

MINIMISING MISCLASSIFICATIONS OF OVER-HEIGHT VEHICLES DUE TO WIND

Bella Nguyen¹ and Ioannis Brilakis²

Abstract: Over-height vehicle strikes with low bridges and tunnels are an ongoing problem worldwide. While previous methods have used vision-based systems to address the over-height warning problem, such methods are sensitive to wind. In this paper, we propose a constraint-based approach to minimise the number of over-height vehicle misclassifications due to windy conditions. The dataset includes a total of 102 over-height vehicles recorded at frame rates of 25 and 30 fps. At this frame rate, we analysed sampling rates to determine the sufficient number of positive frames required to provide accurate warnings to drivers. Optical flow and KLT feature-tracker algorithm was used to detect and track feature points of motion. Motion captured within the region of interest was treated as a standard two-class binary linear classification problem with 1 indicating over-height vehicle presence and 0 indicating noise. The algorithm performed with 100% recall, 83.3% precision and false positive rate of 8.3%.

Keywords: Bridge strike, tunnel strike, over-height vehicle, over-height vehicle detection system, bridge strike prevention.

1 INTRODUCTION

A bridge or tunnel strike is an incidence in which a vehicle, typically a lorry or double-decker bus, tries to pass under a bridge or tunnel that is lower than its height, subsequently colliding with the structure (Nguyen and Brilakis 2016a). Bridge and tunnel strikes are recurrent incidents, often costing thousands of pounds in repairs and causing hours of delays; this disruption that can be felt across various road and rail networks. When a strike occurs, the event can affect five key groups: these include owner/operators of roadways, railways, and bridges/tunnels, as well as both vehicle drivers and the wider public.

Current state of practice in strike prevention can be split into three categories, each with limitations: passive (effective only 10-20% of the time), sacrificial (costly) and active systems (also costly). Although expensive, the active laser-beam systems are the most effective. However, due to the high costs associated with pole erection and installation (£60k+ per pole), adoption has been limited (Dai et al. 2015). Particularly in severe wind, active systems are more prone to false positive warnings.

In this paper, we address the wind sensitivity issue by using a constraint-based approach to minimise the number of over-height (OH) vehicle misclassifications resulting from windy camera conditions. Our approach involves understanding the behaviour of moving vehicles and wind motion in order to prevent misclassifications and resultant erroneous warnings to drivers.

¹ PhD Candidate, Department of Engineering, Laing O'Rourke Centre for Construction Engineering and Technology, University of Cambridge, Cambridge, UK, bbn20@cam.ac.uk

² Laing O'Rourke Lecturer of Construction Engineering, Department of Engineering, Director of the Construction Information Technology Laboratory, University of Cambridge, Cambridge, UK, ib340@cam.ac.uk

2 BACKGROUND

Computer vision-based OH vehicle detection is a relatively new area of research. It has grown in popularity as infrastructure owners have increasingly sought more affordable methods of strike prevention.

2.1 Computer Vision-Based Methods

One of the earliest computer vision-based solutions to the OH problem involves height estimation of moving objects. This approach uses vanishing lines to estimate vehicle height (Shao, Zhou, and Chellappa 2010; Dai et al. 2015). Although these methods have yielded favourable results, they rely on the sometimes problematic assumption that vehicles are visible in the images. On a multi-lane roadway, this is often untenable, given the number of vehicle occlusions in a typical scene.

Nguyen et al. (2016b) expand on the height estimation concept by introducing a threshold line approach which mimics the behaviour of the active laser-beam method. In Fig. 1, under ideal sunny yet minor-windy weather conditions, the distinction between OH vehicle, wind and other noise is distinguishable by the average number of feature points detected. When the number of feature points are high, we expect to see motion consistent with a vehicle passing through the frame; otherwise, any detected features points can be classified as non-vehicle movement. While the threshold line approach performs ideally in sunny weather conditions, the performance drops by nearly 31.0% in windy weather.

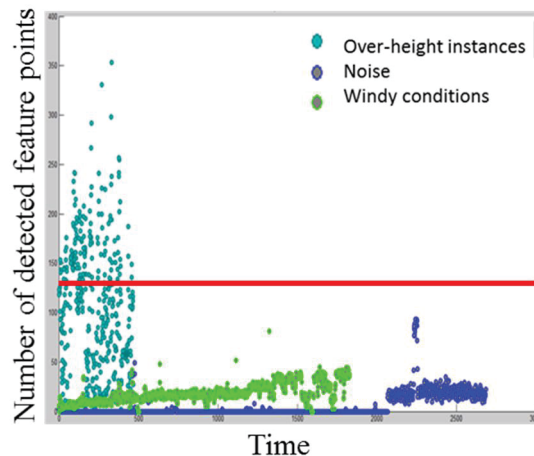


Figure 1: The plot show the number of feature points detected over time in ideal, sunny, minor windy conditions.

