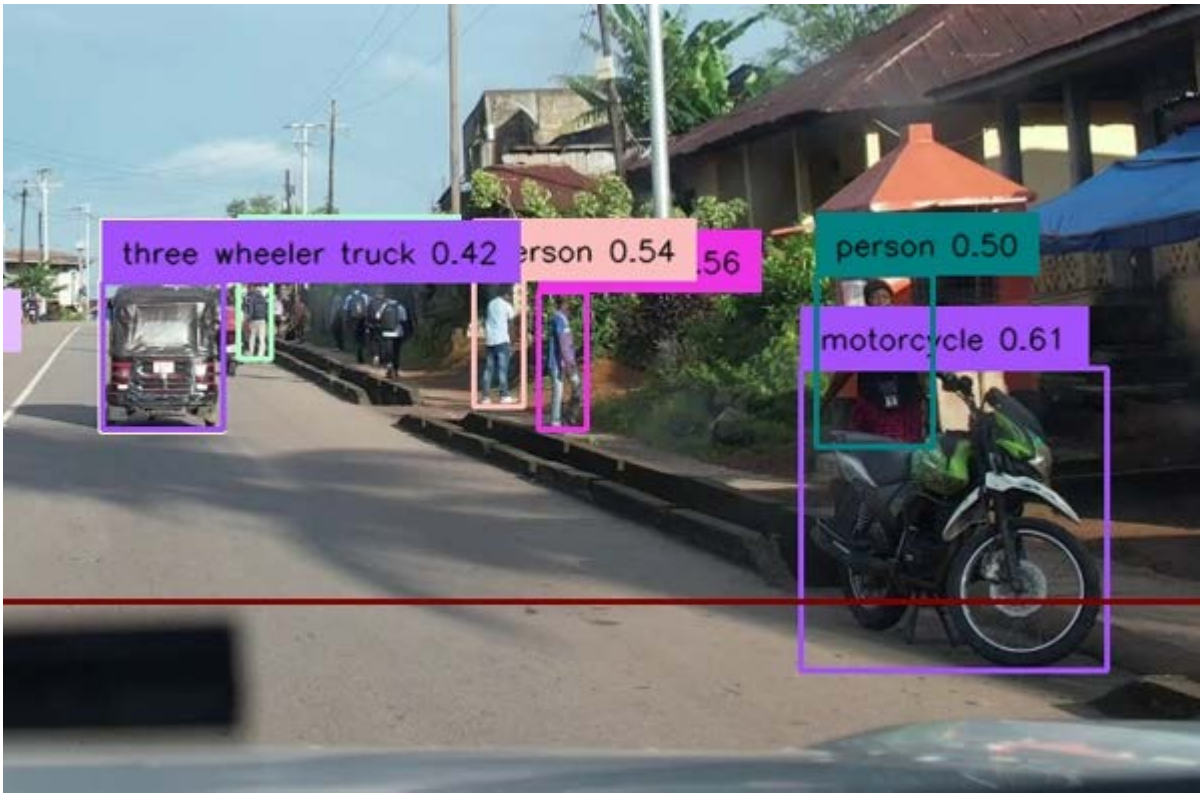


Figure 21: Example of incorrect detection of a minibus and a motorcycle

Figure 22: Example of "short sightedness" of the object detector, as it is unable to detect any of the motorcycles which are obvious to a human viewer

3.4.6 Processing speed

Collecting crowdsourced imagery for an entire city can return thousands of images, all of which will need to be processed to detect transport elements. Mapillary hosts over 12,000 images captured on weekdays in Bo, whereas Ulaanbaatar, Mongolia, has over 120,000. The speed of object detection can therefore pose an issue if the tool is to be used in a city with large amounts of data.

During development we installed *Grounded SAM* on a workstation laptop running a virtualised instance of Ubuntu 20.04 through Windows Subsystem for Linux. GPU-enabled calculation could not be utilised due to driver issues so the object detector was configured to use the CPU, which is a significantly slower mode of operation. In this configuration each image is processed in approximately 8 seconds.

This level of performance is acceptable for smaller cities such as Bo, which can be completed by scheduling processing overnight or on weekends, but is insufficient for Ulaanbaatar, which would take over 11 days.

The most cost-effective performance improvement method would be to solve the driver issues preventing GPU processing on our existing laptop. Alternative solutions would be the installation of object detection software on multiple computers to divide the processing workload, or the use of commercial cloud-based GPU servers. Exploring these options is an important task for real-world implementation of the data collection tool, and this is reflected in the deployment plan in Appendix A: .

3.5 Post processing and aggregation

The output of the object detection stage of the data collection tool is a CSV file containing the number of each object class detected in each image. This CSV is used in the final stage of the workflow to aggregate the results in each grid square and produce a final output file that can be used to visualise the results in GIS software.

Figure 23: An example CSV output from the object detector

```
File Name,Object,Count
1195648367530292.jpg,person,3
1195648367530292.jpg,car,9
1195648367530292.jpg,van,1
1195648367530292.jpg,bus van,1
1195648367530292.jpg,car van,1
1195648367530292.jpg,traffic light,1
1195648367530292.jpg,truck,3
1242085489580295.jpg,car,1
834152093850558.jpg,car,1
767149347325217.jpg,car van,2
179130814093626.jpg,car,7
179130814093626.jpg,car van,1
179130814093626.jpg,truck,1
149866277112142.jpg,car,7
149866277112142.jpg,car van,1
149866277112142.jpg,person,2
149866277112142.jpg,traffic light,1
```

Typically, images are recorded during a surveying trip through the area of interest, so multiple images are collected in sequence. Each image is therefore assigned a sequence ID, either automatically assigned by Mapillary or derived from the video filename in the dashcam survey. Images of the same sequence that were taken a short time apart will likely show a similar picture, so it is necessary to avoid double counting when processing their object counts.

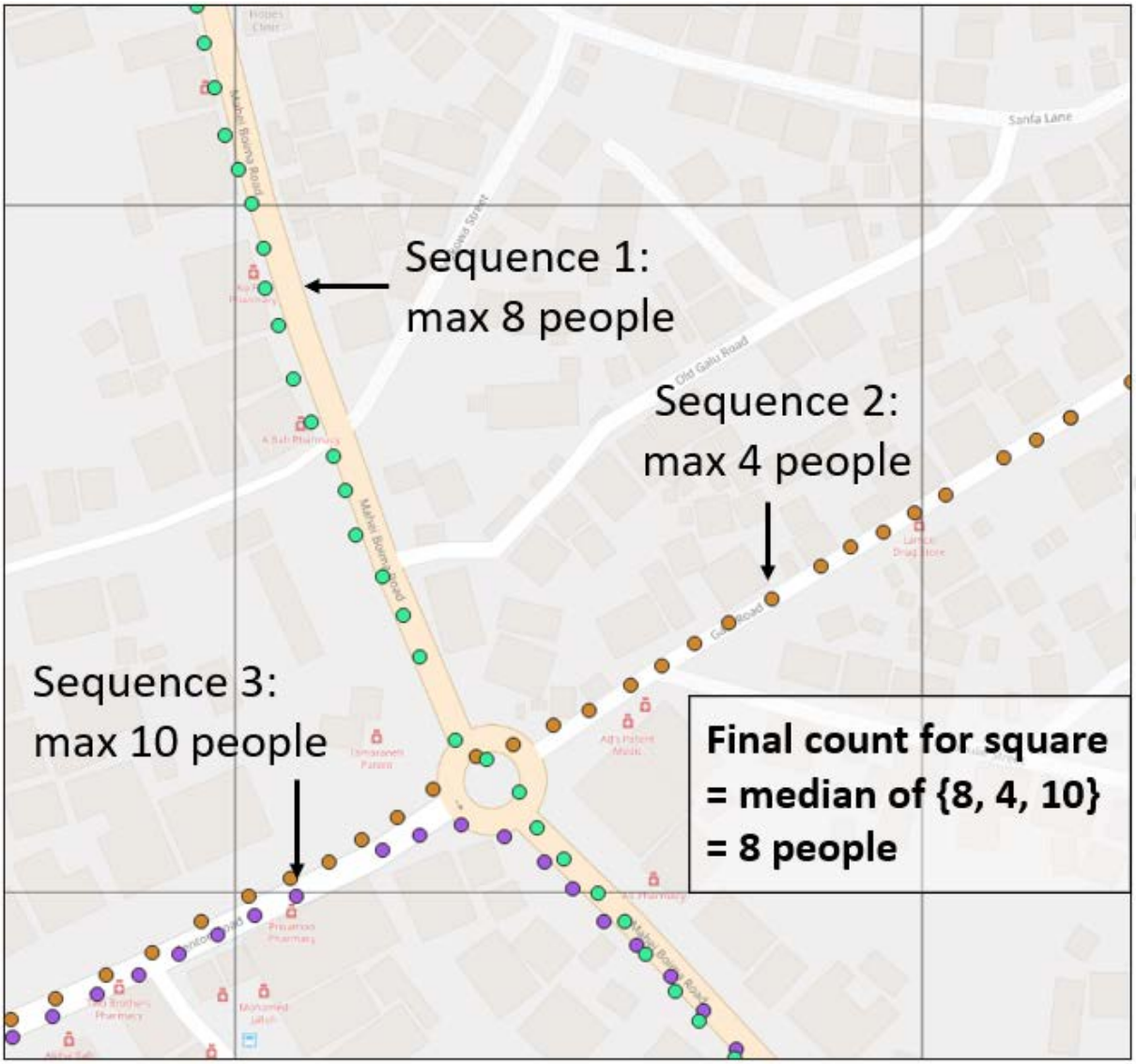
To account for this we take the maximum count for each object class, from all images of a given sequence located in a given grid square.

The mean or median could also be used but in testing this returned results with greater variability, most likely due to some images returning higher or lower counts than their neighbours. This could be caused by

the performance of the object detector, as some images may be easier to detect objects in than others, or it could be caused by lighting conditions, blurriness, or objects blocking the camera's view.

If there are multiple sequences in the grid square, these maximum counts for each class are averaged using the median. These calculation steps are illustrated in Figure 24.

Figure 24: Aggregation of object counts



As discussed in Section 3.4.5, the object detector used is 'short-sighted' and struggles to detect far-away objects. The counts returned for each image are therefore the number of objects in the immediate vicinity of the camera, within approximately 50 metres.

Calculating the total number of objects within a grid square will therefore require using counts from multiple images, which is complicated by the issue of double counting described above. It may be possible to avoid double counting using object tracking but, as discussed in Section 3.4.4, this is difficult to implement using still images from a moving camera, so is unlikely to be feasible.

We therefore present the outputs of our tool not as counts of the number of each object type seen in a particular grid square, but as a relative indication of its prevalence in that grid square. Low values should be interpreted as "less prevalent" and high values as "more prevalent". These values are comparable between different object types so it would be possible to infer that buses are twice as prevalent as cars (subject to regular considerations regarding the size and providence of the input data sample).

3.6 Visualisation

The output of the post processing and aggregation stage is a CSV file listing all of the grid squares and the final count for each object type in each square.

Figure 25: An example of the output of the post processing and aggregation stage

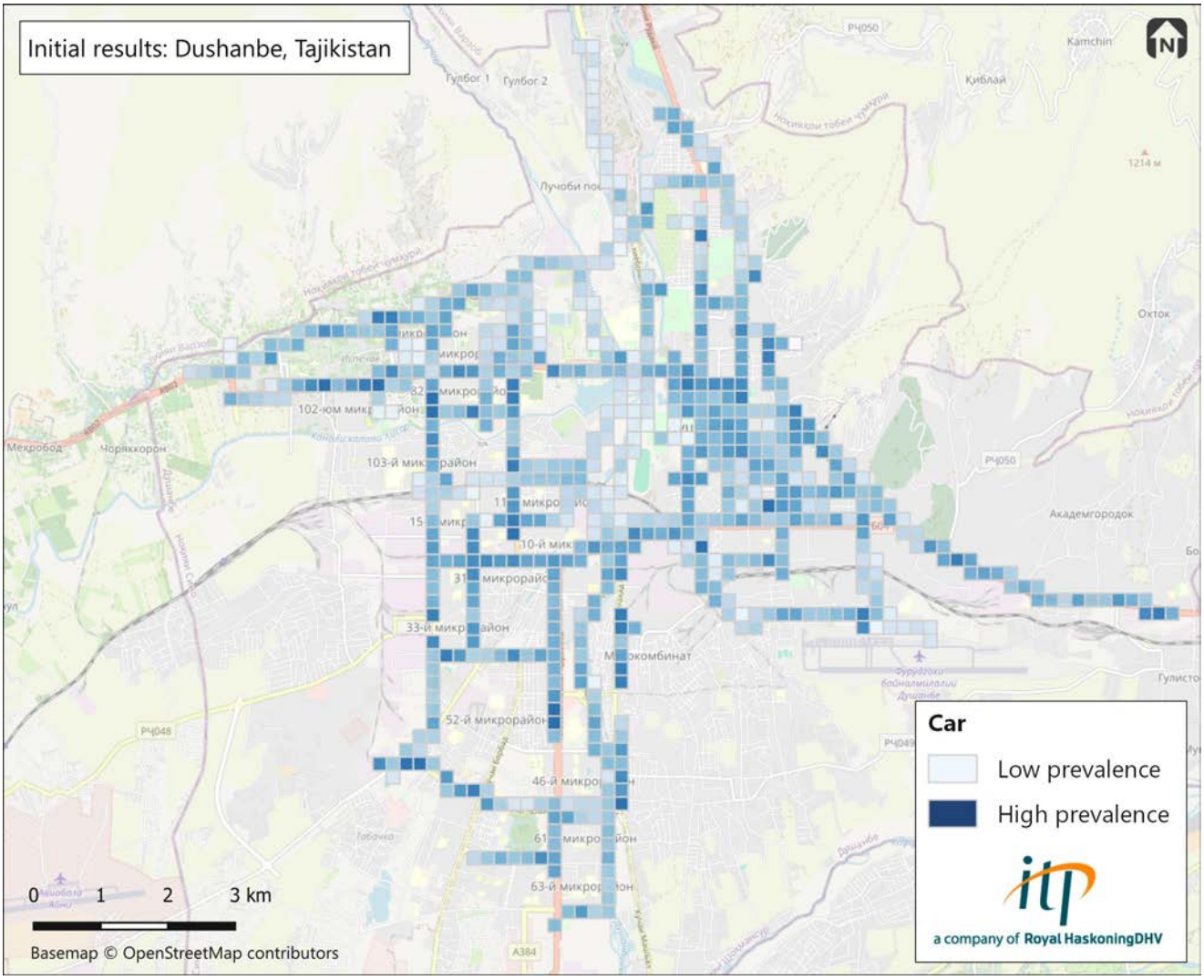
	gridID	bus	car	motorcycle	person	truck	three wheeler
1	1165	0	1	3	8	0	1
2	1166	0	1	2	4	0	0
3	1207	0	1	1	5	0	0
4	1208	0	1	4	12	0	2
5	1209	0	1	4	10	0	1
6	1251	0	2	2	11	0	2
7	1252	0	3	3	8	2	2
8	1253	0	2	4	7	0	1

This can be imported into any GIS software tool for visualisation. During development we used QGIS[16] to create a choropleth map for each object type, as shown in Figure 26. This example shows the results aggregated to the 200 m grid however, for this size of city, it may be preferable to use a larger grid size. This is a relatively simple procedure that can be carried out after the images have been processed by the object detector, so can be carried out quickly if required.

