ACCURATE VEHICLE SPEED ESTIMATION FROM MONOCULAR CAMERA FOOTAGE

D. Bell [1]*, W. Xiao [1], P. James [1]

[1] School of Engineering, Newcastle University, Newcastle Upon Tyne, UK – (d.bell5, wen.xiao, philip.james)@ncl.ac.uk

Commission II, WG II/5

KEY WORDS: Computer Vision, Traffic Monitoring, Vehicle Speed Estimation, Urban Analytics, Machine Learning

ABSTRACT:

A workflow is devised in this paper by which vehicle speeds are estimated semi-automatically via fixed DSLR camera. Deep learning algorithm YOLOv2 was used for vehicle detection, while Simple Online Realtime Tracking (SORT) algorithm enabled for tracking of vehicles. Perspective projection and scale factor were dealt with by remotely mapping corresponding image and real-world coordinates through a homography. The ensuing transformation of camera footage to British National Grid Coordinate System, allowed for the derivation of real-world distances on the planar road surface, and subsequent simultaneous vehicle speed estimations. As monitoring took place in a heavily urbanised environment, where vehicles frequently change speed, estimations were determined consecutively between frames. Speed estimations were validated against a reference dataset containing precise trajectories from a GNSS and IMU equipped vehicle platform. Estimations achieved an average root mean square error and mean absolute percentage error of 0.625 m/s and 20.922 % respectively. The robustness of the method was tested in a real-world context and environmental conditions.

1. INTRODUCTION

1.1 Background

Traffic monitoring systems within urban infrastructure have become an integral component for managing congestion, transport analysis, and for keeping roads safe and efficient. The ubiquity of modern-day sensors can capture the movements and interactions of people through different modes of transport, allowing for the manual analyses of highly voluminous datasets. However, with the introduction of smart city networks, driverless vehicles, 5G, and increased access to greater computing power through GPUs, the way in which traffic is monitored can be revised in order to better manage exceeding loads of information automatically and with greater accuracy. By utilising advanced technology (e.g. deep learning algorithms) in combination with dense and widespread roadside CCTV networks throughout the UK, the greater aim of this paper is to be able to automatically analyse traffic conditions in real-world contexts. In this case, a key component defining traffic conditions – vehicle speeds – will be the focus. Counting the number of passing vehicles will also be considered. Such information is necessary for understanding fine vehicle movements and subsequent interactions which contribute to road conditions. Furthermore, vehicle speed information can be used to aid the monitoring of traffic in specific locations – increasing safety measures through emergency response and managing the efficiency and environmental effects of roads by revealing areas of stress.

1.2 Aim

The aim of this study is to estimate vehicle travel speeds in real time (following detection and tracking) and under real-world conditions using video camera footage. An experimental framework for analytics is developed, where deep learning, tracking, and image mapping systems can undergo optimisation for fast and seamless operation in a combined processing workflow. Utilising different sensors the robustness of the technique is evaluated under varying real-world contexts. Validation and subsequent analysis provide insight into the effectiveness of surveillance speed monitoring in a typical urbanised environment.

2. RELATED WORK

2.1 Vehicle Detection

Traditional computer vision techniques often use Haar-like classifiers and AdaBoost machine learning algorithms for vehicle detection (Wen et al., 2015). Computer vision techniques for the analysis of urban traffic used prior to 2011 can be read in a detailed review by Buch et al. (2011).

However, more recently, Deep Neural Networks (DNNs) have taken precedence in how vehicles are detected (Fan et al., 2016). The robustness of a DNN is especially beneficial when it comes to vehicle detection in poor lighting or camera resolution conditions (Bautista et al., 2016; Chen et al., 2011). A common approach for vehicle detection is to utilise an algorithm which is both fast and precise, therefore allowing for reliable, real-time applications. Single Shot MultiBox Detector (SSD) (Liu et al., 2016), You Only Look Once (YOLO) (Redmon and Farhadi, 2018), and Faster R-CNN (Ren et al., 2016) are among the most notable models which can be used for this purpose in traffic monitoring (Fedorov et al., 2019; Tang et al., 2017).

With use of deep learning, Tang et al. (2018) achieved flow characterisation (including speed estimation), and object re-identification. The vehicle detection system utilised YOLOv2 deep learning algorithm to recognise vehicles across various categories. From resulting 2D bounding boxes, foot point coordinates (midpoint of lower corners) underwent affine transformation into 3D space preceding speed estimation. Instead, for enhanced speed estimation precision, Mask-RCNN

* Corresponding author

(Kaiming et al., 2018) algorithm can be used to map discrete pixels through segmentation (Kumar et al., 2018).

2.2 Vehicle Tracking

With deep learning detection as a backdrop, computer vision applications have been adapted to track moving vehicles. This function is necessary for discrete and simultaneous vehicle tracking. The latest research in this field concerns multi-camera vehicle tracking and reidentification. Using learned high-level features for vehicle instance representation, vehicles can be tracked and reidentified on a city-wide scale (Nguyen et al., 2019; Wu et al., 2018). Likewise, large-scale tracking is achievable with an optimisation technique using spatiotemporal vehicle trajectories combined with visual feature recognition (Tan et al., 2019). Alternatively, an example which does not utilise machine learning for detection, instead uses graph partitioning and matching with trajectory analysis to achieve multiple object and vehicle tracking from surveillance videos (Lin et al., 2012). However, in this paper vehicles are only tracked across single camera views, without any personal identification.