2.2 Related Methods

In an OH context, although the camera is intended to be static and fastened to a secure bracket, severe wind can still cause motion in the system, due to the slenderness of the mounting pole and its height relative to the road plane. This problem can be addressed using feature- or intensity-based methods.

2.2.1 Feature-based methods

Feature-based methods make use of distinguishable attributes such as edges (Canny 1986) and corners (Harris and Stephens 1988) to locate areas of interest applicable for OH detection. Corners are common features to track, due to their distinctive edges and significant change in intensity values in all directions. Corners can be expressed as

$$E(u, v) = \sum_{x, y} w(x, y) [I(x + u, y + v) - I(x, y)]^2 \quad (1)$$

representing the window function and difference in intensity shifts. For distinctive patches, the values are larger representing a great change in intensity i.e. corners of moving objects.

2.2.2 Intensity-based methods

Optical flow is a widely-used intensity-based method which estimates the motion of image velocities or discrete image displacements between two image frames at frames I and I_{i+1} in combination with the Kanade-Lucas-Tomasi (KLT) feature-tracker algorithm. The optical flow estimation can be computed for 2D cases as the intensity $I(x, y, t)$ is shifted by u, v and y between the two image frames using a brightness constancy constraint given as

$$I(x, y, t) = I(x + u, y + v, t + 1) \quad (2)$$

where the x and y components represent the velocity of $I(x, y, t)$.

2.3 Motivation

While there exist previous methods that use computer vision-based systems to address the OH problem, they are sensitive to wind. In Fig.2 (a) and (b), where wind is present, the feature points are indistinguishable windy camera movements resulting in false positive detections. Here, we extend the research of Nguyen et al. (2017) to address the wind sensitivity issue. We do this by understanding the behaviour of OH vehicles and windy weather conditions, with the aim of minimising misclassification due to windy conditions.

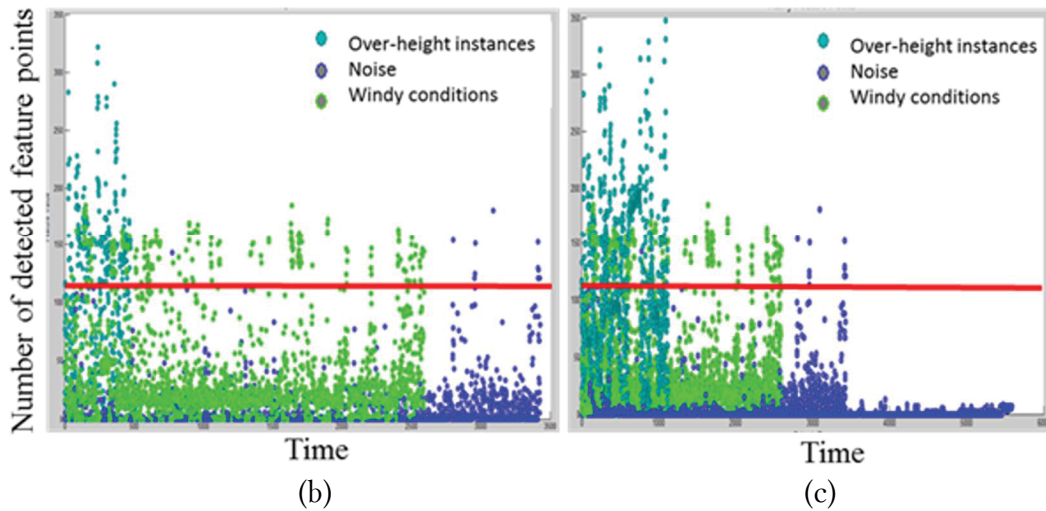


Figure 2: Detected features points in (a) cloudy and windy and (b) rainy and windy weather conditions.