We note these outputs can also be used to derive more advanced statistics, such as highlighting areas with high foot traffic and vehicle traffic to identify priority areas for safety interventions. We did not test this analysis method during development, however.

[16] https://qgis.org

Figure 26: Example output showing data for cars in Dushanbe, Tajikistan.



3.7 Additional data collection method identified during development

An exciting co-benefit to the downloading of large numbers of geo-tagged and timestamped images is that the metadata can be used to calculate the travel speed of the survey vehicle as it moves around the city. This data is useful for a variety of transport work, for example network analysis or operational planning of new public transport routes.

Although it is relatively simple to collect GPS traces, it requires the time of surveyors to collect and so can be relatively expensive. As with the main purpose of the computer vision data collection tool, being able to collect GPS speed data for free would be extremely useful for transport projects in LMICs.

The extraction of GPS data from Mapillary is an add-on to the data selection and processing script described in Section 3.3. Little additional processing is required of the image metadata to compile a GPS trace, with the main operation being extracting the coordinates into separate longitude and latitude columns, and ordering the observations sequentially for better compatibility with GPS processing and visualisation tools.

During development we extracted GPS data for two cities, Dushanbe (Tajikistan) and Maputo (Mozambique), and processed this data using our own GPS matching software. Results can be seen in Section 4 of this report.

3.8 Software Licences

The data collection tool uses third party software packages or libraries, most of which have their own licences. We have reviewed the Python libraries used, and all licences are suitable for the intended use case of the data collection tool, including with regards to commercial use. The object detector used during development, *Grounded SAM*, is released under an Apache 2.0 licence[17], which also allows commercial use. We note that other object detectors may use different licences, which would have to be taken into account when considering further development of the tool.

[17] https://github.com/IDEA-Research/Grounded-Segment-Anything?tab=Apache-2.0-1-ov-file#readme

4. Demonstration case studies

4.1 Bo, Sierra Leone

We used Sierra Leone's second largest city, Bo, as our test city during development. Our team has recent experience working in the capital of Sierra Leone, Freetown, which gave us access to reliable local partners to aid in data collection, and Bo has good availability of crowdsourced data though Mapillary.

4.1.1 Mapillary data

Mapillary data covers all major roads in the city and a large number of minor roads, many of which are unpaved. Most images have been captured on Sunday (16,000 images), although a reasonable number have been recorded on Mondays and Tuesdays. Including images from all days, the hourly profile of image capture is relatively even, with most images captured in the interpeak period (10 am – 4 pm). When filtering out weekend captures, however, there is a prominent peak in observations around midday.

Using the iterative filtering method described in Section 3.3, we obtained a sample of 8,836 images captured on Mondays and Tuesdays only, predominantly between the times of 11 am and 2 pm. As can be seen in Figure 29 the geographic coverage was greatly reduced from the unfiltered dataset, creating the need to "top up" the data with the dashcam survey.

Figure 27: Mapillary data availability in Bo (source: Mapillary.com)

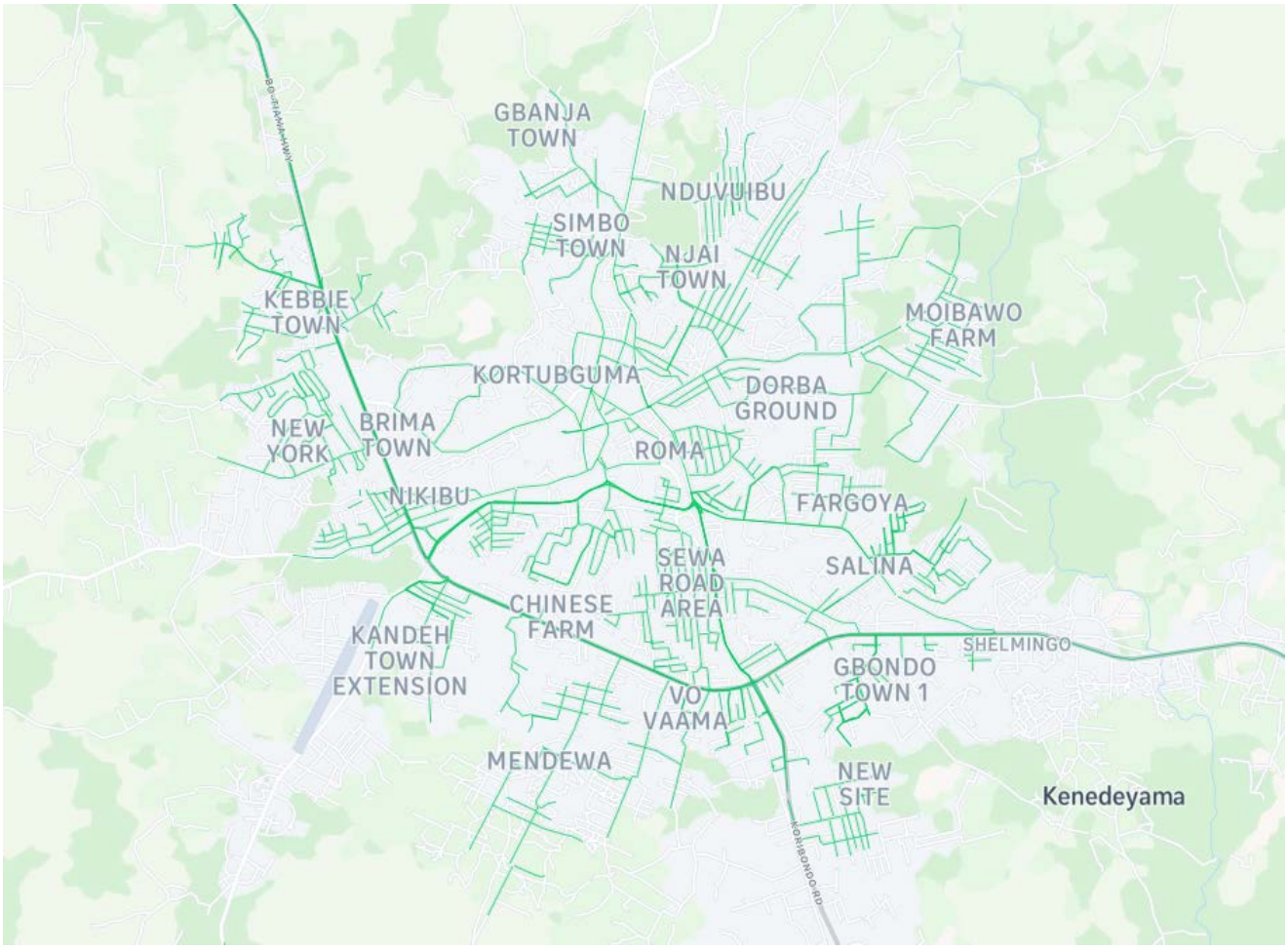


Figure 28: Temporal distribution of Mapillary image capture in Bo. Left: images captured by day of week [0 = Monday], Right: images captured by hour of day, weekdays only).

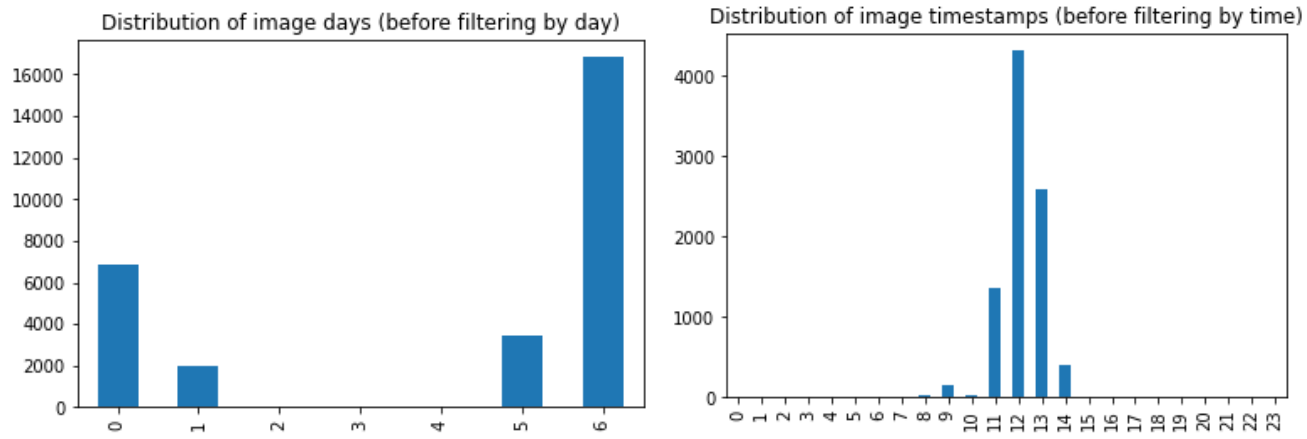
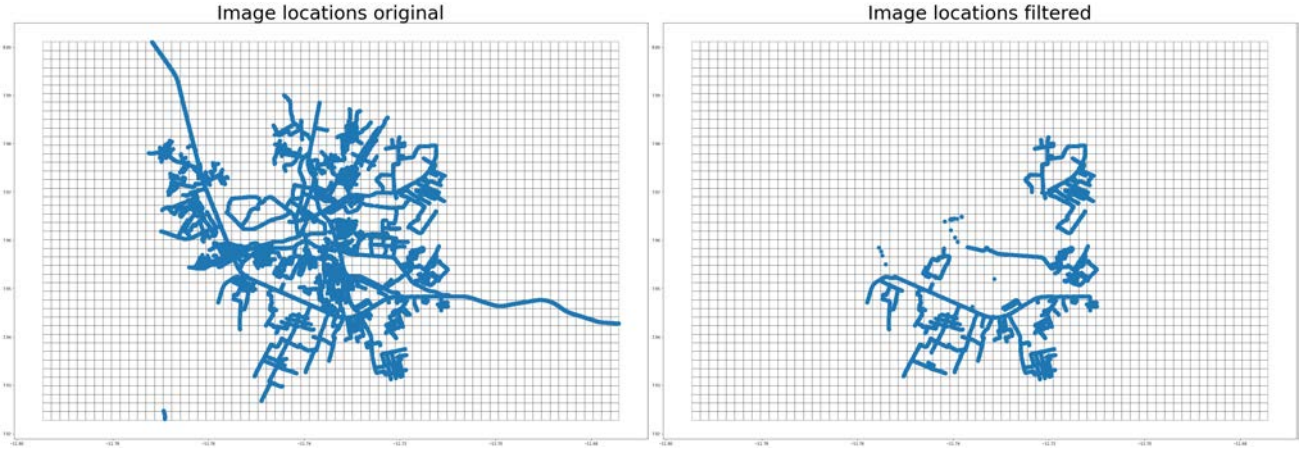


Figure 29: Geographic coverage of Mapillary data in Bo before and after filtering



4.1.2 Dashcam survey

To fill the gaps in the Mapillary data we partnered with a data collection company in Sierra Leone, Dalan Development Consultants[18], to carry out dashcam surveys in Bo. These surveys used a dashboard-mounted camera in a car to collect video while a GPS traces was simultaneously recorded using the *Geo Tracker*[19] smartphone app.

The survey was to be completed over two days, with the first surveying the major roads and the second some of the larger minor and residential roads. The survey was specified so that all observations were carried out between 7 am and 9 am on weekdays only. Unfortunately it was not possible to match the availability of data from Mapillary as the Mapillary data selection process had not been developed when the survey was designed.

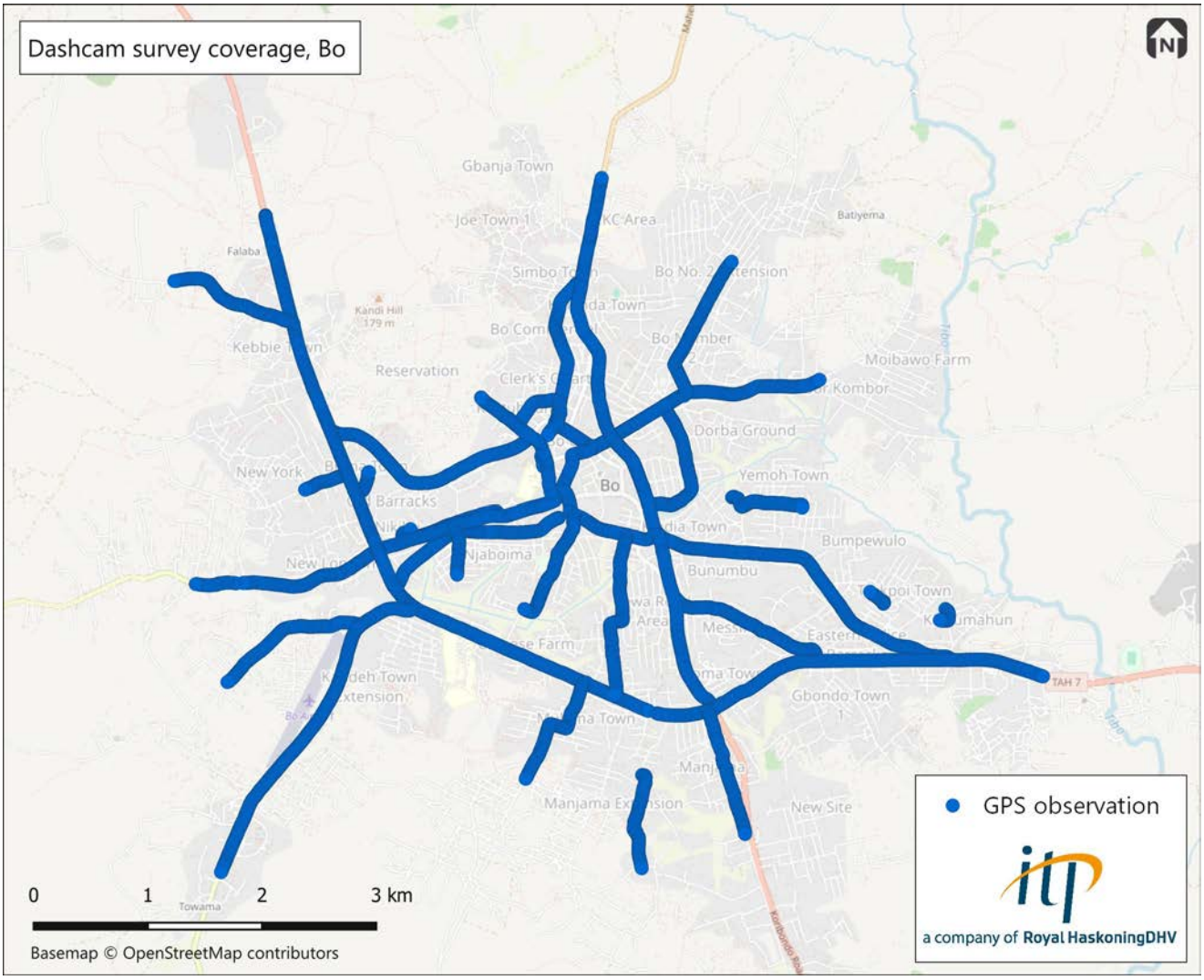
42 km of road was surveyed on the first day, from which 3,031 images were extracted to match the GPS observations. A map of survey coverage is shown in

[18] https://www.dalanconsult.com
[19] https://geo-tracker.org

Figure 30.

