RESEARCH PAPER

Integrating bridge influence surface and computer vision for bridge weigh-in-motion in complicated traffic scenarios

Author list

Name	Email	Affiliation
Xudong Jian	jxd_engineer@tongji.edu.cn	1,2
Ye Xia*	yxia@tongji.edu.cn	2,4
Shouwang Sun	sunshouwang@szewec.com	3
Limin Sun	lmsun@tongji.edu.cn	1,2,4

1 State Key Laboratory for Disaster Reduction in Civil Engineering, Tongji University, Shanghai, China.

2 Department of Bridge Engineering, Tongji University, Shanghai, China.

3 Shenzhen Expressway Engineering Consulting Company Limited, Shenzhen, China

4 Shanghai Qizhi Institute, Shanghai, China

Correspondence

*Ye Xia, Building of Bridge Engineering, Tongji University, 1239 Siping Road, 200092, Shanghai, China. Email: yxia@tongji.edu.cn

Funding Information

The research is supported by the National Key R&D Program of China (2019YFB1600702), National Natural Science Foundation of China (Grant No. 51978508), the Science and Technology Commission of Shanghai Municipality (Grant No. 19DZ1203004), and the Technology Cooperation Project of Shanghai Qizhi Institute (Grant No.SYXF0120020109)

ABSTRACT

Complicated traffic scenarios, including random change of vehicles' speed and lane, as well as the simultaneous presence of multiple vehicles on bridge, are main obstacles that prevents bridge weigh-in-motion (BWIM) technique from reliable and accurate application. To tackle the complicated traffic problems of BWIM, this paper develops a novel BWIM method which integrates deep-learning-based computer vision technique and bridge influence surface theory. In this study, bridge strains and traffic videos are recorded synchronously as the data source of BWIM. The computer vision technique is employed to detect and track vehicles and corresponding axles from traffic videos so that spatio-temporal paths of vehicle loads on the bridge can be obtained. Then a novel method is proposed to identify the strain influence surface (SIS) of the bridge structure based on the time-synchronized strain signals and vehicle paths. After the SIS is identified, the axle weight (AW) and gross vehicle weight (GVW) can be identified by integrating the SIS, time-synchronized bridge strain, and vehicle paths. For illustration and verification, the proposed method is applied to identify AW and GVW in scale model experiments, in which the vehicle-bridge system is designed with high fidelity, and various complicated traffic scenarios are simulated. Results confirm that the proposed method contributes to improve the existing BWIM technique with respect to complicated traffic scenarios.

KEYWORDS

bridge weigh-in-motion; bridge influence surface; complicated traffic problem; computer vision; deep learning

1 Introduction

1.1 Motivation and related work

The acquisition of traffic loads on bridges is essential for determining the structural and maintenance requirements of bridges and road pavements. Besides, traffic information, such as traffic flow, vehicle speed, and so on, is of interest in the planning of traffic infrastructure, economic statistics as well as enforcement surveys. As a technique invented to measure traffic information on bridges, the bridge weigh-in-motion (BWIM) technique has received considerable attention for its flexibility, durability and unbiased accuracy in comparison to other weighing methods such as static weigh or pavement-based weigh-in-motion system [1, 2].

During the past four decades, numerous methods for implementing BWIM have appeared, and most of them are built on the static algorithm proposed by Moses [3]. The Moses's algorithm uses the influence line theory, where vehicle weights can be identified from the influence line values at the location of axles and the bridge strain readings. In practice, the bridge strain can be easily measured by strain gauges attached to longitudinal bridge members, while the influence line values at axles' location are more difficult to get. In Moses's pioneering practice of BWIM [4], the vehicle axle spacing and velocity are measured by tape switches placed on pavements so that the axle location can be estimated by assuming the vehicle moves at a constant speed and invariant transverse location. Though field tests had verified the feasibility of the approach, it is concluded that accurate weight identification cannot be realized in complicated traffic scenarios. For instance, Thillainath and Hood [5] observed a variation in the strain readings of up to 30%, which may introduce a 30% error for weighing with Moses's algorithm, during a BWIM calibration practice in which the same vehicle travelled at the same speed, but in different transverse locations. Now that the strain signals are accurately recorded by strain sensors, the accuracy degradation can be attributed to difficulties in accurately determining the influence line value at the location of each axle when vehicles move in a complicated behavior on the bridge.

The accurate location information of axles is the prerequisite for obtaining the corresponding influence value. In order to identify the location of vehicle axles, early BWIM practice made use of traffic sensors such as road tubes or pavement-embedded axle detectors [6]. However, any usage of those sensors would diminish BWIM's advantages over the traditional weigh-in-motion technique. Later new strategies named 'nothing-on-road'(NOR) or 'free of axle detector'(FAD) are investigated to locate axles in aspects of signal post-processing [7-9], shear strain method [10-15], and so forth [16, 17]. In addition, some studies [18-21] paid more attention to mitigating the aforementioned complicated traffic problems for BWIM. Nonetheless, these studies still have difficulty in determining the contributions of individual axles to the global bridge response in various complicated traffic scenarios because none of them is able to continuously and accurately identify the location of every axle at each moment during the vehicle passing the bridge. Therefore, the complicated traffic problems are so far key issues preventing the BWIM technique from a reliable and massive application, and they remain to be thoroughly resolved [1, 2].