3 PROPOSED SOLUTION

3.1 Overall Framework

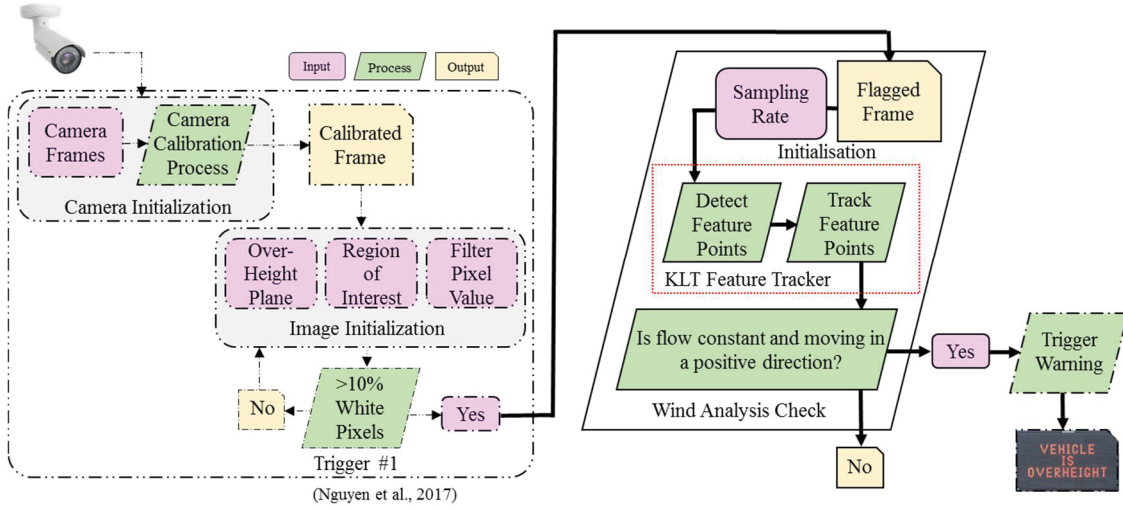


Figure 3: overall framework of over-height vehicle detection system to account for wind sensitivity

The framework shown in Fig. 3 extends Nguyen et. al (2017). The extension addresses the wind sensitivity issue by identifying common characteristics of OH vehicles and wind behaviours. We assume that OH vehicles behaviours are: (1) over a specific height relative to the road plane, (2) travelling in a specific direction (easterly direction at distance d from the camera position) and (3) travelling at a constant speed. We assume the vehicle is not decelerating and/or stationary (parked) in front of the camera view. We can apply these assumptions to a windy scenario. In this scenario, we assume that the camera will endure constant swaying, and that this motion will occur over a significant number of frames (5+) rather than forming an isolated event (1 or 2 frames). Any motion captured within the region of interest is analysed, and can be treated as a binary linear classification problem such that

$$\text{Trigger Warning System} = \begin{cases} 1, \text{overheight vehicle} \\ 0, \text{noise} \end{cases} \quad (3)$$

1=OH vehicle and 2=noise (caused by wind or camera movement).

3.1.1 The algorithm

The algorithm adds to the assumptions stated in Section 3.1 with a new control variable to analyse the direction of movement of features. This control variable analyses the direction of flow movement and scores each feature-point. The algorithm is initialised when the flagged frame is passed through into the wind analysis check. The sampling rate information is used as input to activate the KLT feature-tracker detection algorithm. Any motion passing through the region of interest is detected and tracked. Each point is tracked over five consecutive frames and analysed with reference to its neighbours to determine whether the flow is constant and moving in a positive direction. If the flow is constant i.e. monotonically increasing in the horizontal pixel dimension, then a warning is displayed on the OH sign. If the motion is inconsistent and disconnected, the instances are classified as noise, and the process starts over.

Step 1 (Wind analysis check): The flagged image is initialised using the input information of the sampling rate. ‘Sampling rate’ refers to the processed interval of frames in a given dataset. For example, if a video is recorded at 25 frames per second, the algorithm can be set to analyse every 1, 5, 10, 30 frames etc., in order to minimise the amount of processing time required. If a vehicle is travelling at 30 mph at a recorded frame rate of 25 fps, we expect the OH vehicle to be captured in 90 ± 5 frames. If we were to sample once every 5 frames, we would expect a positive occurrence 19 times for the same vehicle, equating to 19 positive warnings. This is excessive; only one warning is required for every OH classification, and so the sampling rate needs to be analysed.

Step 2: In this step, we use optical flow to find the physical movement of the 2D projection of the feature points relative to the 2D displacement of pixel patches on the image plane. The initial video frame is a 2D grayscale image, I_i and the feature points are detected using corresponding interest points between a pair of images using local neighbourhoods and the corner Harris algorithm. The algorithm finds the corners and extracts neighbourhood features. The neighbourhood features are matched and locations of the corresponding feature points are retrieved for each consecutive image. This can be expressed as

$$(I_i, I_{i+1}, I_{i+2}, I_{i+3}, I_{i+4}, \dots, I_{i+N})$$

where $N \leq$ number of positive frames.