The second day of surveying was postponed due to the start of the rainy season making unpaved minor roads difficult to pass by car. Resumption was further delayed by political instability in Sierra Leone following violent confrontations in Freetown in late November 2023, but the second survey was completed in February 2024.

Figure 30: Dashcam survey coverage in Bo



4.1.3 Results

The results obtained with the data collection tool show walking, motorcycles and cars to be the main modes of transport in Bo, with three wheelers and buses significantly less common. This matches our expectations following conversations with our local partners, who highlighted the role of okada in the city.

Pedestrians and motorcycles show a broadly similar pattern of distribution, with the highest concentrations in the city centre near Mahei Boima Road. There are secondary concentration points to the north around Simbo town, the south east near the Eastern Police Barracks and in the west of the city around the intersection of Bo-Tiama Highway and Mattru Road. There are relatively high numbers of pedestrians seen along all major roads, however, indicating that walking is a common form of mobility.

In contrast, cars are relatively widespread, showing up across most of the city, but in much lower numbers than pedestrians or motorcycles. Higher concentrations are confined to the major roads, particularly the Bo-Tiama and Bo-Kenema Highways running through the south of the city, and in the city centre.

Figure 31: Data collection tool output for “person” object detection in Bo, Sierra Leone

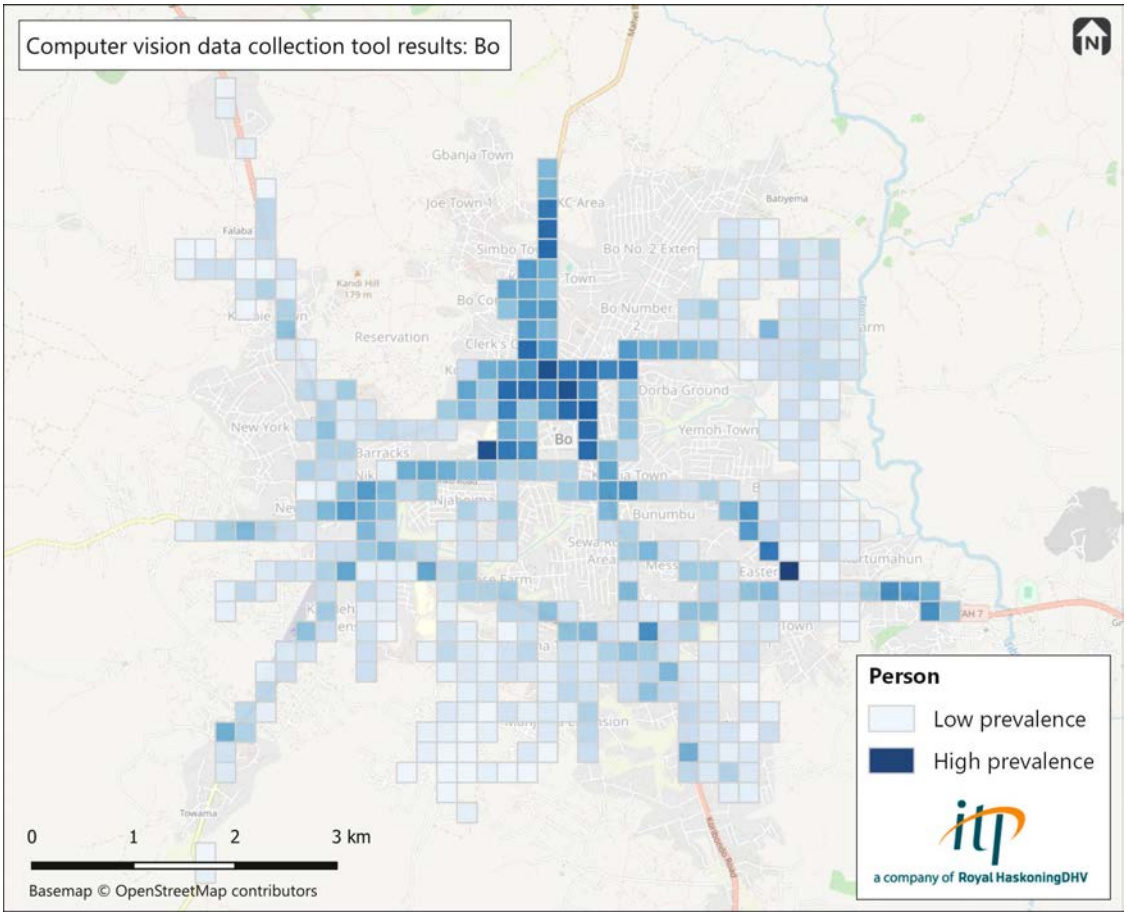


Figure 32: Data collection tool output for “motorcycle” object detection in Bo, Sierra Leone

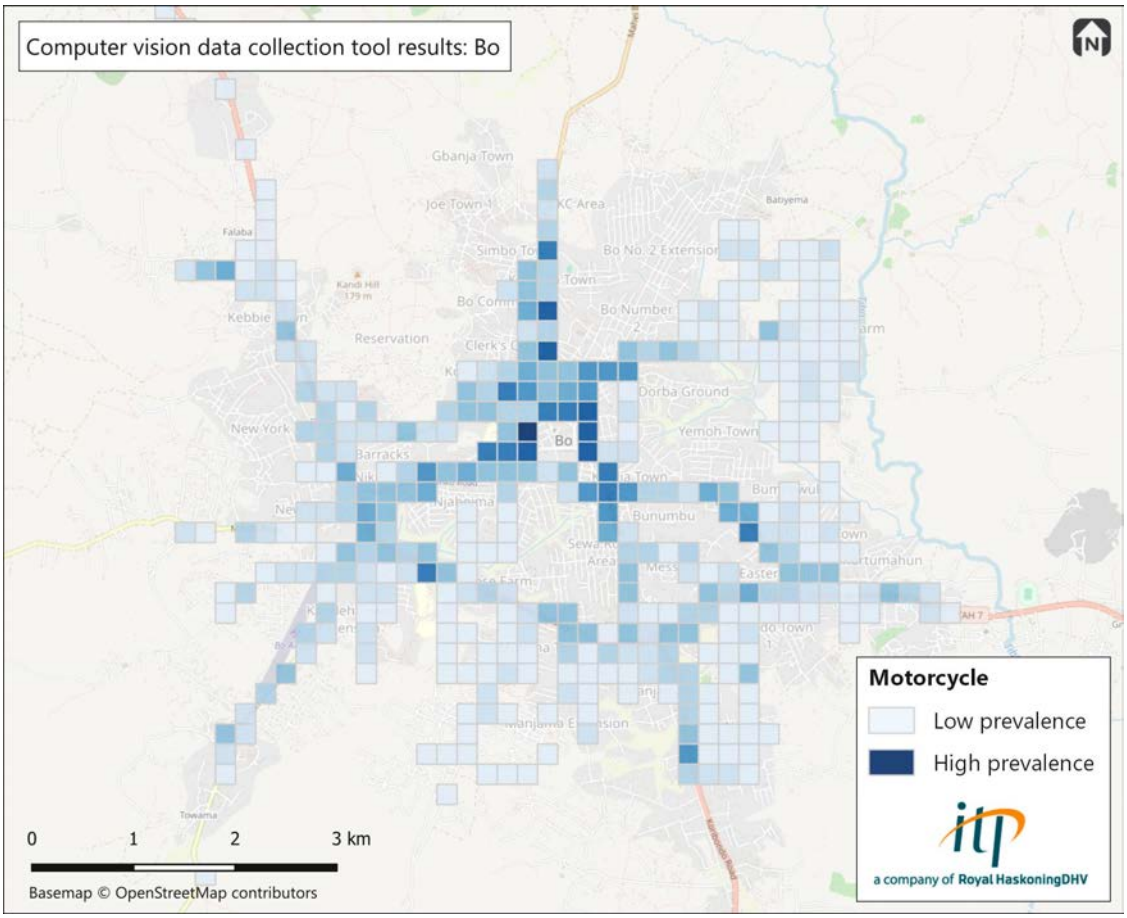


Figure 33: Data collection tool output for “car” object detection in Bo, Sierra Leone

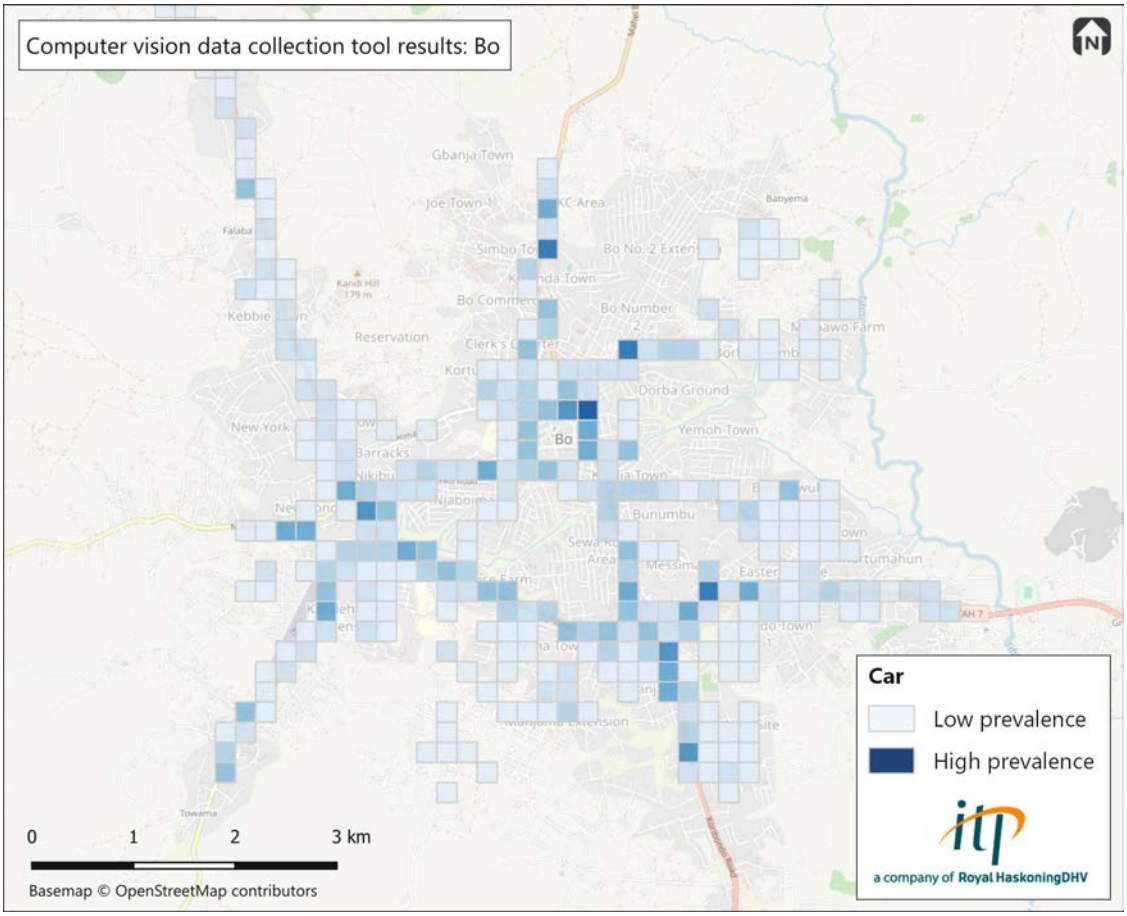


Figure 34: Data collection tool output for “three wheeler” object detection in Bo, Sierra Leone

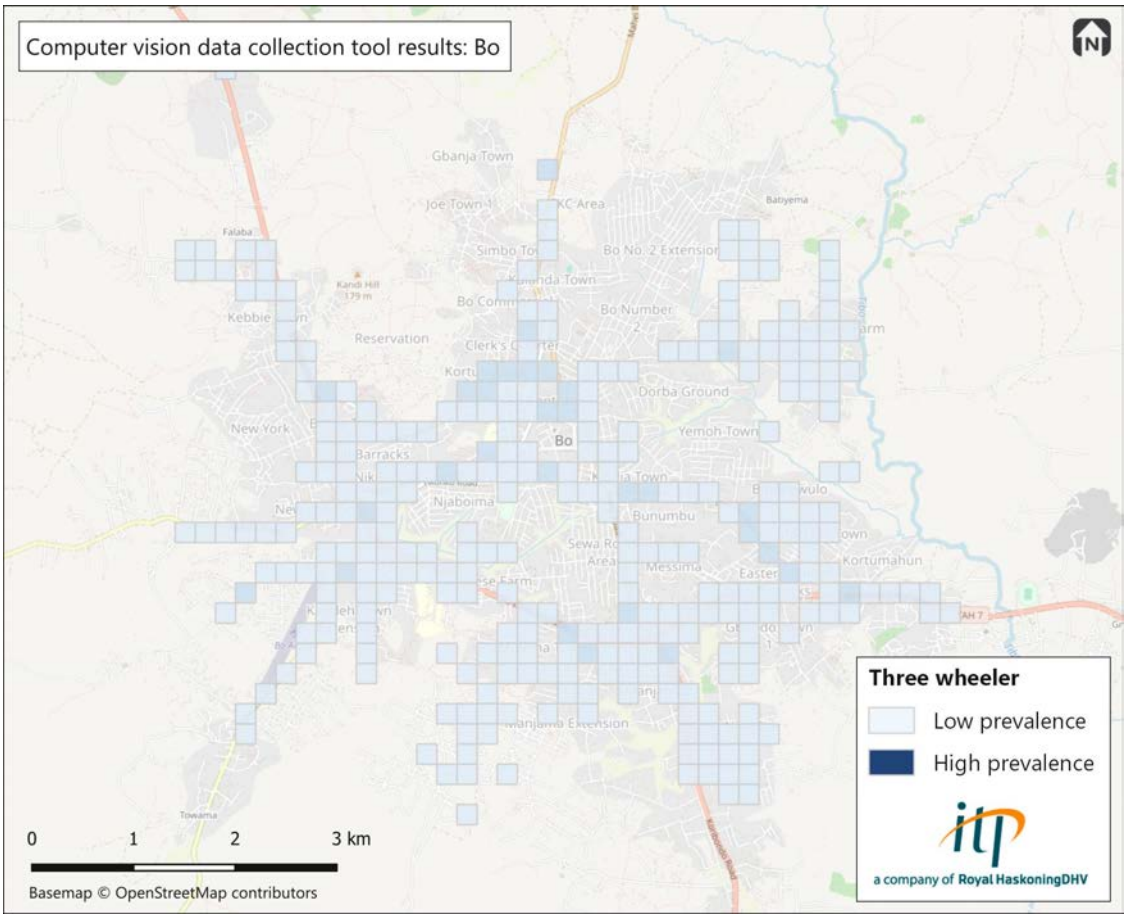


Figure 35: Data collection tool output for “bus” object detection in Bo, Sierra Leone