2.3 Camera Calibration for Speed Estimation

Computer vision techniques have also been developed with the capability of estimating vehicle speeds. These often extend detection and tracking methods by manipulating image geometry for measuring real-world distances. Usual practice of camera calibration, to accurately find interior, exterior, and distortion parameters is not always possible with pre-existing fixed monocular surveillance setups – and especially challenging when considering multiple camera locations such as in a network.

A common approach in correcting for perspective effects, resulting from camera position and orientation relative to the road surface, restores affine properties of a road plane by manually labelling two pairs of parallel lines, with pairs being orthogonal to one another (Kumar et al., 2018; Schoepflin and Dailey, 2003; Shi et al., 2018). Reference distances measured or assumed from features such as road markings or standard lane widths (Huang, 2018; Tran et al., 2018), combined with 'vanishing points' at which parallel lines meet on the image domain, provide parameters enabling for camera calibration through algorithmic optimisation (Tang et al., 2018). Vanishing point methods have also been adapted for fully automatic application with the use of vehicle dimension estimation, vehicle motion analysis and diamond space accumulation algorithms (Dubská et al., 2015; Giannakeris and Briassouli, 2018; Sochor et al., 2017). However, with the use of background modelling and cluster analysis of vehicle trajectories derived from video footage, perspective transformation is not strictly necessary for deriving vehicle speed estimations (Xiong, 2018). Furthermore, speed estimations can be derived directly from tracking information through prior knowledge of road speed limits and calculated assumptions of vehicle motion in relation to the camera (Hua et al., 2018).

3. DATA ACQUISITION

Over the course of a two-hour period on 31.05.19, a small van fitted with a dual frequency GNSS receiver and IMU device was driven around the study area situated in the city centre of Newcastle upon Tyne, UK. A Canon EOS 6D Mark II camera was manually positioned as if to capture the passing vehicle from the perspective of an elevated roadside CCTV camera. This camera was set to record at 1080/60p, 25 frames per second, with a full-frame image stabilized lens of 24mm focal length and f/3.5 aperture. Multiple passes were recorded as the vehicle changed travel directions. Weather conditions on the day of study were calm and well lit (despite an overcast sky). This therefore resulted in limited environmental interference which could influence detection, tracking, and speed estimation results – such as distortions or blurring caused by water droplets on the camera lens or movement in windy conditions. This made for perfect baseline testing conditions. Overall, four experimental cases were extracted from the passing vehicle.

4. METHODLOLOGY

Counting and estimating vehicle speeds necessitated the formation of a workflow. Pre-recorded footage initially underwent detection to define bounding box areas around vehicles within the image space. Immediately following this, vehicle detections were assigned ID numbers for counting and tracking throughout the monitoring period. Finally, by mapping the image space to a real-world coordinate system, traversed vehicle distances could be measured for simultaneous speed estimations. The workflow is presented in Figure 1.

4.1 YOLOv2 Vehicle Detection

A deep learning approach was selected for detecting vehicles within the video footage. Pre-trained model (on the COCO dataset (Lin et al., 2015)), YOLOv2, was the algorithm of choice due to its fast processing speed (potentially allowing for real-time processing application), its well tested precision, and compatibility with the tracking program.

4.2 Multiple-Object Tracking

Using the Hungarian algorithm and Kalman Filtering, Simple Online Realtime Tracking (SORT) algorithm was selected for tracking of detected vehicles (Bewley et al., 2016). Detection and tracking techniques were merged into a single processing workflow, proving advantageous when aiming towards a streamlined and automated approach. Output video and CSV files containing tracking IDs, frame numbers, and bounding box image space coordinates for each detection, were retrieved.

4.3 Road Scene Mapping

In correcting for perspective projection and scale, it was initially required that pre-processing steps were used to find corresponding pixel to real-world coordinate relationships. A subsequent perspective transformation was then used to find relationship vectors to map real-world coordinates onto the image space, allowing for real-world distance measurements within footage.

By assuming that the investigated real-world road surfaces are planar (no deviation in elevation), it can be assumed that there is a shared homography relationship with the road surfaces represented in corresponding video footage.

Numerous easy to identify points, spread somewhat evenly on the road surface, such as edges of road markings, were selected on the image road surface (see Figure 2). Corresponding real-world coordinates (in British National Grid), were found for each using Google Earth and The World Coordinate Converter ("TWCC," 2020). This method was inferior to others in terms of accuracy as point positions had to be selected by eye, but was