Step 3: The feature points are initialised and tracked to specify the initial point and video frame location in $[x\ y]$ coordinates. The point tracker tracks a set of points using the KLT feature-tracking algorithm from one frame to the next. A block size is initiated to specify the size of the neighbourhood represented by a two-component vector $[height, width]$ around each point being tracked. The neighbourhood corresponds to the spatial matrix area and the block size is set to a minimum to decrease computation time. The output points are an N -by-2 array of $[x\ y]$ coordinates that correspond to the new point locations in the subsequent frame. The validity of the points is scored using a confidence scoring technique for each point between 0 (*poor*) and 1 (*perfect*). The scores are a function of the sum of squared differences between the previous and new neighbourhoods. The values correspond to the degree of similarity between the neighbourhood around the previous and new location. Since motion projects to nearby points in the image, we expect spatial coherence and constant feature flow along the x-horizontal dimension of the image. We hypothesize that this modified framework will minimise false positive classifications, and hence provide accurate warnings to drivers.

4 EXPERIMENTS & RESULTS

This section provides details of the data collection, camera specifications and methods used in the experiment, as well as classification accuracy results. The implementation was conducted on 30-50 mph speed limit roadways, which are typical where low bridges exist. The dataset consisted of video involving obstructions such as moving trees or bushes, and building façades with potential background movement, as shown in Fig. 4.



Figure 4: Data Collection Examples

The camera specifications used for the experiment was a Sony a7R ii 42 MP camera capturing a total of 88.6 mins of windy weather video footage @ 25 and 30 fps with a resolution of 1920 x 1080. Manual swaying of the pole was implemented at times throughout the data collection, in order to simulate wind movement. In total, 102 OH vehicles were recorded, yielding 132,900 positive frame instances of OH vehicles. The data was processed using MATLAB R2016a Computer Vision Toolbox on an Intel Core i7-4790. The camera was mounted on a fixed tripod pole where the θ_{yaw} was at 90° and θ_{pitch} at 10° . Although the typical UK low bridge is considered to be 5.03m (Department of Transport), in this experiment, we set the camera height between 2.5m and 3.2m to evaluate the flows of the average tall vehicle. We chose to lower the height of the camera to allow for a larger data set.

OH vehicle detection requires a sufficient number of frames to accurately detect the vehicle. The purpose is to distinguish between true OH presence and movement resulting from wind. If the frame and sampling rates are lowered, there is a higher probability of false negatives, whereas at higher rates we have excess frames which increases processing and computing costs. Therefore, an evaluation of the frame and sampling rates is required to determine the optimal parameters.

The experiment was divided into two parts, involving videos recorded at 25 fps and at 30fps. For each frame rate, we evaluated the system at 5-frame interval sampling rates i.e. every 1, 5, 10, 15, 20, 25 30 frames to determine the optimal sampling rate with sufficient accuracy. Within each sampling rate, we determined the upper and lower limits of the number of positive frames at each frame rate. The number of positive frames is important, as the trigger is set to warn the driver when five consecutive frames have positive motion. Five consecutive frames were chosen as this provided sufficient positive frames to track should the features points fail to track successfully in the initial frames. Optical flow was used to detect and track the motion features using the Harris corner detection and KLT feature-tracker algorithm. A confidence score was assigned to each point to assess validity.

4.1 Results

As shown in Table 1, frame rates used were 25 and 30 fps. If we were to take the dataset recorded at 25 fps with a sample rate of every 1 frame, on average, we would expect to find 49.5 positive frames of OH vehicle instances ranging from an upper limit of 62, lower limit of 37 and average of 49.5 positive frames. This resulted in an average of 8.5 warnings per OH occurrence. Although the algorithm was able to recall positive instances, the multiple warnings decreased the precision to 85.9% compared to a ground truth of 9.9

positive warnings. The values in grey* did not meet the minimum required number of positive frames for consecutive flow analysis and were therefore removed from the dataset. The eliminated values included both 25 and 30 fps video sampled every 15, 20, 25, and 30 frames. The ideal parameters were found to be 25 fps at a sample rate of every 10 frames. The average number of positive frames with these parameters is 12, with an average warning of 2. The performance of the algorithm is 83.3%.