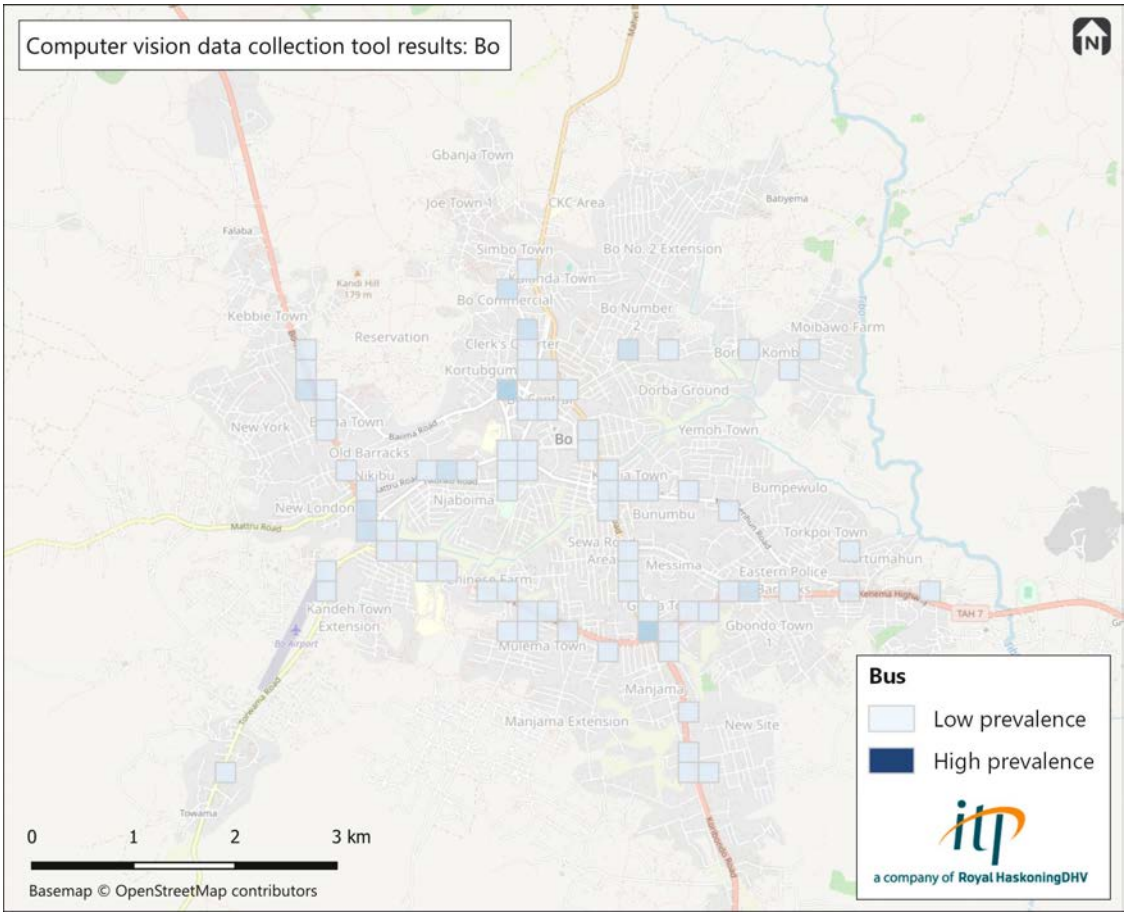
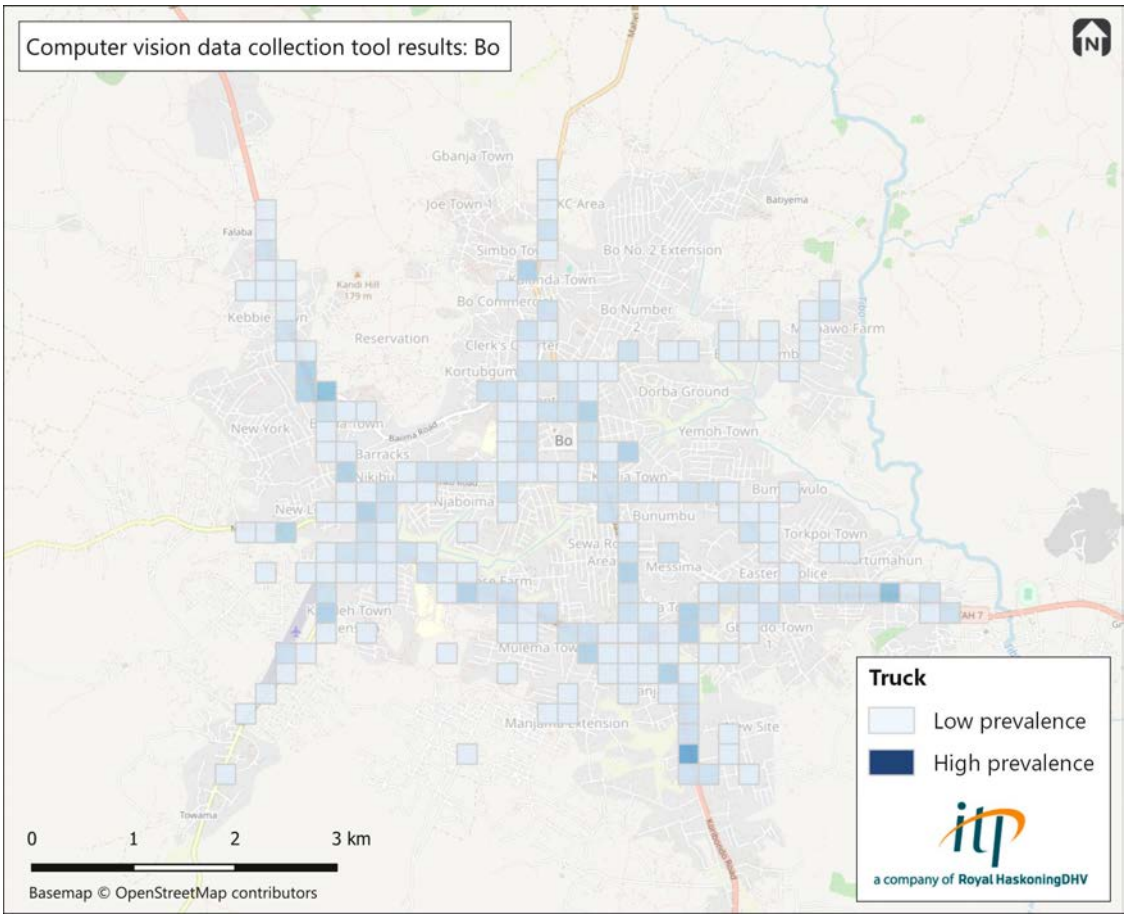


Figure 36: Data collection tool output for “truck” object detection in Bo, Sierra Leone



4.2 Dushanbe, Tajikistan

ITP has an ongoing project in Dushanbe, Tajikistan. Tajikistan is the poorest of the post-Soviet states and has a GDP per capita (PPP, current international dollars) of $5360, comparable to the Republic of the Congo ($5550) and sitting between some countries in Sub-Saharan Africa such as the Cameroon ($4660) and Nigeria ($6150).

As in other post-Soviet states, dashcam use is relatively common, which has resulted in relatively good availability of data on Mapillary. Coverage after filtering was good, as can be seen in Figure 7 and Figure 8, so no additional 'top up' data was required for the 29,472 Mapillary images.

Unlike Bo, Dushanbe's transport system does not use three wheelers or motorcycle taxis, and car ownership is higher. The main public transport modes are bus, trolleybus, marshrutka and car-based shared taxis. These shared taxis are extremely difficult to distinguish from private cars, with human reviewers unable to reliably tell them apart. We therefore do not split out shared taxis from private cars in our testing.

As can be seen in Figure 42, we used the GPS data extracted from the Mapillary dataset (as described in Section 3.7) to plot the network road speeds during the data collection trips. While the data inputs were derived using the computer vision data collection tool, the GPS observations were linked to the road network and aggregated using a separate tool developed by ITP.

Figure 37: Data collection tool output for "car" object detection in Dushanbe, Tajikistan

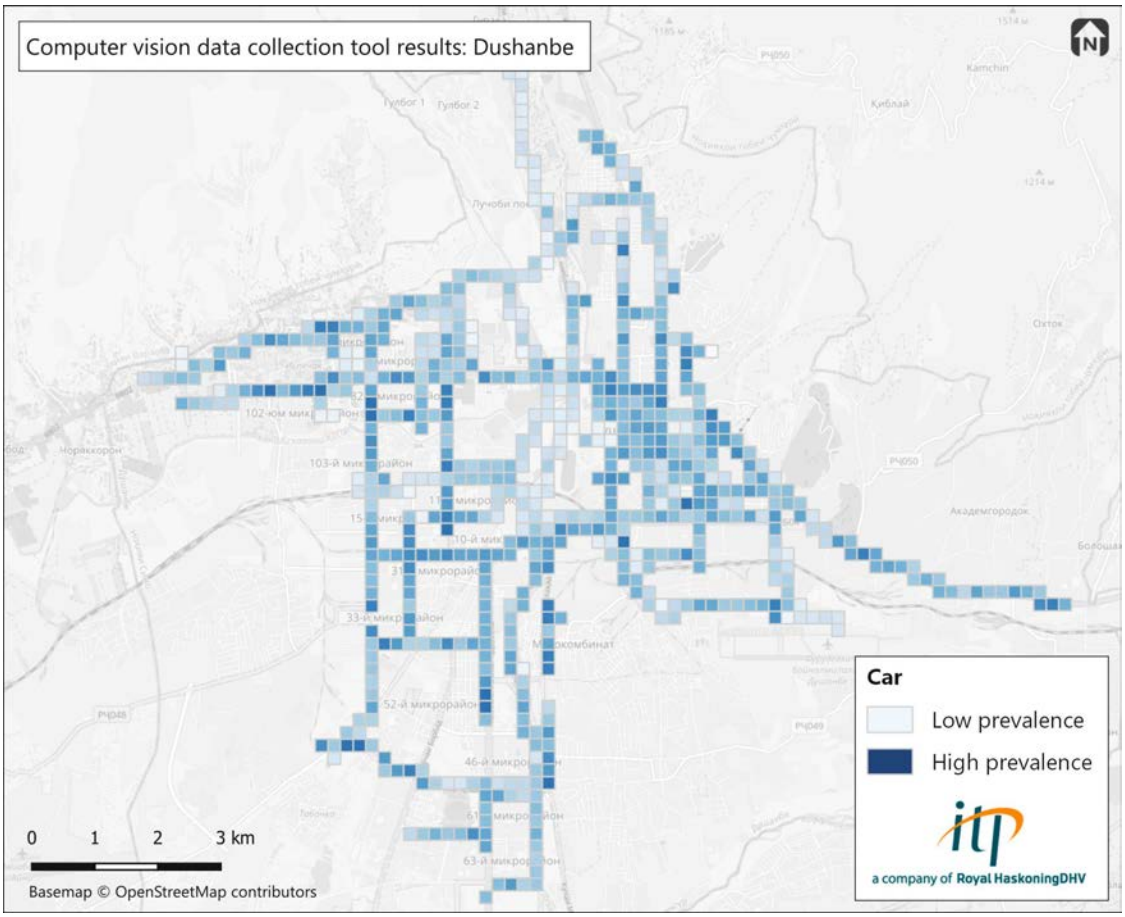


Figure 38: Data collection tool output for “person” object detection in Dushanbe, Tajikistan

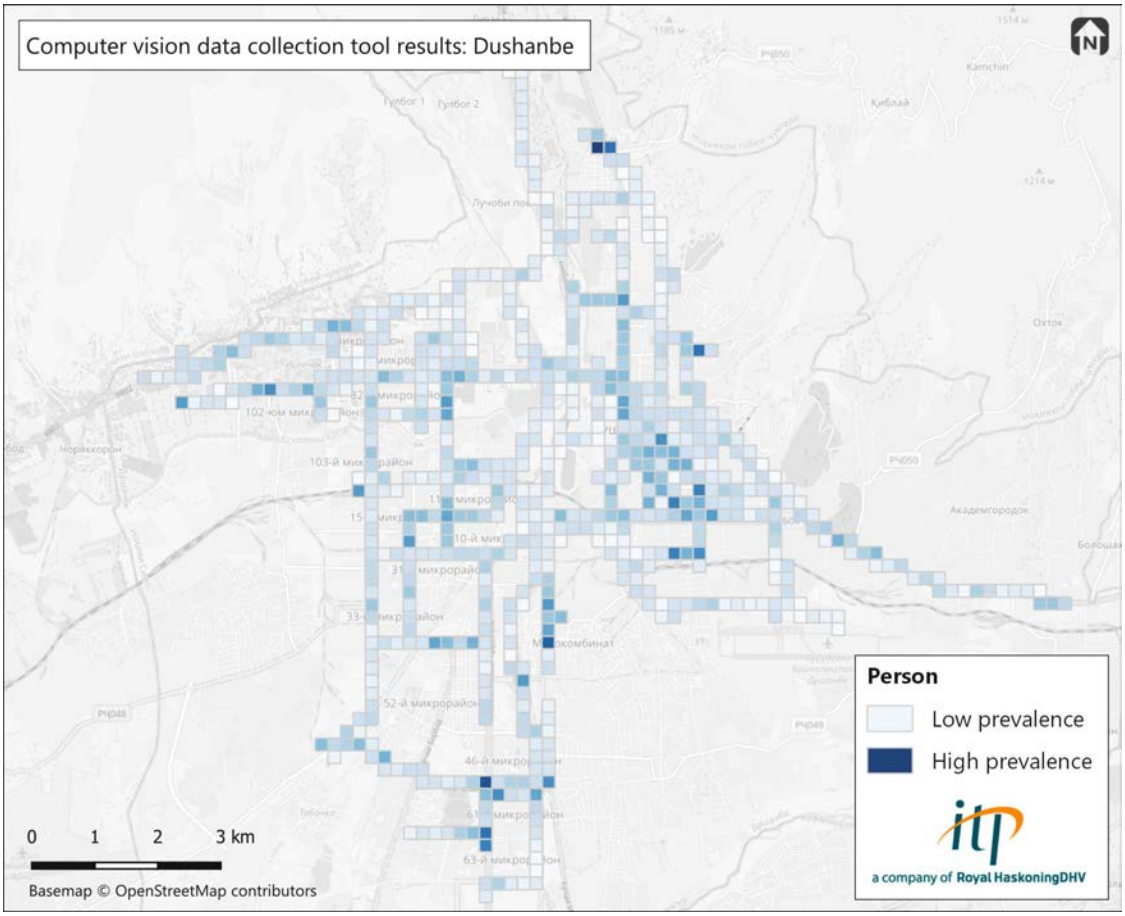


Figure 39: Data collection tool output for “bus” object detection in Dushanbe, Tajikistan