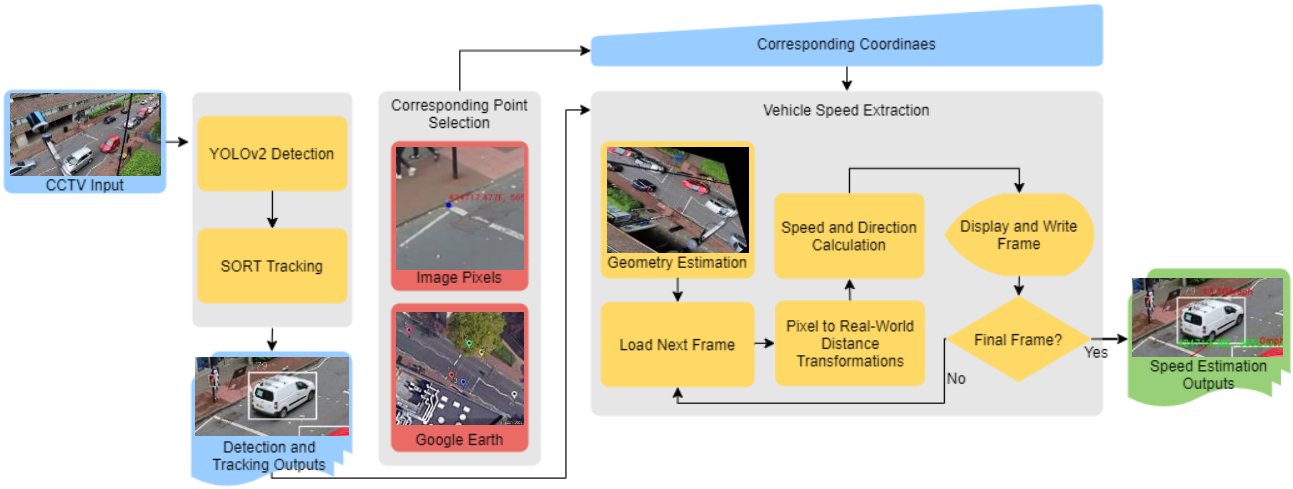

Figure 1. Processing Workflow

chosen for practicality purposes as well as improved scalability potential. Using this method allowed for points to be collected quickly and safely without requiring tedious and possibly disruptive physical measurements to be taken at each point location. Furthermore, a relative accuracy could be assumed in each camera view, as pixel and real-world coordinates were selected for each instance. This therefore limited positioning errors from multiplying, as would be the case at larger scales.

Figure 2. Original camera view with selected coordinates (red)

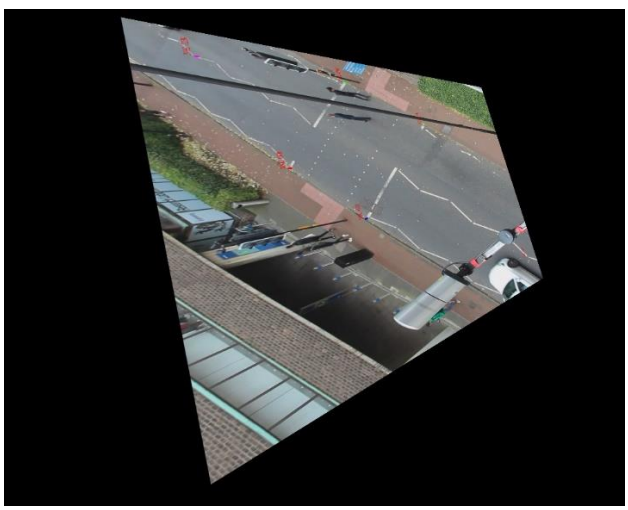

Figure 3. Transformed camera view to fit British National Grid Coordinate System

In finding a planar homography transformation, RANSAC-based robust reconstruction method was used to optimally unite the two planes through iterative calculations. Inlying points, agreeing to within a reprojection threshold of 5, were used to determine two pairs of parallel vanishing lines – one pair running parallel to the direction of moving traffic, and one pair parallel to the horizon. From each pair, a vanishing point was derived at the position at which parallel lines appeared to converge. A subsequent 3x3 homography matrix (Equation 1) was produced to define translation, rotation, and scaling parameters, constrained by eight degrees of freedom. By applying the homography transformation to the image, an output (such as Figure 3) can be observed.

$$s\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = H\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (1)$$

where H = homography
s = scale factor
x', y' = image coordinates
x, y = object coordinates

4.4 Speed Extraction

As used monitoring cameras were in a heavily urbanised city centre location, which had constant, and often congested traffic, it could not be assumed vehicles would be moving at a constant speed between any two points on the road surface. Frequent changes in acceleration therefore prevented the use of average speed estimation between two separate points. Instead, vehicle speeds were updated for each frame throughout the footage.

Traces of vehicle travel were drawn from the centre-midpoint positions of bounding boxes to follow the motion of the detected vehicle between consecutive frames. This position was chosen as it is generally the closest average position to the road plane. To determine the real-world distances of vehicle traces, start and end points were automatically transformed from pixel coordinates to British National Grid coordinates using parameters found prior for the homography. Time stamps were then approximated for each video frame based on time of initial recording and frame rate of the footage. Speed estimates were then made (in m/s) from real-world distance and time measurements. Traces were also used to identify whole circle bearings of vehicle travel from within the transformed image

space. Averaging these readings for general movement direction proves useful when undertaking spatiotemporal analyses.

Persistently updating speed between frames meant for a large set of fluctuating values and noise. Subsequently, a basic filter was used to remove extreme values (such as those well above speed limits), before a six-degree polynomial filter was applied. By fitting such a trendline, the smooth motion of the vehicle was replicated for an improved ability to minimise the effect of variation in detected bounding boxes.