In recent years, the continuing progress in computer vision (CV) techniques shed new light on the complicated traffic problem of BWIM research and practice. Previous studies [22-28] have shown that CV can not only recognize vehicle configurations including vehicle type, number, and spacing of axles in a cost-effective manner, but it can accurately identify the spatio-temporal path of every vehicle on the bridge. This advantage is enhanced even further due to the surge of deep learning, which drastically improves the efficiency and robustness in computer visual object detection and recognition [29]. However, those studies do not combine computer vision with BWIM as the vehicle tracks they obtain are not used to identify vehicle weights. Inspired by those studies, Xia and Jian et al. [30, 31] first took advantage of deep-learning-based CV to continuously and accurately locate the vehicles on bridges. Then the vehicles' real-time locations are fused with bridge strains to identify vehicle weights. BWIM results obtained from a field test were encouraging despite the presences of two vehicles, manifesting the feasibility of integrating cost-effective CV technique with BWIM. But the accuracy and efficiency of the computer vision techniques used in those studies remain to be improved. Unacceptable identification errors of vehicle weights still occur occasionally when the vehicles deviate from the traffic lane because the usage of the 1-D influence line is inherently unable to consider the transverse movement of vehicles. Besides, the BWIM algorithm they use to weight vehicles only uses peak values of bridge strain to weigh vehicles, which can lead to the loss of weighing accuracy. In addition, they cannot identify the axle weight of vehicles either. According to Quilligan et al.[18], extending the 1-D influence line to the 2-D influence surface is a logical move to handle that problem. Quilligan [32] also pointed out that using an influence surface of the bridge requires that the transverse and the longitudinal locations of the crossing vehicles be known, but they did not manage to find an automatic and efficient method to locate vehicles in real time.

1.2 Contribution of this work

For the purpose of addressing the remaining complicated traffic problems of the BWIM technique, this paper develops a new BWIM method integrating bridge influence surface theory and deep-learning-based CV technique. The method consists of two major parts, namely, detection of vehicles with CV and identification of vehicle weights with strain influence surface (SIS) of bridge structures. The framework of the proposed method is illustrated in **FIGURE 1**. The visual sensing section is designed to detect, track, and locate vehicles from the traffic videos, and in the meantime the strain sensing section is functioning to measure bridge strains that are time-synchronized with video streaming. The complicated traffic problem of the existing BWIM technique can be effectively resolved with the proposed framework. This is because the computer vision techniques can detect and locate all vehicles on the bridge in a reliable, cost-effective, and real-time manner, and the bridge influence surface theory is thereby enabled to weigh vehicles in complicated traffic scenarios by utilizing both the spatial and temporal strain.