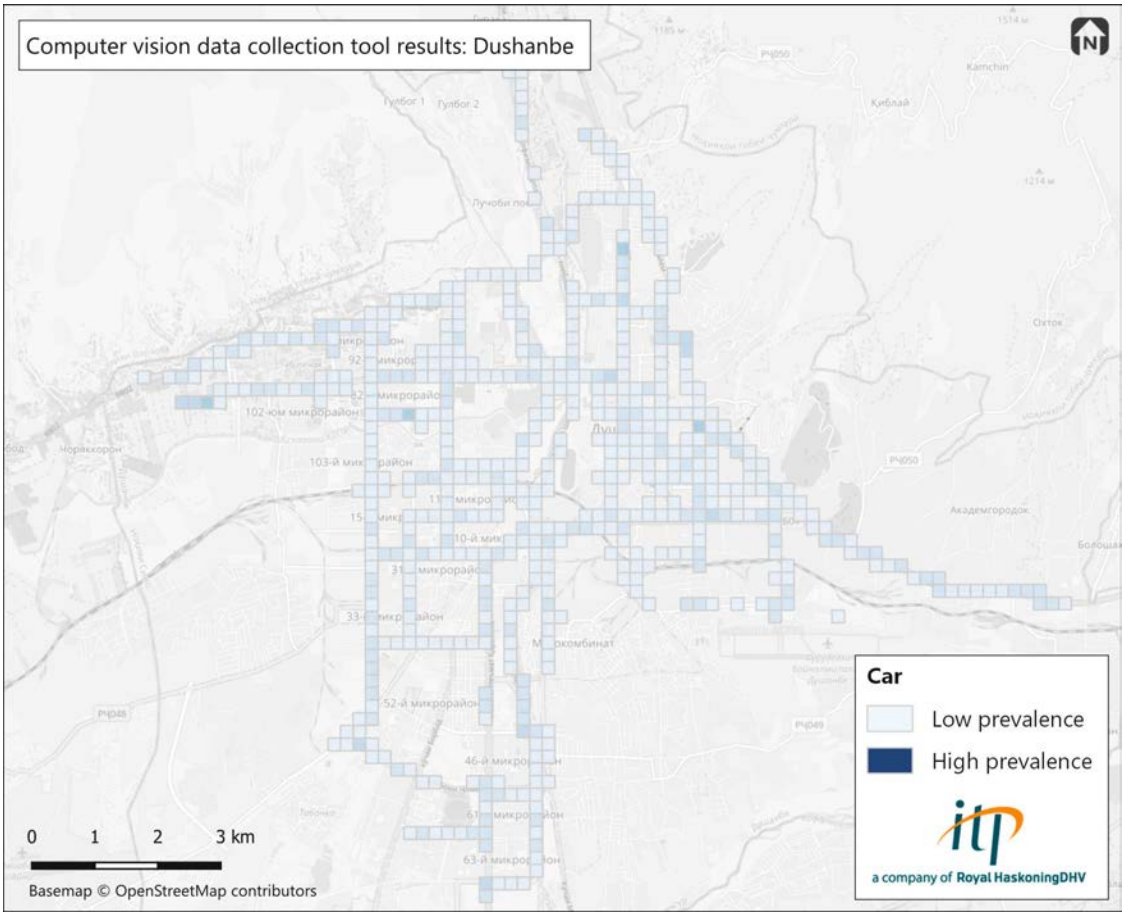


Figure 40: Data collection tool output for “van” object detection in Dushanbe, Tajikistan

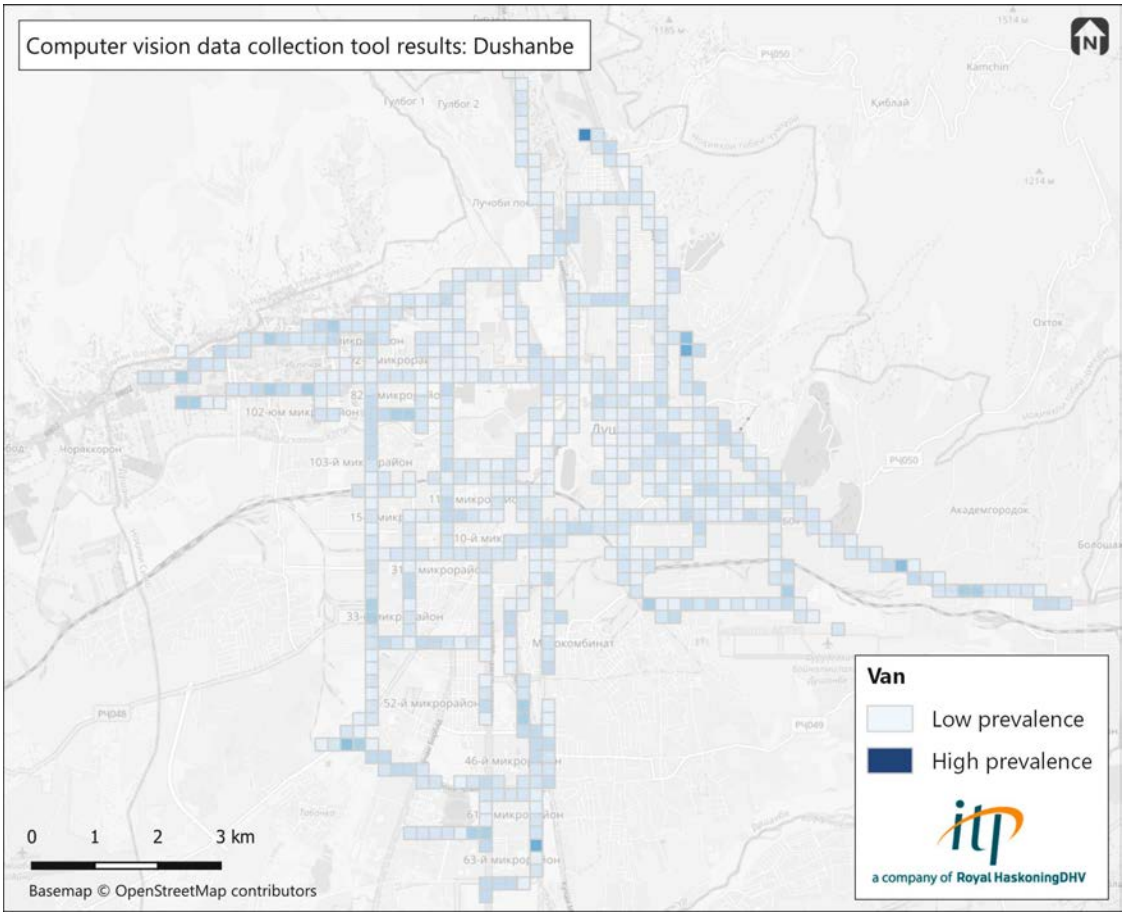


Figure 41: Data collection tool output for “truck” object detection in Dushanbe, Tajikistan

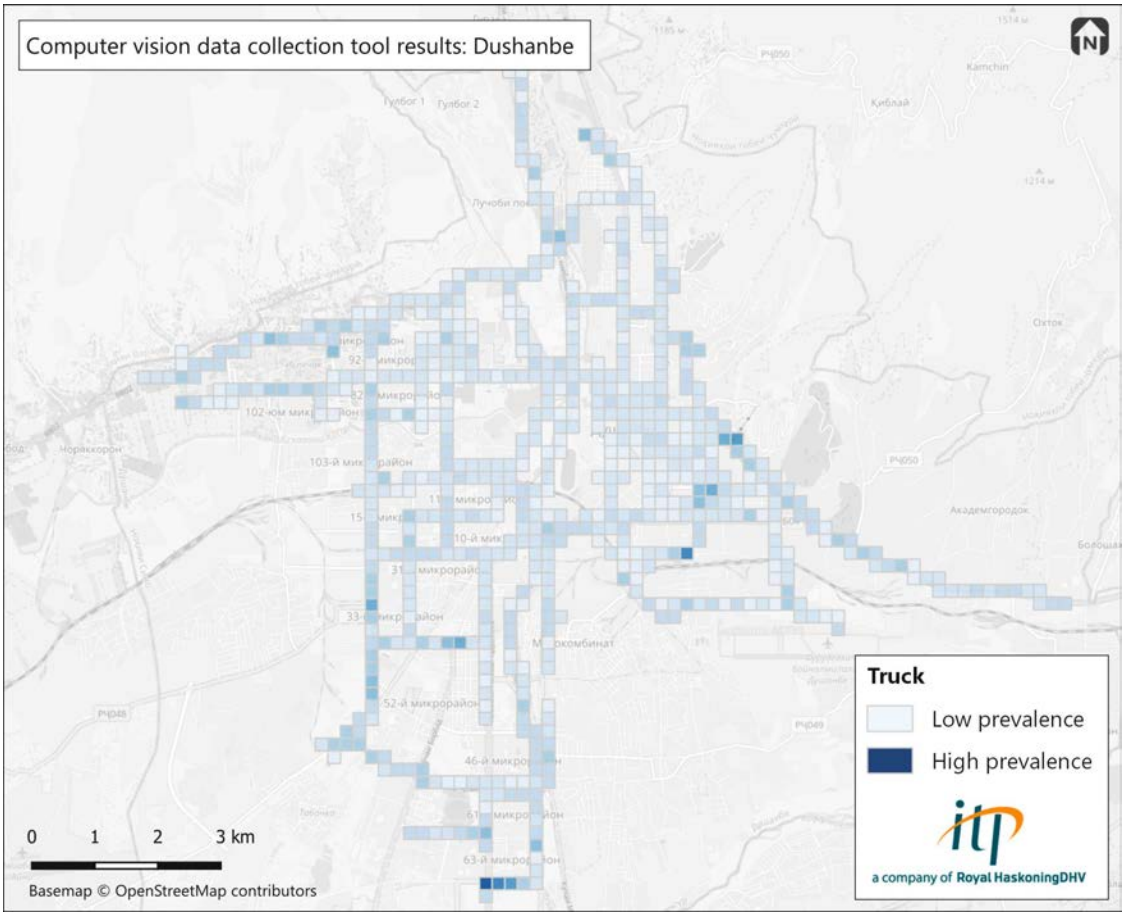
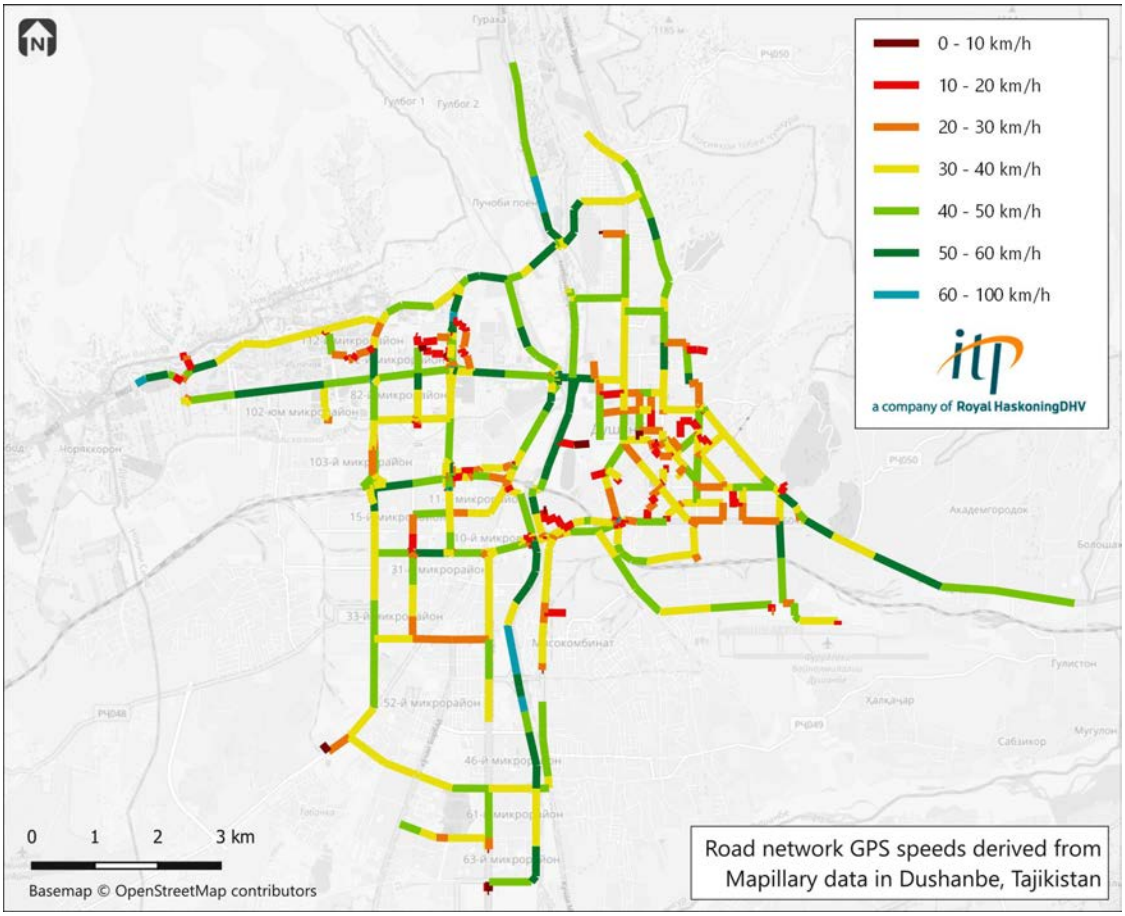


Figure 42: Road network GPS speeds derived from Mapillary imagery for Dushanbe, Tajikistan



4.3 Maputo, Mozambique

We tested the tool in Maputo as, like Dushanbe, ITP is participating in an ongoing project in the city. In addition, the Maputo transport agency, AMT, has collected and uploaded a large number of 360° images to Mapillary and may be interested in finding additional uses for them. The Grounded SAM object detector used during development appears to recognise objects in these images sufficiently well, however additional post-processing may need to be used to account for the greater field of view, as this may increase the counts obtained in these images. Please note that no correction has been applied for the results shown below.

Figure 43: Example of 360° Mapillary image collected by AMT in Maputo

Figure 44: Example of 360° image with detected objects

Figure 45: Data collection tool output for "person" object detection in Maputo, Mozambique

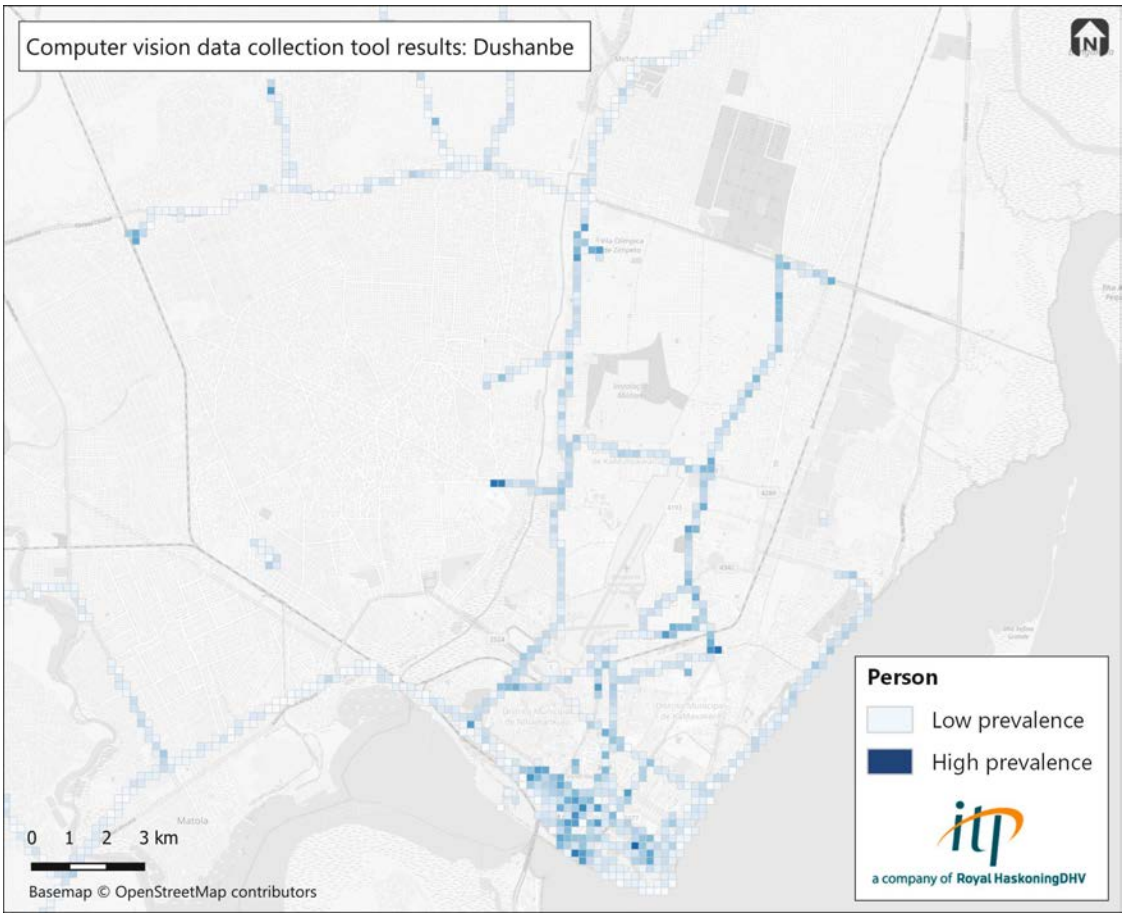


Figure 46: Data collection tool output for “car” object detection in Maputo, Mozambique