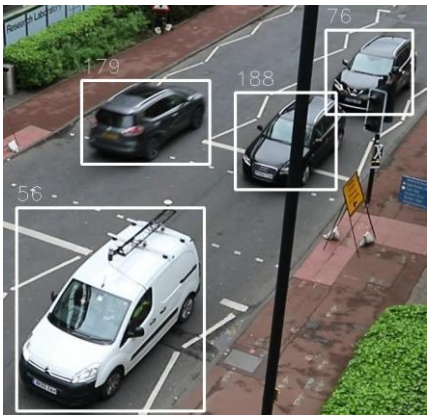

Figure 4. Detection and tracking outcomes, with bounding box detections (white), and tracking ID numbers (top left of bounding boxes, white)

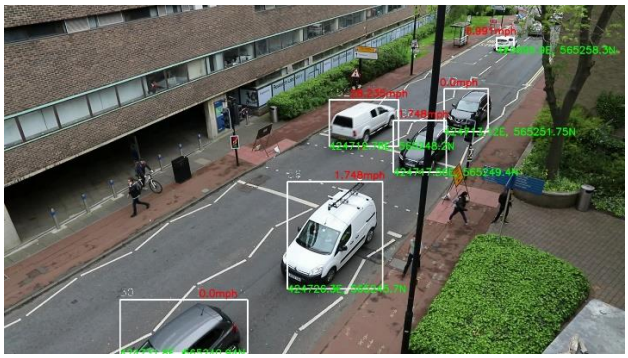

Figure 5. Results imposed on DSLR camera footage to display bounding box detections (white), automatically assigned vehicle tracking numbers (white), travel traces (with corresponding real-world coordinates) (green), and speed estimation results (red).

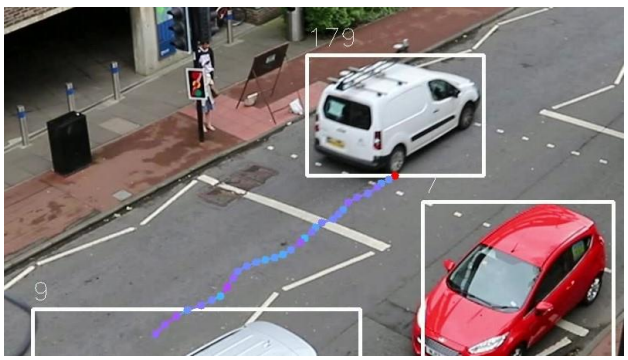

Figure 6. Vehicle platform fitted with GNSS and IMU devices (white van, tracking ID 179). Blue and pink markers represent vehicle speed (gradient from blue, low values, to pink, higher values approaching speed limit) and position over the past 30 frames. Red marker represents current vehicle position.

4.5 Validation

Data recorded from the GNSS receiver and IMU were processed to find precise coordinates and trajectories, which were then used to determine vehicle travel speeds between consecutive epochs. Using GPS time information from the geolocation function on the camera, estimated and precise vehicle speeds were aligned in time. For this, reference speeds were thinned from 100 readings per second to match the corresponding video frame rate – therefore allowing direct speed comparisons in time. As for estimated speeds, reference speeds were smoothed using the same technique. Several metrics, such as root mean square error, mean absolute error, and mean absolute percentage error were determined when comparing datasets.

5. EXPERIMENTAL RESULTS

5.1 Detection and Tracking Results

The YoloV2 and SORT algorithms successfully monitored vehicles as they passed through the image space. By visually inspecting the result footage it was clear that detection bounding boxes were largely true-positive, remaining fixed on the vehicle throughout. False-positive outcomes were uncommon but did not influence speed estimations, so were disregarded. On occasion, there were several frames in which the detection algorithm missed a vehicle. However, this did not impede on the ability of SORT to re-identify the vehicle for continuous speed estimation. Therefore, as the primary focus of this paper was vehicle speed estimation, a statistical evaluation of detection rate was not carried out. Instances when temporary detection loss did occur, was often due to occlusion from other vehicles on the opposite side the road. This was not so much of an issue for speed monitoring – once the vehicle was re-identified, average speed was calculated from the last known time and position. Table 1 shows the length of time the refence vehicle was in camera view, as well as number of detected instances (frames). Such information is necessary in understanding averaged speed results with respect to quantity of data available to produce an accurate forecast.

Experiment No.	Total Time in Camera View (s)	Footage FPS	Detected Instances (frames)
1	54.16	25	1343
2	9.32	25	234
3	17.52	25	425
4	63.16	25	1510

Table 1. Recording and Detection Results

The principle axis pointed diagonally across the road surface, like that of many existing road surveillance cameras. Additionally, this maximised the length of time vehicles were within frame and subsequent speed measurement volumes. Detected instances of experiment 1 and 4 are much larger than in other cases. This is due to the vehicle being stationary or moving slowly within footage for extended periods, as evidenced in Figure 7 speed plots.

5.2 Speed Estimation Statistics

Speed values of each instance were collectively summed and divided by number of instances to estimate mean speeds, which could be used for macroscopic traffic flow modelling. These, as well as absolute differences between means, are presented in

Table 2. Absolute difference in mean speed give general indication of result quality over the whole monitoring period.

Experiment No.	Reference Mean Speed (m/s)	Estimated Mean Speed (m/s)	Absolute Difference (m/s)
1	0.452	1.012	0.560
2	4.393	5.098	0.706
3	2.825	2.498	0.327
4	0.872	0.973	0.101
Average	-	-	0.424

Table 2. Mean Speed Evaluation

However, as vehicles were not usually travelling at constant speeds, such simplified metrics limit understanding of result quality. Furthermore, estimated speeds deviated both above and below reference speeds, which potentially allowed for cancelling out of differences. Despite this, the results presented here allow broad understanding of result quality.