Table 1: Results recorded at 25 and 30 fps, sampled at 5-frame intervals

30 - 50 mph roadways, dataset: 102 over-height vehicles									
	Sampling rate @ every x frames	# of positive frames			warning to driver		metrics		
		low	avg	upp	GROUND TRUTH: # of possible warnings per OHV	avg. # of warnings issued to each OHV	recall	precision	false positive
@ 25 fps	1 frame	37	49.5	62	9.9	8.5	100%	85.9%	11.7%
	5 frames	11	21.5	32	4.3	3.6	100%	83.7%	8.5%
	10 frames	7	12	17	2.4	2	100%	83.3%	8.3%
	15 frames	3	7.5	12		*			
@ 30 fps	1 frame	43	57	71	11.4	9.6	100%	84.2%	13.5%
	5 frames	17	28	39	5.6	4.6	100%	82.1%	10.0%
	10 frames	9	16	23	3.2	2.6	100%	81.3%	11.9%
	15 frames	6	7.3	8.5	2.2	0.9	100%	40.9%	20.0%
	20 frames	4	6	8		*			

*Did not meet minimum required number of positive frames

5 DISCUSSION & CONCLUSIONS

This experiment recorded a total of 102 OH vehicles at frame rates of 25 and 30 fps in order to minimise misclassifications due to wind. At these frame rates, we compared sample rates to determine the sufficient number of positive frames to provide accurate warnings to drivers. The results showed that, for 30 - 50 mph roadways, the optimal frame / sample rates are 25 fps sampled every 10 frames. At this optimal setting, the average number of positive frames is 12. If the trigger is set at five consecutive frames, there is a likelihood of at least one warning to the OH driver, while minimizing the number of multiple warnings. As a result, the computing and processing speeds are minimally impaired. The performance of the algorithm is 83.3% and false positive rate of 8.3%.

The precision of the algorithm is affected by the number of multiple warnings given for positive OH instances. Therefore, the ideal number to target is 1 warning per OH vehicle with an allowable tolerance of +1. Sampling rates above 15 frames were discarded, as they did not yield the minimum required number of positive frames for processing. If the trigger is set at a five-consecutive frame minimum, and the lower limit of positive frames is 3, this means that there is a shortage of sufficient frames for processing to meet the five-consecutive frame minimum. Therefore, these sampling rates were not considered further. In addition, although at 30 fps with a sampling rate of every 15 frames the lower limit of the five-consecutive frame minimum was met, this configuration showed poor algorithm performance. This configuration was not able to return at least 1 warning due to the low number of frames, which resulted in poor tracking; more frames are required for sufficient processing. The algorithm was able to minimise the number of misclassifications due to wind by using the behaviours of OH vehicles and windy conditions. The algorithm was able to recall 100% of true positives, and to differentiate motion resulting from OH vehicles from noise (i.e. wind). However, as a result multiple warnings may be given to a single OH vehicle. Future work will examine further constraints to eliminate these multiple warnings.

6 ACKNOWLEDGMENTS

This material is based upon work supported by Career Integration Grants (CIG) - Marie Curie Actions. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the institutes mentioned above.

7 REFERENCES

- Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6), pp. 679–698. <https://doi.org/10.1109/TPAMI.1986.4767851>
- Dai, F., Park, M.-W., Sandidge, M. and Brilakis, I. (2015). A vision-based method for on-road truck height measurement in proactive prevention of collision with overpasses and tunnels. *Automation in Construction*, 50, pp. 29–39. <https://doi.org/10.1016/j.autcon.2014.10.005>
- Harris, C. and Stephens, M. (1988). A combined corner and edge detector. *Proceedings of the Alvey Vision Conference*, pp. 147–151. <https://doi.org/10.5244/C.2.23>
- Nguyen, B. and Brilakis, I. (2016a). Understanding the problem of bridge and tunnel strikes caused by over-height vehicles. *Journal of Transportation Research Procedia*, 14, pp. 3915–3924. <https://doi.org/10.1016/j.trpro.2016.05.481>
- Nguyen, B., Brilakis, I. and Vela, A. P. (2016b). Vision-based over-height vehicle detection. Transportation Research Board 95th Annual Meeting, No. 16-355.
- Nguyen, B., Brilakis, I. and Vela, A. P. (2017). Optimized parameters for over-height vehicle detection under variable weather conditions. *Journal of Computing in Civil Engineering*, in publishing.
- Shao, J., Zhou, S. K. and Chellappa, R. (2010). Robust height estimation of moving objects from uncalibrated videos. *IEEE Transactions on Image Processing*, 19(8), pp. 2221–2232. <https://doi.org/10.1109/TIP.2010.2046368>