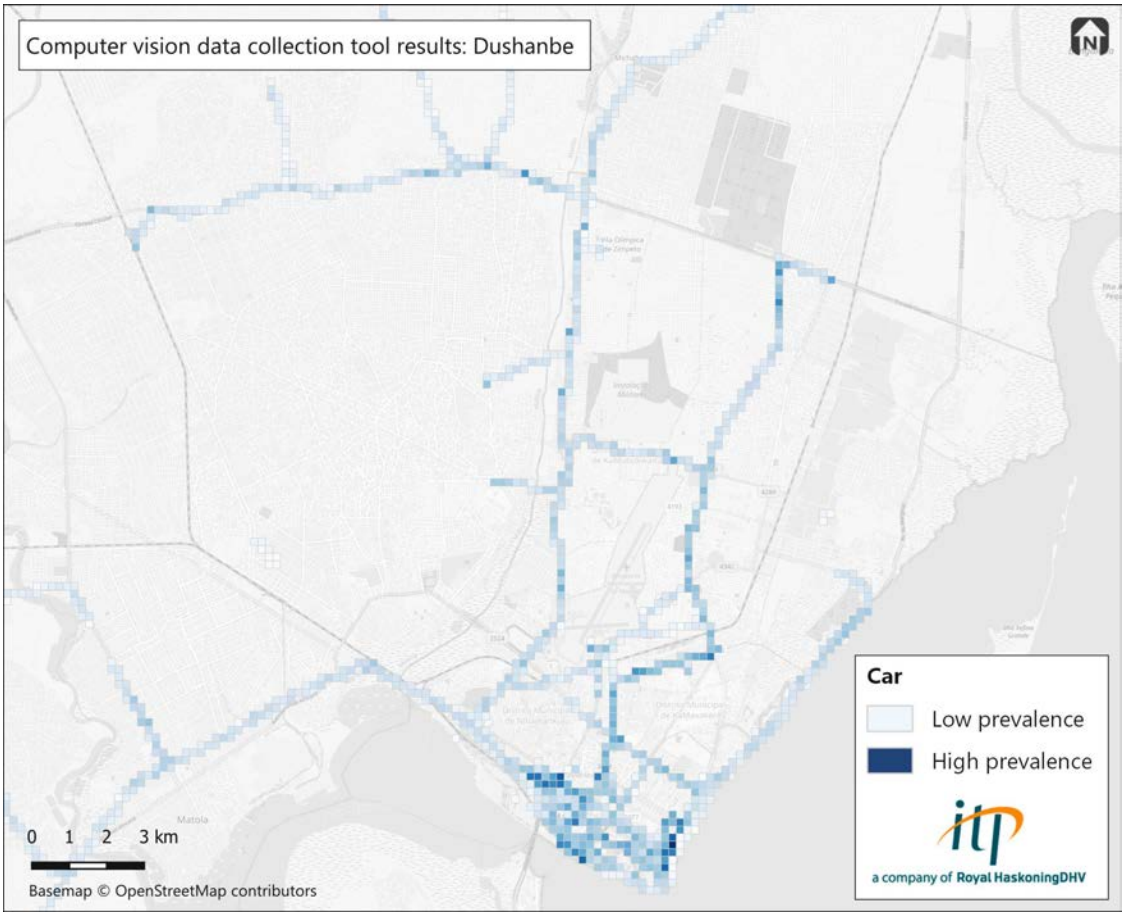


Figure 47: Data collection tool output for “bus” object detection in Maputo, Mozambique

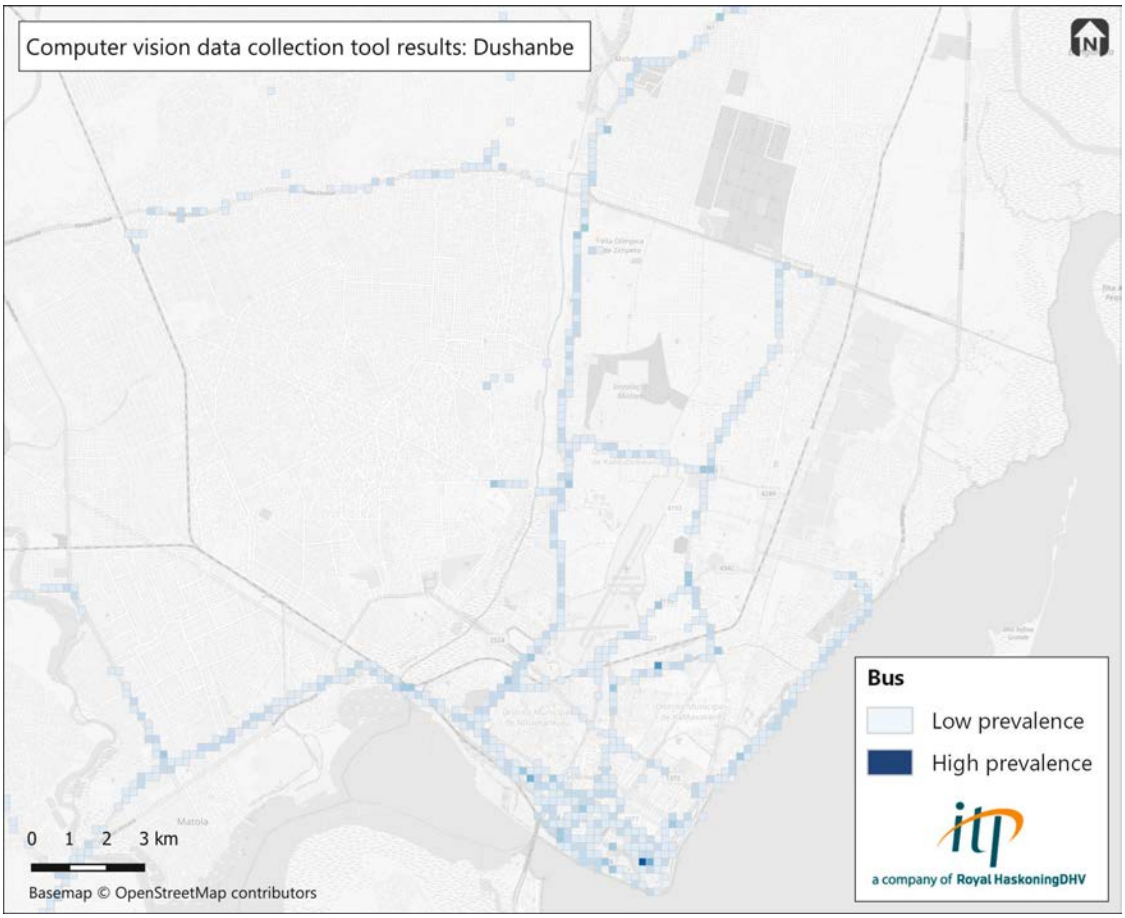


Figure 48: Data collection tool output for “truck” object detection in Maputo, Mozambique

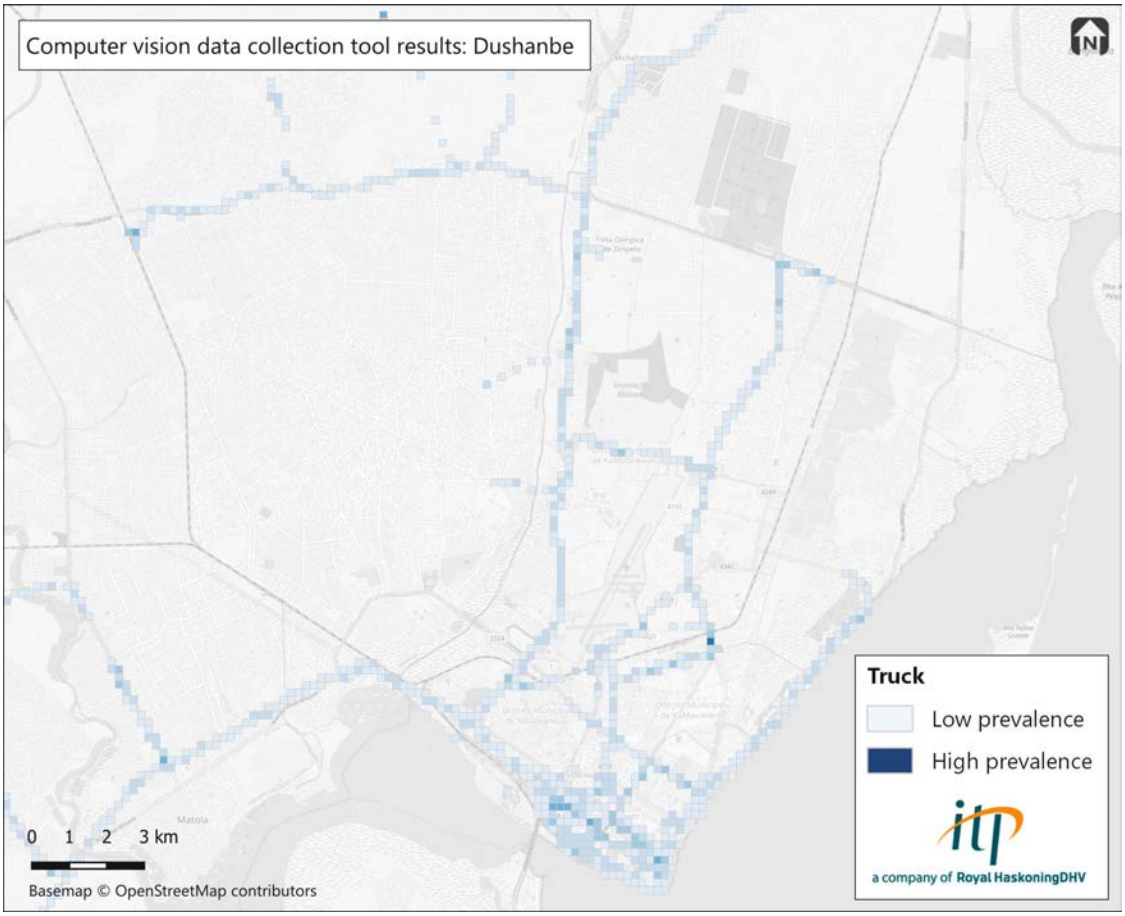


Figure 49: Data collection tool output for “van” object detection in Maputo, Mozambique

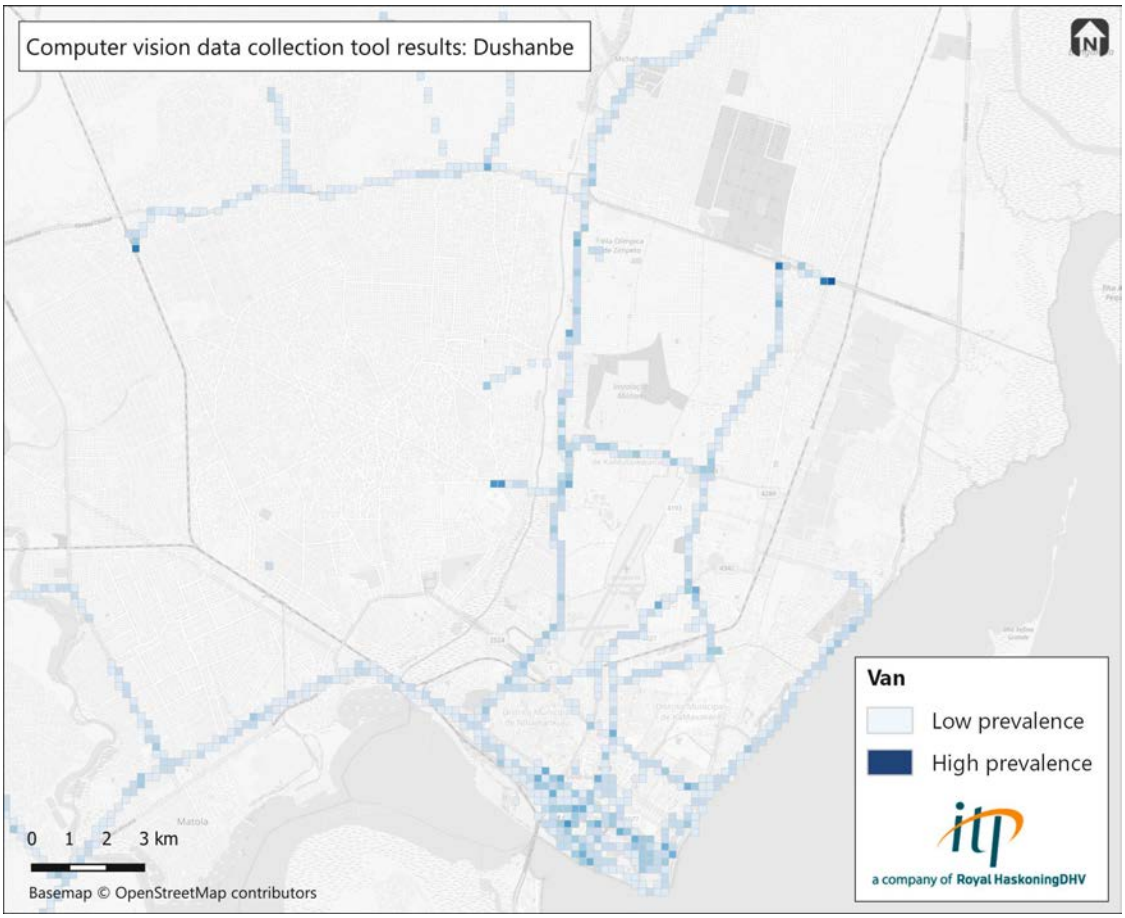
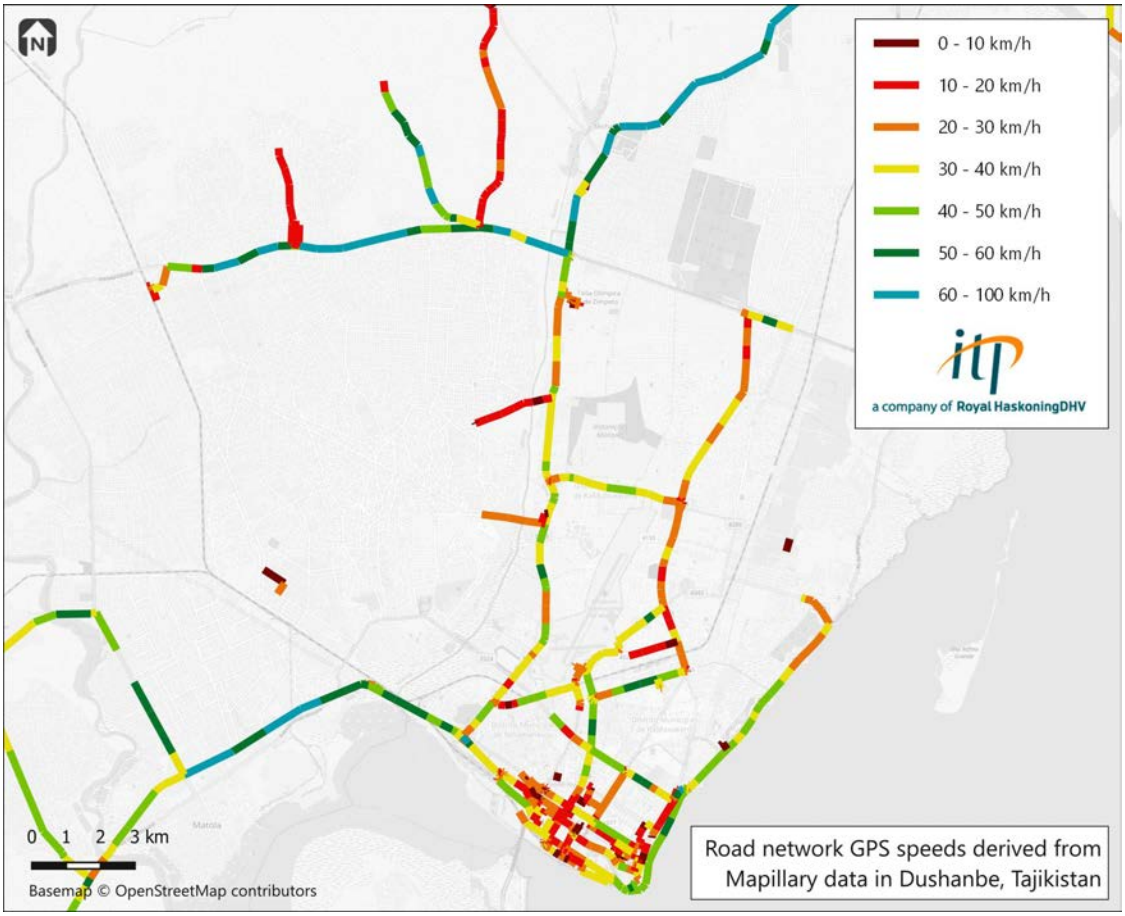