On the other hand, individual vehicle movements are tracked on a sub second basis, which enables detailed traffic microsimulation. Table 3 presents statistics aimed at better understanding the magnitude of errors. Mean absolute error (MAE) takes the absolute values of individual instances before collective summation and division by number of instances. Mean absolute percentage error (MAPE) assess the quality of forecasted data against the reference dataset. As this is expressed as a percentage, greater understanding can be given to quality of results with respect to volume of speed data.

Experiment No.	RMSE (m/s)	MAE (m/s)	MAPE (%)
1	0.821	0.760	33.014
2	1.171	0.892	35.532
3	0.378	0.333	11.878
4	0.130	0.118	3.264
Average	0.625	0.526	20.922

Table 3. Statistical Speed Evaluation

In terms of order of accuracy, MAE values agree with those of RMSE. As seen in Table 3, experiment 4 outcomes continue to show minimal error across all measurement types, resulting in best performance with an RMSE of 0.130 m/s and MAE of 0.118 m/s. On the other hand, experiment 2 consistently shows the poorest outcomes, with an RMSE of 1.171 m/s and a MAE of 0.892 m/s. Over all experiments, RMSE averages to a value of 0.625 m/s, whereas MAE has an average value of 0.526 m/s. The MAPE values of the four experiments varied regardless of the magnitude of vehicle speed.

5.3 Speed Plots

Experiment 1 (Figure 7) depicts speed initially in a state of rapid decline, before a levelling off as the vehicle comes to a stop due to a build-up of traffic preventing further movement. This stationary position continues for much of the monitoring period, and therefore results in a low reference mean speed of 0.452 m/s. Around the 62 second mark the vehicle gathers speed as surrounding traffic clears. On the other hand, the shortest of all monitoring cases, experiment 2 (Figure 7), presents the vehicle moving at a near consistent speed at approximately 5-6 m/s for the first half. Following this, the vehicle gradually slows to a stop, again due to traffic, outside of camera view. Reference mean speed is highest in this case, at 4.393 m/s.

Experiments 3 and 4 (Figure 7) show the results of vehicle movements in the same location at peak congestion, where traffic is often slow moving or stagnant. Over the course of about 20 seconds, experiment 3 demonstrates the vehicle maintaining slow, but consistent speed, before beginning to slow to a stop as it moves from the camera view. On the other hand, the vehicle recorded in experiment 4 remains within camera view for much longer (~65 seconds), as it moves to a near stop on two occasions due to a build-up of traffic. Experiment 4 shows a strong relationship between estimated and reference data throughout, resulting in an RMSE value of 0.130 m/s and MAPE of 3.264 %. Results in experiment 3 are not so closely related, but maintain a slightly offset relationship to reference data as the vehicle accelerates. Consequently, this results in an RMSE of 0.378 m/s and MAPE of 11.878 %.