Figure 50: Road network GPS speeds derived from Mapillary imagery for Maputo, Mozambique

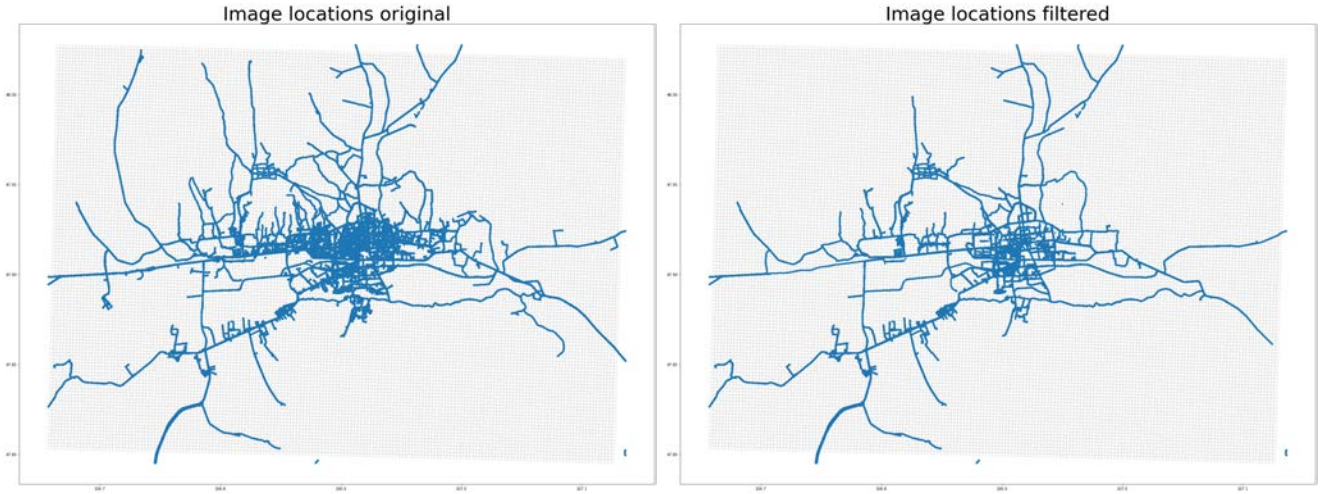


4.4 Ulaanbaatar

Ulaanbaatar has a very large amount of data available on Mapillary, with over 550,000 images present. Unfortunately the majority of these are collected on weekends, which typically exhibit different travel behaviour to weekdays. After removing weekends, 179,451 images remain, which is still an extremely large dataset and is currently unfeasible for processing with our development system (see section 3.4.6).

Removing weekends reduces the density of the spatial data coverage, however most key areas are still covered well, as seen in Figure 51.

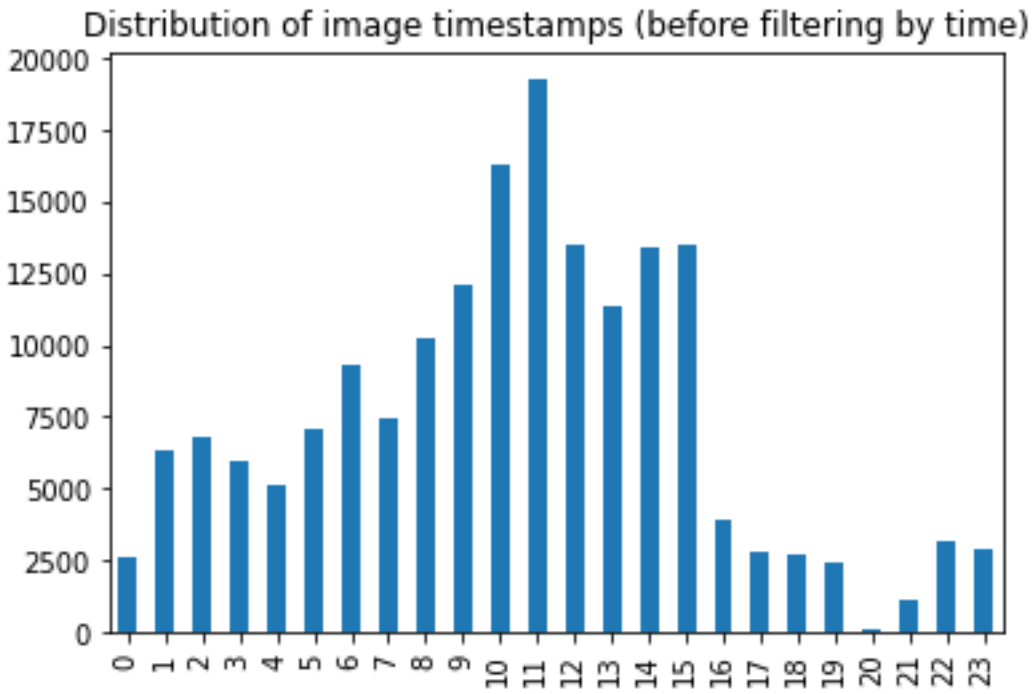
Figure 51: Coverage before and after filtering out weekend data in Ulaanbaatar, Mongolia



An unexpected issue with the Ulaanbaatar Mapillary data is in timestamp conversion. The profile of image collection times for the weekday sample shows a significant number of images collected at night, an unlikely survey time. Performing spot checks on a small number of images, we hypothesise that the

timestamps of these images have been modified in some way, resulting in inaccurate times when returned by the Mapillary API. For example, some contributors may have converted the timestamp from local time to UTC before upload and this process was repeated by the Mapillary processing system, resulting in an incorrect time offset.

Figure 52: Unusual profile of image capture time for weekday Mapillary data in Ulaanbaatar



5. Dissemination activities

This section describes the dissemination activities carried out over the course of our development work. Our two primary objectives in doing this were:

1. To showcase the capabilities of the tool to project partners and clients, demonstrating how it could be used to improve the quality of their work and reduce costs (or make data collection viable even on very small projects). To this end, we engaged with staff in government organisations in LMICs, international financial institutions and development funding bodies.
2. To promote computer vision technology as an option for low-cost data collection in LMICs, whether this uses our data collection tool or not. Greater availability of data in LMICs would significantly improve the quality of transport improvement schemes, and computer vision could enable long term data collection with low ongoing costs. In addition to the target audiences mentioned above, we also looked to engage with more academic transport practitioners to present to them a novel data collection methodology and learn from their experiences in similar work.

5.1 Direct engagement with project partners

Dushanbe, Tajikistan

We presented the data collection tool to members of the organisation who funded ITP's previous work in Dushanbe, Tajikistan, which was the project that inspired the development of this software, as data availability in the city was very poor at the time.

Key lesson: the client team asked about the reliability of the tool's outputs and how they could be validated, which is a common question to our presentations. This highlights one of the biggest blockers to roll-out of this tool in real transport projects – giving stakeholders the confidence that the results can be trusted and are fit for purpose, particularly if they disagree with that person's own understanding of an area. The next stage of development work should therefore focus on testing and benchmarking the results obtained from the tool against locations with traditional manual traffic counts, to provide evidence to reassure stakeholders.

Key lesson: Tajikistan appears to be particularly well-suited to the deployment of the data collection tool as the use of dashcams in cars is relatively common, meaning that there is a large potential data source that could be used for analysis. Future work in Dushanbe could therefore present a good opportunity to further develop the tool and create a track record of its use in real transport projects. Central Asia has two other LMICs, Kyrgyzstan and Uzbekistan, which could also be prioritised for initial roll-out.

Key lesson: the vehicle types seen in Dushanbe, and other countries in Central Asia, are all well-supported by the classes found in common computer vision training datasets, meaning that the limitations highlighted in section 3.4.5 are not as much of a hinderance as they would be in Africa, South Asia, or Southeast Asia. This further supports the case for further testing of the tool in Central Asia, during which workflows could be developed to train the computer vision model to detect more transport modes, such as three wheelers, which would make the tool more applicable for use in other regions.

Maputo, Mozambique

We contacted a previous project partner in the Maputo transport authority, Agência Metropolitana de Transportes (AMT), but unfortunately we were unable to arrange a live demonstration. We were able to share a presentation by email, however, and we received some feedback on the potential for the tool's deployment in Maputo.

As noted in section 4.3, AMT has collected a large number of 360° images along key transport corridors and a low-income neighbourhood in the city. They have found this to be a cheap method of data collection, and computer vision would allow for large amounts of intelligence to be generated without spending significant amounts of staff-hours processing it manually.

Our contact was very interested in the potential of our tool, and computer vision in general, for the evaluation of intervention programmes for both transport and urban planning. For example, if a road was paved and widened, the number of pedestrians, street traders and vehicles observed could be compared before and after the work was carried out to examine the impact it has on the social fabric of the area, as streets are often used as public spaces.

Key lesson: there are likely additional use-cases for the data collection tool beyond collecting data for city transport strategies, such as the evaluation of transport intervention schemes.

Key lesson: use of the tool in this way would likely require training of the object detector to recognise more object types, such as umbrellas (to help identify street traders) and puddles (to characterise the road surface quality). This reinforces the already identified need to develop a computer vision training workflow.

Other project partners

We attempted to arrange meetings with contacts from previous projects in Kenya and Sierra Leone but unfortunately we were unsuccessful due to the time constraints of our partners. We provided digital copies of the presentation to share some information but we were unable to discuss this in detail over email.

5.2 Inclusion of Data Collection Tool in transport planning training course for practitioners in Southeast Asia

We presented the data collection tool to a global organisation promoting the development sustainable transport, with the intention of including our data collection tool as a case study in a transport data training course currently under development. This course will be targeted at staff in transport authorities in Southeast Asia, so the inclusion of our case study would help us present novel methods of using computer vision technology to an audience of transport professionals in a mix of LMICs and UMICs.

We note that our contact asked about the data collection tool after seeing our article on the ITP website (see section 5.4 below), highlighting the success of this form of dissemination.

Key lesson: promoting novel uses of computer vision for transport data collection to organisations offering training courses would be a good approach for the wider dissemination of the technology. It would be desirable for the use of computer vision to be widespread among the transport community, not limited only to specialist businesses who could charge a premium for its use. Currently, state of the art computer vision software is freely available with no licence fees[20], making its use more affordable for students, academics, small businesses and governmental organisations.

5.3 Conferences

5.3.1 AGILE 2024 conference poster

Our poster submission to the AGILE 2024 conference[21] was accepted and we will present it in person at the event in Glasgow, Scotland, in June 2024.

This conference will provide a good opportunity to directly engage with transport data scientists in both academia, government and industry. In particular we will aim to highlight the utility of crowdsourced data in LMICs by focusing on how we can minimise the costs of our data collection tool by using imagery from Mapillary. Our poster describing our tool's methodology will be made freely available through the AGILE: GIScience Series[22], enabling other researchers to learn from and build on our work.

5.3.2 HVT pre-conference event at African Transport Research Conference 2024

We produced a 13-minute presentation for the HVT programme's pre-conference event at ATRC-2024, which was attended by a mix of academics, practitioners and researchers. This event allowed us to present directly to transport professionals from a number of LMICs in Africa, detailing how our computer vision tool works and providing some examples of the results and how they can be used. We finished the presentation with a short discussion on computer vision in general and how, once the system has been set up, it can allow for continuous data collection with low ongoing costs.

[20] For example, YOLO models released by Ultralytics: https://github.com/ultralytics/ultralytics
[21] https://agile-gi.eu/conference-2024
[22] https://www.agile-giscience-series.net/

5.4 Article on ITP website

We have published an article on our website[23] describing our data collection tool, how it can be used in the types of work ITP typically performs, and providing some example outputs for Bo, Sierra Leone. This article was cross-promoted on Twitter and LinkedIn, with the LinkedIn post receiving 730 impressions, 35 engagements and 18 clicks-throughs to the article. The article itself on the ITP website was viewed by 47 users.

6. Conclusions

In this T-TRIID project we have developed a prototype data collection tool to provide data that allows the characterisation of a city's transport system, using computer vision technology with street-level imagery. We have demonstrated the tool's operation in three cities: Bo (Sierra Leone), Dushanbe (Tajikistan) and Maputo (Mozambique), and produced heatmaps of the relative prevalence of people, cars, motorcycles, three wheelers, buses and trucks across each city.

The core functionality of the tool has implemented in a series of Python scripts, however technical improvements to some areas are required before it is ready for full deployment in real projects. As noted in the deployment plan (see Appendix A:), the key improvements required relate to the object detector, both in the reliability of detection and speed. Possible solutions are training the model with tagged images of three wheelers to improve detection and reduce false positives, or changing the object detector to a different piece of software. We plan to continue development through incremental improvement during ITPs consultancy work, with the aim of this technology being a commonly used component of our data collection toolkit.

We do not intend for this tool to replace detailed traffic counts, but instead provide higher-level data for the entire study area, suitable for transport consultancy projects such as transport strategy development. Such projects often have relatively small budgets with little capacity for city-wide data collection, so a key component for successful deployment is low cost of operation. In its current form the tool can produce outputs with only a few hours of active staff time and, though using crowdsourced data, no monetary expenses.

In real-world deployment, however, a more effective solution may be to use free crowdsourced data as much as possible, with targeted dashcam surveys used to 'top up' any spatial or temporal gaps. Although this would incur larger costs than the all-crowdsourced use case, it should still prove cheaper whole-city data than traditional staffed surveys.

During development we identified an additional output that can be produced using crowdsourced data: road network travel speeds (see Section 3.7). This extra data can be extremely useful to transport strategy development and the analysis public transport, so simple acquisition of this information significantly increases the value offered by the tool.

6.1 Training object detector for better performance

For the tool to be most effective in LMICs, it is necessary to be able to detect relevant vehicle types reliably and accurately. The most important vehicle types where detection performance is poor are three wheelers and truck-based modes such as poda poda and tro tro. These modes are not included in common datasets used to train computer vision models, so it will likely be necessary to train our own model.

The most important requirement for this is the creation of a suitable training dataset of images of the required transport modes. Each image must be tagged to specify the class of each relevant object type that is present, including its position and extent within the image.

The trained model's ability to detect and distinguish objects is based on its ability to recognise patterns in the pixels of an image, so the level of variety in the training dataset is a critical factor in its real-world performance. It is very important that the training dataset provides a wide range of viewing angles, distances, lighting conditions and backgrounds for each object class. Ideally the training data would be taken from dashcam-style imagery, as this is most similar to the imagery used by the data collection tool.

[23] https://www.itpworld.net/news-and-views/2024/found-footage-making-the-most-of-pre-existing-data-with-computer-vision

The exact size of the training dataset is difficult to estimate as the detection performance of the trained model is dependent on a wide range of factors, but we expect good results to require at least a few hundred instances of each object class.

The creation of training datasets can be labour-intensive, which is the main blocker to this exercise. We do not anticipate sourcing suitable imagery to be difficult, as there are many sources freely available on the internet, but it is likely that tagging would have to be done manually.

One possible workaround could be to use a pre-existing training dataset, and we have already identified a few examples from Bangladesh[24, 25, 26] and India[27]. Pre-existing datasets should be used with care, however, ensuring that they are of suitable quality and technical compatibility, and that they are released under a licence that permits its use for this purpose.

Another potential hurdle may be in computer hardware limitations, as training a model requires more resources than using that model to detect objects. If this is found to be an issue, cloud computing services may be a cost-effective option of accessing more powerful hardware.

6.2 Wider application of computer vision for low-cost data collection

Computer vision holds great potential for transport data collection. In addition to the application we have described in this report, there are several other tools offered by UK companies, such as Vivacity[28] (traffic monitoring, traffic signal control), Tracsis[29] (traffic surveys) and VisionTrack[30] (video telematics, road safety) which can use similar data for different purposes.

Transport organisations around the world routinely record large amounts of video, such as CCTV security cameras on buildings and in public transport vehicles and utilising these vast data sources to make transport improvements more effective can be a powerful tool for helping people travel safely, affordably and efficiently.

Although use of computer vision is not yet commonplace in all LMICs, improving awareness of its capabilities among local transport professionals, particularly how it can make use of pre-existing data for very cost-effective data collection, would enable its benefits to be more quickly realised once it is available.

[24] https://doi.org/10.1016/j.dib.2020.106465
[25] https://doi.org/10.48550/arXiv.2401.10659
[26] https://doi.org/10.17632/7xvcvxgphb.1
[27] IDD: India Driving Dataset, https://idd.insaan.iiit.ac.in/dataset/details/
[28] https://vivacitylabs.com
[29] https://tracsistraffic.com
[30] https://www.visiontrack.com

APPENDIX A: DEPLOYMENT PLAN

The purpose of this deployment plan is to set out the considerations that must be taken into account for successful roll out of the data collection tool. Here we define successful roll out as the tool being used on two of ITP's live consultancy projects by the end of 2024.

Purpose and goals of the Computer Vision Data Collection Tool

Short Term

We will use the innovative nature of our tool in its current state to showcase our capabilities and start conversations with our regular clients and collaborators. As we are still learning how to utilise the tool and extract maximum value from the outputs, we will consider these early demonstrations an extension of the development process. This incremental product development strategy has proven successful for ITP in the past, such as with our TransitWand public transport survey tool. It allows us to utilise core functionality while developing more advanced components, helping funding the tool's development without external investment.

Long Term

The tool would be a standard part of our consultancy offering and would be a common inception-stage task for when working in a new city. This use-case is based on the current state of the tool and our (currently limited) experience using it. This role may change or evolve as we gain more practical experience in its use.

The cost of producing heatmaps for multiple modes should be no more than a few hundred GBP, which would be primarily used for sourcing the imagery from pre-existing sources (e.g. Mapillary or Google Maps). Active staff time should be in the order of 1 hour, and we should have the full set of results calculated in no more than 2 days from start (this accounts for the computer processing time).

Whilst the tool will at first be offered as a consultancy product/service, the ability to produce realistic transport data in a rapid timeframe will be of significant benefit to our clients in LMICs. ITP will also continuously monitor the best way for LMICs to benefit from this technology.

Description of product to be deployed

In the near future the "computer vision data collection tool" will be a 'workflow' implemented using a collection of Python scripts, rather than a unified piece of software with a user interface. This is acceptable at this stage of development as it will only be used by our current team members, who have an intimate understanding of its workings, and it will likely be incrementally improved with each project it is deployed on.

In the medium term (around six months to one year from now) we should refactor and consolidate the current scripts to make the tool easier to run and maintain. In the long term (1-2 years from now) we should look to develop a simple user interface, allowing self-service use by non-expert ITP staff.

Software requirements

The environment we used for development consisted of the following packages:

For image downloading and processing scripts

Python 3.11.5, Pandas 1.5.2, Geopandas 0.14.0, Numpy 1.24.1, Timezonefinder 6.2.0, Shapely 2.0.1, Mercantile 1.2.1, Matplotlib 3.6.3, Contextily 1.3.0, vt2geojson 0.2.1, gpxpy 1.5.0, opencv-python 4.8.0.76,

For object detector

During development we used *Grounded-Segment-Anything*[31] for object detection and segmentation. This framework is a combination of the *Grounding DINO*[32] object detector and the *segment-anything*[33] image segmentation model, allowing the user to detect objects in an image using text inputs.

This software was run on a computer running Ubuntu 20.04, with Grounded-Segment-Anything installed using the Docker image provided by the authors via the official GitHub repository.

[31] https://github.com/IDEA-Research/Grounded-Segment-Anything
[32] https://github.com/IDEA-Research/GroundingDINO, see Liu et al. (2023)
[33] https://github.com/facebookresearch/segment-anything, see Kirillov et al. (2023)

Use cases and competitors

With the tool's current outputs and capabilities, we can collect basic data that gives us an estimate of the prevalence of each transport mode across the study area. Although the tool returns a numeric count of the objects it detects, these should not be used as actual counts due to image sampling considerations and aggregation methods used in our processing. Instead, a high numeric value should be interpreted as meaning "more prevalent", and a low value as "less prevalent".

Improved object detection would improve the accuracy and reliability of the results, but ultimately it is unlikely that the tool will deviate significantly from this type of output. This is due to the limitations of the input data we use – vehicle-mounted cameras operated by hobbyists in their own time, with motivations that may not align with our intentions for the data. As the camera is moving, specific locations are unlikely to be surveyed frequently and, when they are surveyed, it may be under vastly different conditions (e.g. time of day, day of the week, weather conditions). We therefore intend to focus on high-level intelligence produced for a wide area at a low cost, rather than detailed analysis of specific locations.

This stands in contrast to a similar application of the same object detection technology: computer-vision powered traffic counting, which produces a detailed, time-disaggregated count of different transport modes for a specific location (or set of locations). There is already an established commercial market offering these services in HICs, for example Vivacity Labs and Tracsis in the UK or many other companies worldwide. Roll-out in LMICs is more limited, although we expect there will be swift adoption once suitable commercial conditions are in place.

While these services may appear similar, or superior, to the capabilities of our tool, computer-vision powered traffic counts differ in two key areas:

1. Geographic scope: our tool aims to produce outputs across the whole study area, rather than specific count sites. Detailed counts could be produced for a whole city, but with a greatly increased cost:
2. Cost: our tool is intended to be very low cost, in the order of a few hundred British Pounds, mostly in the form of staff-time spent operating the tool. In contrast, commissioning a survey company to perform counts on all major roads in a city would cost far more, from tens of thousands to hundreds of thousands of pounds[34].

Ultimately our tool's unique selling point is the use of pre-existing data, either crowdsourced or available commercially[35], to make transport data available for smaller-budget consultancy projects, which would otherwise lack the resources to commission surveys.

Anticipated use in ITPs projects in LMICs

Although our data collection tool's outputs are not well suited to detailed technical design work, ITP is a transport planning consultancy, not an engineering consultancy. The work we do in LMICs is often high-level or early-stage, where such fine detail isn't required.

We therefore see ourselves deploying the data collection tool in two main use cases:

1. When working in a new location, to help our team gain an understanding of the transport situation in each part of the study area. For example, neighbourhood A may have few cars but lots of motorcycles, whereas neighbourhood B may have high car use and limited active travel.
2. When working in a location with poor data availability. This can be the case where either the data has not been collected, for example where local government does not have the capacity to manage their transport system and collect the necessary data, or where we are unable to source existing information from our project partners, which does happen from time to time when the local client is not well-engaged with the project.

Example project uses could be the division of a city into zones classified by primary modes of travel, the identification of priority areas for the introduction of pedestrian infrastructure, or identifying the areas that are well-served or not-well served by public transport.

[34] Based on grant award information published by Nottingham City Council for the use of Vivacity cameras to cover an approximately 3 km radius in 2019. The estimated cost for installation and use of 11 Vivacity cameras with computer vision was £13,000 for Phase 1 (three months of operation). Overall project budget for a further six months, with the addition of ANPR cameras and "floating" travel time data from Google, was estimated at £96,000.

[35] Google charges $0.007 USD per image obtained through their Static Street View API. Purchasing access to 100,000 images to cover a city with our tool would therefore cost $700 USD.

Geographic target areas

The data collection tool can be used in any country around the world, as long as street-level imagery is available. We have developed the tool using Sierra Leone, an LMIC, as our test case to align with ITP's intended use for the tool in our international consultancy work, which is almost exclusively focused on LMICs. These countries often have poor data availability and so projects in these locations would benefit from additional data sources – any data is better than no data.

There is no technical reason that we could not use the tool in HICs as well. Data availability in HICs is typically good, including the availability of street-level imagery. This improves the quality of the outputs of our tool, but also makes them less useful as it is more likely to be alternative data sources providing similar information (for example, the permanent traffic count data published by the UK Department for Transport[36]).

The main technical consideration for deploying our tool in different countries is the performance of our object detection model for common transport modes. Modes that are unique to a particular country (for example, Jeepneys in the Philippines) may be difficult to reliably distinguish from similar modes if that mode is not commonly seen in the datasets used to train the object detection model. Alternatively, some common modes may be inherently difficult to distinguish from privately-operated versions of the same vehicle – for example shared taxis in Sierra Leone, which do not carry markings and so are difficult to distinguish from private cars.

Risks

Key risks to the successful use of the data collection tool in ITP's project work are:

1. ITP's commercial priorities do not allow opportunities for the deployment and development of the tool. Suitable projects may not be offered by our clients, or ITP may be unsuccessful in securing such work. In such a scenario, mitigation measures would be to look for additional external research and development funding sources.
2. Clients may not understand the tool's purpose and limitations, and may not be interested in its use because it does not provide the level of detail they think is required for transport planning work. Mitigation measures would be to produce explanatory marketing materials for the tool, including case studies of how it has been used in different locations or project contexts, and the value it brought to the work.
3. If demand for the tool is high, the project team members may not be able to keep up with requests for its use. Mitigation measures would be to train other ITP staff in the detailed use and troubleshooting of the tool, and the development of a user-friendly interface which would allow non-specialist staff to use the tool without assistance.

Accountability mapping

To ensure the maximum value is obtained from our work in this research project, we have assigned key roles and responsibilities to our team members for the deployment of the tool in ITP's project work.

Giles Lipscombe will be responsible for overall product development, technical development of the input and output data processing procedures, training and product support, and the promotion of its use in ITP projects.

Murşit Sezen will be responsible for the technical development of the object detection component of the model.

Mark Dimond will be responsible for the commercial strategy, to ensure that its capabilities align with and complement ITP's project order book, and with ITP's other data tools.

Next steps

To enable successful deployment of the tool in ITP projects, we have identified the following areas as requiring further work in the short term:

Internal promotion of the tool for ITP staff, particularly project managers. This would likely be in the form of demonstrations and webinars. The short-term aim of this task is to find projects where use of the tool could be offered to the client as an "added value" bonus, which would provide further testing and development opportunities.

- We will be presenting at ITP's "Automation for Consultancy" day in March 2024.

[36] https://roadtraffic.dft.gov.uk

Technical training for selected ITP staff to teach them how to use the data collection tool. The aim of this task is to expand the number of staff able to operate the tool, mitigating staffing risks to its deployment.

External marketing and dissemination for clients, project partners and academia. The aim of this task is to showcase our capabilities to potential project partners and encouraging them to consider how they might make use of the tool for their own work. We have identified target partners as:

- Project partners in Maputo (Mozambique), Dushanbe (Tajikistan) and Ulaanbaatar (Mongolia) through on-going ITP projects. Contact will be made through ITP project managers, and we will present ready-made outputs for each city using Mapillary data. This has presented a challenge in that Ulaanbaatar, in particular, has a very large number of images available through Mapillary (over 500,000) and our computers can process one image every 8 seconds.
- Sierra Leone Ministry of Transport and Aviation, through our contacts from our recent work in Freetown.
- We will submit a poster to the AGILE 2024 conference[37]. The conference will take place in June 2024, with the submission deadline on 10th February.
- ITP Director, Jon Parker, will be presenting on Artificial Intelligence in transport at the University of the West of England (UWE) in his role as Chair of the CIHT (Chartered Institution of Highways & Transportation) AI Task and Finish Group.

Priority technical improvements

1. Improve object detection performance for three wheelers (auto rickshaws) and heavy trucks.
2. Improve object detection speed. Currently we use the CPU (central processing unit) of a laptop, which can process one image in approximately 8 seconds. While this is acceptable for smaller cities or areas with less data, a large city with good data, such as Ulaanbaatar in Mongolia has over 550,000 images available on Mapillary – which would require more than 50 days of processing. One solution would be to trial using the laptop's GPU (graphics processing unit) with the same object detection model, which may allow for significantly quicker processing. An alternative could also be to test a different object detection model, although this would require more testing to ensure the detection is sufficiently reliable and it is able to detect all required object types.
3. Refactor Python scripts to streamline and improve the ease of maintenance. Although it will not directly affect the capabilities and outputs of the model, it would simply the modification of the tool during any early live-project deployments.

[37] https://agile-gi.eu/conference-2024/call-for-papers-2024

HaskoningDHV UK Ltd., trading as Integrated Transport Planning
1st Floor,
1 Broadway,
Nottingham,
NG1 1PR,
UK
Tel: +44 1733 334455
Email: giles.lipscombe@itp.rhdhv.com
Web: itpworld.net

HIGH VOLUME TRANSPORT
APPLIED RESEARCH

UK International Development
Partnership | Progress | Prosperity