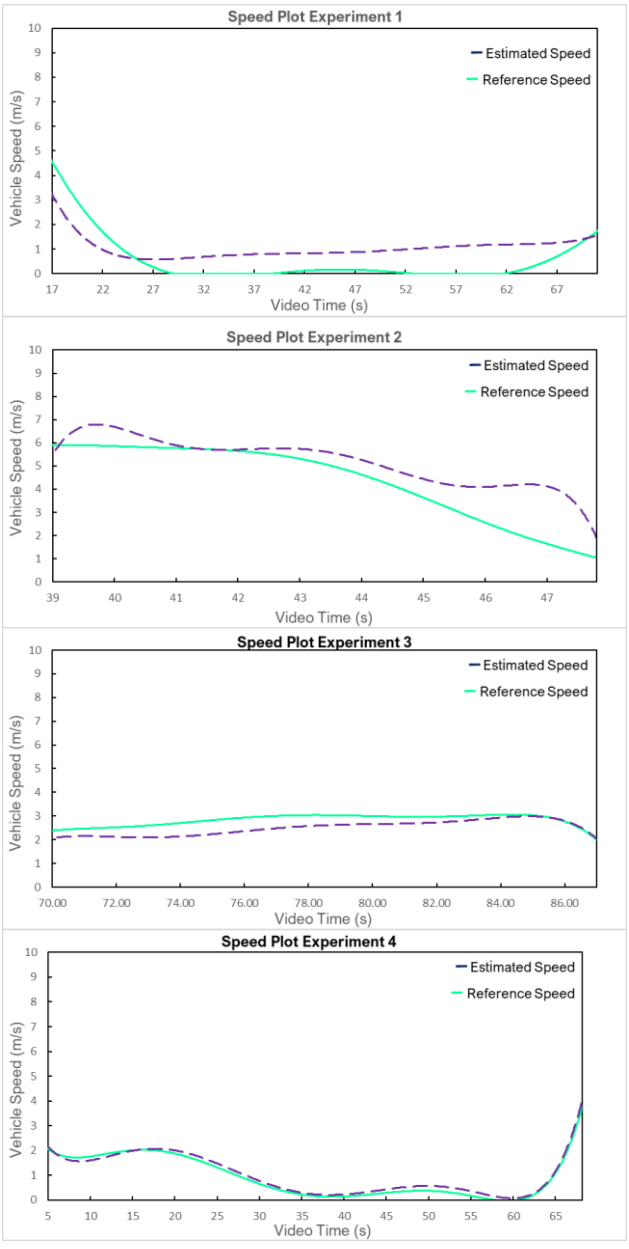

Figure 7. DSLR Speed Estimation Results

5.4 Speed Analysis

Speed plots depict estimated vehicle speeds over the course of video recording time, with respect to corresponding reference speeds. Experiment 1 (Figure 7) vehicle reference speeds suggest slow or stationary conditions for much of the monitoring period. Despite estimated speeds also presenting low speeds throughout, there is no indication of the vehicle being stationary at any point in time. The difference between slow moving and stationary traffic may be minimal in terms of values, but in traffic flow analysis such a difference may classify as an alternate conditional state. These differences may well have been caused due to noise generated from bounding box detections. With re-detection occurring at each consecutive frame, newly generated bounding boxes do not always align with previous rectangles. Instead, boxes may be offset or of varying size with respect to true vehicle position and dimensions within the image space. As the detection algorithm is optimised for speed, taking 'one look' at each frame before classification and moving to the next, it does not have the ability to solve for fine grained positioning. Such anomalies are not as noticeable as the vehicle is moving, and largely cancel out as bounding boxes alternate between falling behind and ahead of true vehicle position. However, as the vehicle is stationary, all noise generated is positive and therefore averages to a low speed.

On the other hand, estimated results presented for experiment 4 show the vehicle moving slowly over the course of monitoring, almost approaching a stop on two occasions. However, as the vehicle does not come to a complete stop for an extended period, estimations appear to better match reference data.

Results from DSLR experiment 2 and experiment 3, show the vehicle gradually accelerating and decelerating, respectively, throughout the course of monitoring. Despite some positive offset, estimations of experiment 3 appear to more consistently match reference speeds than those of experiment 2. This is likely due to rate of change being more dramatic in the latter case, over a shorter period of monitoring. This further supports the idea that, with longer monitoring time, better results are produced using this technique.

6. DISCUSSION

Over all metrics, experiment 4 demonstrated the highest achieving values, whereas experiment 2 generally demonstrated the poorest outcomes. This coincides with length of monitoring time and subsequent number of frame detections. It therefore suggests this experimental framework operates best with a greater volume of estimated speed data. This can be achieved by positioning and orienting the camera to increase road area visibility, or by increasing recording framerate. The speed of the vehicle also determines length of time in camera view and subsequent data volume. Nevertheless, if applied to sensor networks in a real-world context, such operations may not be feasible with fixed CCTV cameras. However, despite what is suggested by highest and lowest achieving results, other experimentation cases (1 and 3) do not support this trend. It is more likely that volume of data is only one contributing factor towards result outcomes. Factors such as accuracy of camera calibration, lens distortion, and consistency of detection and tracking precision also largely influence final speed estimations. Raw speed results (especially from those which were estimated) proved to be very noisy. This is most likely due to the inconsistent positioning of the bounding box as it re-detected the vehicle in each frame. By not re-evaluating each detection, the algorithm lacked in accurately defining the boundaries of vehicles, as a more time costly segmentation mask (such as Mask-RCNN) would achieve. Despite this, detection precision is maintained throughout this process, producing mostly true-positive outcomes. As speed estimations are measured by distance traversed between consecutive frames, many false and noisy speed readings occur. Comparing reference and estimated speed results in this raw form would produce poor correlation. However, by smoothing the results, as was accomplished in this case, improved correlation between datasets was achieved.

Overall, experimental speed estimation results had an RMSE of 0.625 m/s. In terms of typical traffic monitoring, this outcome is more than suitable – allowing for understanding into mirco-level vehicle speeds and, when applied to traffic simulations, general flow characteristics. When simply looking at the average magnitude of errors, rather than quadratic scoring, overall MAE results present an average value of 0.526 m/s. As MAE gives a relatively low weight to large errors, in comparison to RMSE, this was to be expected.

In comparison to other, similar methods (using deep learning algorithms for detection and homography style techniques for image mapping), but which estimate vehicle speeds on highways rather than non-free-flowing urban roads, the used technique in this paper presents promising performance. For example, (Kumar et al., 2018) who use Mask-RCNN for detection, achieved an RMSE of 9.54mph (4.265 m/s). Similarly, (Shi et al., 2018) also use Mask-RCNN to obtain a speed estimation RMSE of 6.667mph (2.98 m/s). Despite both using an algorithm of superior precision, errors here are greater than experiments presented in this paper. Track 1 of the NVIDIA AI City Challenge presents various techniques for speed estimation using a monocular camera setup (Naphade et al., 2018). Here, RMSE values range from 4.096 mph – 27.302 mph (1.831 m/s – 12.205 m/s). Again, the average RMSE of 0.625 m/s presented in this paper reveals a promoting outcome.

No evaluation was carried out into precision and recall of the algorithm in this case. However, with a VOC 2007 mAP of 78.6 %, it was decided YOLOv2 would be a robust enough solution. Likewise, no evaluation was carried out into success rates of this tracking algorithm. Despite this, SORT qualified as the best open source multiple object tracker in the 2015 Multiple Object Tracking benchmark challenge (Milan et al., 2015), and thus provided the necessary processing speed and accuracy required for the task at hand.

Furthermore, as vehicles move through the image space, size, orientation and appearance of the vehicle changes with respect to the camera view. Speed estimations will be affected by these perspective effects. However, by smoothing speed values, such errors are largely minimised. Alternatively, one solution to this would be to adopt the approach taken by (Sochor et al., 2018), who model vehicle boundaries in 3D. As a result, vehicle position can be obtained with greater accuracy and consistency as it traverses the scene.

One other factor which must be taken into consideration when comparing results, is the reliability of reference data. GNSS and IMU devices were processed for deriving precise vehicle trajectories. However, as experimentation took place in a city centre environment, urban canyon effects, such as multipath, may have been present.

Measuring vehicle speed in rapid succession over the course of monitoring may be applicable within several applications –

broad and fine scale analyses of free flowing/congested traffic, anomaly/incident detection, or automated monitoring. Such a technique may also prove beneficial in better understanding the impact of traffic speeds and volumes on the wider urban environment. With use of other sensor networks, multimodal analyses can be conducted in relation to air quality, noise pollution, or the planning and maintenance of infrastructure. However, for safety critical monitoring applications, such as incident/anomaly detection, or more detailed spatial analysis, the speed estimation errors achieved in this paper are not always tolerable. In the UK, conventional speed cameras are expected to be consistently within 1mph (~0.45 m/s) error, which, when looking at RMSE, is achieved by two of the four experiments. It means that comparable accuracy can be expected by ordinary video cameras, showing great potential to be used for city scale traffic monitoring.

7. CONCLUSIONS

A workflow was designed and developed to semi-automatically detect, track, and estimate vehicle speeds. Current results are promising and therefore pave the way for further development and investigations as the project moves towards fully-automated and real-time solutions.

The used technique successfully estimated vehicle speeds to an average RMSE of 0.625 m/s from a DSLR camera setup. This was accomplished travelling along congested city centre roads, rather than on highways where vehicles are assumed to move at near constant speeds. Therefore, unlike related research, which often takes a single speed average over a defined distance, the scenario here necessitated taking multiple speed measurements throughout monitoring to account for frequent changes in vehicle acceleration.

Continuing research plans consist of extending the framework for more detailed investigation into the robustness of the used technique. This can be accomplished by testing with more camera sensors and over longer periods of time for more varying context and environmental conditions (such as different levels of daylight, or adverse weather conditions).

REFERENCES

Bautista, C.M., Dy, C.A., Mañalac, M.I., Orbe, R.A., Ii, M.C., 2016. Convolutional neural network for vehicle detection in low resolution traffic videos, in: IEEE Region 10 Symposium (TENSYMP). IEEE, Bali, Indonesia, pp. 277–281. doi:10.1109/TENCONSpring.2016.7519418.

Bewley, A., Ge, Z., Ott, L., Ramos, F., Upcroft, B., 2016. Simple online and realtime tracking, in: Proceedings - International Conference on Image Processing, ICIP. doi:10.1109/ICIP.2016.7533003.

Buch, N., Velastin, S.A., Orwell, J., 2011. A Review of Computer Vision Techniques for the Analysis of Urban Traffic. IEEE Trans. Intell. Transp. Syst. 12, pp. 920–939. doi:10.1109/TITS.2011.2119372.

Chen, Y., Wu, B., Member, S., Huang, H., Fan, C., 2011. A Real-Time Vision System for Nighttime Vehicle Detection and Traffic Surveillance. IEEE Trans. Ind. Electron. 58, 2030–2044. doi:10.1109/TIE.2010.2055771.

Dubská, M., Herout, A., Juránek, R., Sochor, J., 2015. Fully Automatic Roadside Camera Calibration for Traffic Surveillance. IEEE Trans. Intell. Transp. Syst. 16, 1162–1171. doi:10.1109/TITS.2014.2352854.

Fan, Q., Brown, L., Smith, J., 2016. A Closer Look at Faster R-CNN for Vehicle Detection. 2016 IEEE Intell. Veh. Symp. 124–129. doi:10.1109/IVS.2016.7535375

Fedorov, A., Nikolskaia, K., Ivanov, S., Shepelev, V., Minbaleev, A., 2019. Traffic flow estimation with data from a video surveillance camera. J. Big Data 6. doi:10.1186/s40537-019-0234-z

Giannakeris, P., Briassouli, A., 2018. Speed Estimation and Abnormality Detection from Surveillance Cameras, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). IEEE, Salt Lake City, UT, USA, pp. 93–99. doi:10.1109/CVPRW.2018.00020.

Hua, S., Kapoor, M., Anastasiu, D.C., 2018. Vehicle Tracking and Speed Estimation from Traffic Videos, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). CVF, Salt Lake City, UT. doi:10.1109/CVPRW.2018.00028.

Huang, T., 2018. Traffic Speed Estimation from Surveillance Video Data Institute for Transportation , Iowa State University, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). CVF, Salt Lake City, UT, pp. 161–165. doi:10.1109/CVPRW.2018.00029.

Kaiming, H., Georgia, G., Dollár, P., Ross, G., 2018. Mask R-CNN. Computing Research Repository (CoRR), Cornell University (https://arxiv.org/abs/1703.06870).

Kumar, A., Khorramshahi, P., Lin, W., Dhar, P., Chen, J., Chellappa, R., 2018. A Semi-Automatic 2D solution for Vehicle Speed Estimation from Monocular Videos, in IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). CVF, Salt Lake City, UT, pp. 137–144. doi:10.1109/CVPRW.2018.00026.

Lin, L., Lu, Y., Pan, Y., Chen, X., 2012. Integrating Graph Partitioning and Matching for Trajectory Analysis in Video Surveillance. IEEE Trans. Image Process. 21, 4844–4857. doi:10.1109/TIP.2012.2211373

Lin, T.-Y., Maire, M., Belongie, S., Bourdev, L., Girshick, R., Hays, J., Perona, P., Ramanan, D., Zitnick, C.L., Dollar, P., 2015. Microsoft COCO : Common Objects in Context. Computing Research Repository (CoRR), Cornell University (https://arxiv.org/abs/1405.0312).

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C., Berg, A.C., 2016. SSD : Single Shot MultiBox Detector. Computing Research Repository (CoRR), Cornell University (https://arxiv.org/abs/1512.02325).

Milan, A., Reid, I., Roth, S., Schindler, K., Leal-taix, L., 2015. MOTChallenge 2015 : Towards a Benchmark for Multi-Target Tracking. Computing Research Repository (CoRR), Cornell University (https://arxiv.org/abs/1504.01942).

Naphade, M., Chang, M.-C., Sharma, A., Anastasiu, D.C., Jagarlamudi, V., Chakraborty, P., Huang, T., Wang, S., Liu, M.-Y., Chellappa, R., Hwang, J.-N., Lyu, S., 2018. The 2018 NVIDIA AI City Challenge San José, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition

Workshops (CVPRW). Salt Lake City, UT, pp. 53–60. doi:10.1109/CVPRW.2018.00015

Nguyen, K.-T., Hoang, T.-H., Le, T.-N., Bui, N.-M., Do, T.-L., Vo-Ho, V.-K., Lương, Q., Tran, M.-K., Nguyen, T.-A., Truong, T.-D., Nguyen, V.-T., Do, M., Tran, M.-T., 2019. Vehicle Re-identification with Learned Representation and Spatial Verification and Abnormality Detection with Multi-Adaptive Vehicle Detectors for Traffic Video Analysis, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops. IEEE.

Redmon, J., Farhadi, A., 2018. YOLOv3: An Incremental Improvement. Computing Research Repository (CoRR), Cornell University (https://arxiv.org/abs/1804.02767).

Ren, S., He, K., Girshick, R., Sun, J., 2016. Faster R-CNN : Towards Real-Time Object Detection with Region Proposal Networks. Computing Research Repository (CoRR), Cornell University (https://arxiv.org/abs/1506.01497), pp. 1–14.

Schoepflin, T.N., Dailey, D.J., 2003. Algorithms for Estimating Mean Vehicle Speed Using Uncalibrated Traffic Management Cameras by. Seattle, Washington.

Shi, H., Wang, Z., Zhang, Y., Wang, X., Huang, T., 2018. Geometry-aware Traffic Flow Analysis by Detection and Tracking, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). CVF, Salt Lake City, UT, pp. 116–120. doi:10.1109/CVPRW.2018.00023.

Sochor, J., Jur, R., Herout, A., 2017. Traffic Surveillance Camera Calibration by 3D Model Bounding Box Alignment for Accurate Vehicle Speed Measurement. Comput. Vis. Image Underst. 87–98. doi:10.1016/j.cviu.2017.05.015.

Sochor, J., Spanhel, J., Herout, A., 2018. BoxCars : Improving Fine-Grained Recognition of Vehicles Using 3-D Bounding Boxes in Traffic Surveillance, in: IEEE Transactions on Intelligent Transportation Systems. doi:10.1109/TITS.2018.2799228. pp. 97–108.

Tan, X., Wang, Z., Jiang, M., Yang, X., Wang, J., Gao, Y., Su, X., Ye, X., Yuan, Y., He, D., Wen, S., Ding, E., 2019. Multi-camera vehicle tracking and re-identification based on visual and spatial-temporal features, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops. CVF, Salt Lake City, UT, USA, pp. 275–284.

Tang, Z., Wang, G., Liu, T., Lee, Y., Jahn, A., Liu, X., He, X., Hwang, J., 2017. Multiple-Kernel Based Vehicle Tracking Using 3D Deformable Model and Camera Self-Calibration. IEEE Smart World NVIDIA AI City Chall.

Tang, Z., Wang, G., Xiao, H., Zheng, A., Hwang, J., 2018. Single-camera and inter-camera vehicle tracking and 3D speed estimation based on fusion of visual and semantic features, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). IEEE, Salt Lake City, UT, USA, pp. 108–115. doi:10.1109/CVPRW.2018.00022.

Tran, M., Dinh-duy, T., Truong, T., Ton-that, V., Do, T., 2018. Traffic Flow Analysis with Multiple Adaptive Vehicle Detectors and Velocity Estimation with Landmark-based Scanlines, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). CVF, Salt Lake City, UT, pp. 100–107.

TWCC [WWW Document], 2020. URL https://twcc.fr/en/# [Accessed: 1 February 2020].

Wen, X., Shao, L., Xue, Y., Fang, W., 2015. A rapid learning algorithm for vehicle classification. Inf. Sci. (Ny). pp. 295, 395–406. doi:10.1016/j.ins.2014.10.040.

Wu, C., Liu, C., Chiang, C., Tu, W., Chien, S., 2018. Vehicle Re-Identification with the Space-Time Prior, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). CVF, Salt Lake City, UT.

Xiong, L., 2018. Dual-Mode Vehicle Motion Pattern Learning for High Performance Road Traffic Anomaly Detection, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). CVF, Salt Lake City, UT. doi:10.1109/CVPRW.2018.00027.