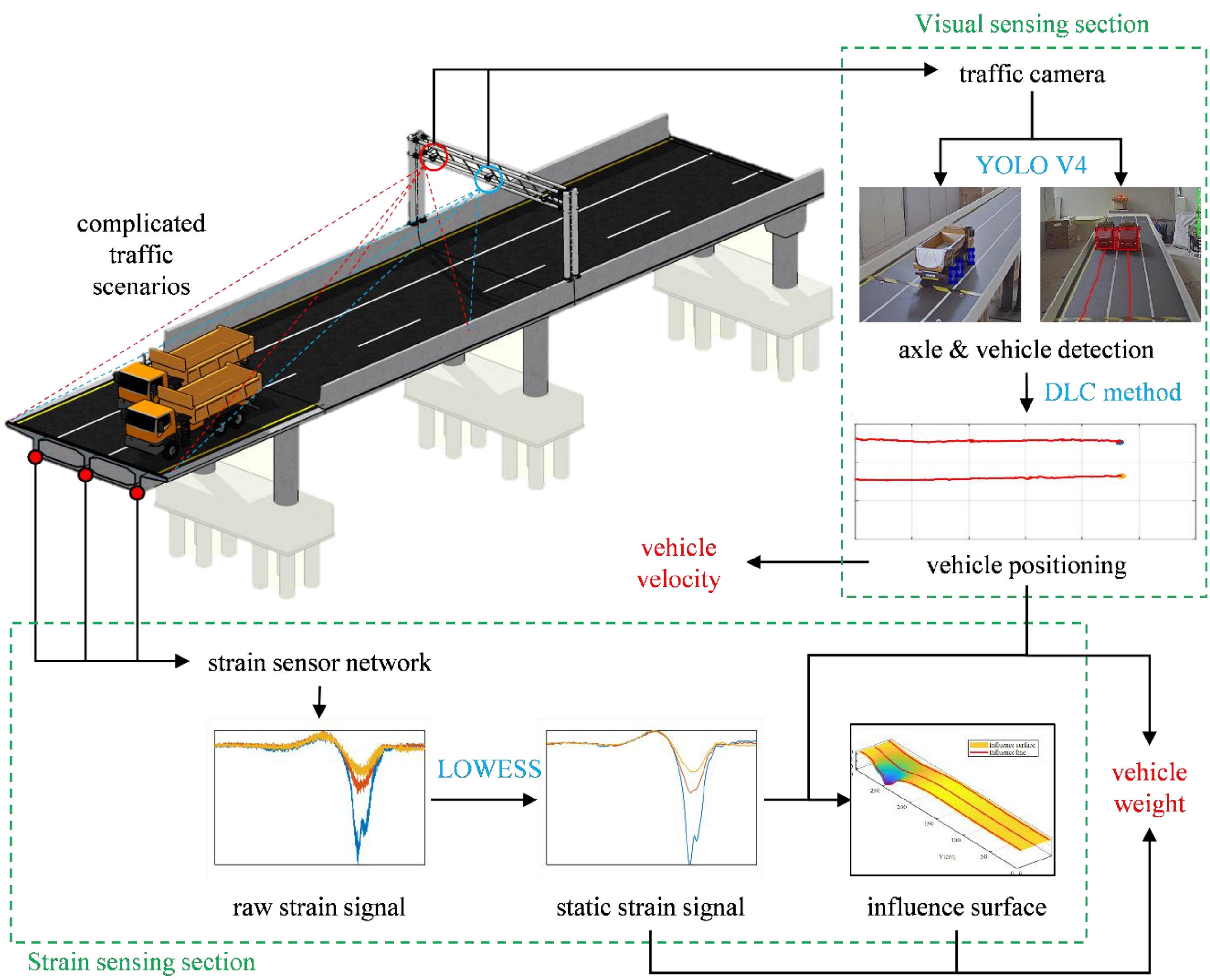


FIGURE 1 Complicated-traffic-problem oriented framework for BWIM application

The main contributions and novelties of this paper lay in the following aspects: 1) A novel BWIM method integrating the newly released deep-learning-based computer vision technique and influence surface theory is proposed. The usage of computer vision enables the proposed method to accuracy locate vehicles in real time, so that the vehicle weights in complicated traffic scenarios that previous BWIM studies cannot handle well can be reliably and accurately identified. 2) A new optimization-based method is proposed to identify the strain influence surface of bridges in a model-free manner, and the method has advantages on robustness and applicability compared with previous studies.

The remainder of this paper is organized as follows: In the section of visual sensing, the introduction of YOLO V4, a recently released CV model based on deep learning, is given. Training strategies of the model and its detection results are discussed so that the coordinate transformation can be introduced for the localization of vehicles and axles on the bridge. In the section of strain sensing, an optimization-based method to construct the SIS of bridge structures is proposed first. Then, a method that exploits both spatial and temporal strain to identify the gross weight of vehicles is put forward. At last, a series of experiments on a vehicle-bridge scale model with high fidelity are carried out to illustrate and evaluate the proposed BWIM method in various traffic scenarios.

2 Visual sensing

2.1 Detecting vehicles and axles with YOLO V4

Computer vision (CV) is a technique that aims to use computers for understanding and automating tasks from digital images or videos like a human visual system. In the field of computer vision, there are several fundamental visual recognition tasks: image classification, object detection, and semantic interpretation. As previously mentioned, one key issue of the BWIM technique is the detection of vehicles. With the rapid development of hardware facilities and software methods, video surveillance has been ubiquitously deployed to monitor the traffic condition of roads and bridges, which paves the way for employing the CV technique to automatically acquire vehicle information such as vehicle position, axle number, and spacing, for BWIM purposes. However, traditional CV methods experience difficulties in getting the desired vehicle information with satisfactory speed, stability, and accuracy, especially when there are complex background, geometric distortion, and illumination variation in traffic videos [33].

Fortunately, the deep learning methods shed new light on the usage of CV for BWIM problems. Over the last few years, deep learning has emerged as a powerful technology, making tremendous progress in many fields, with CV being one of the most prominent cases. Numerous scientific studies and engineering practices have validated that the deep-learning-based CV significantly outperforms the traditional CV in a wide range of object detection and classification tasks. The ambition and concept behind deep-learning-based CV is to mimic the visual cognition behavior of human by establishing computational models of multiple processing layers to learn and represent visual data with multiple levels of abstraction. Therefore, in contrast to hand-crafted descriptors used in traditional CV detectors, deep-learning-based CV generates feature representations from raw pixels to high-level semantic information, which is learned automatically from the training image data. Furthermore, benefiting from the learning ability, deep-learning-based CV can obtain better detection results in complex contexts and large datasets [34].

In this study, an open-source deep-learning-based CV method, YOLO (You Only Look Once) V4, is adopted to detect and recognize vehicles from traffic videos. The newly released YOLO V4 model can be trained by a single GPU conveniently and achieve real-time, high-quality as well as convincing object detection results. The advantages of YOLO V4 come from its data enhancement tricks and optimization of the neural network structure. The detailed theory of YOLO V4 is not the focus of this paper. More details can be found in the literature [35]. The main steps of implementing YOLO V4 for BWIM purposes are illustrated in **FIGURE 2** and summarised as follows:

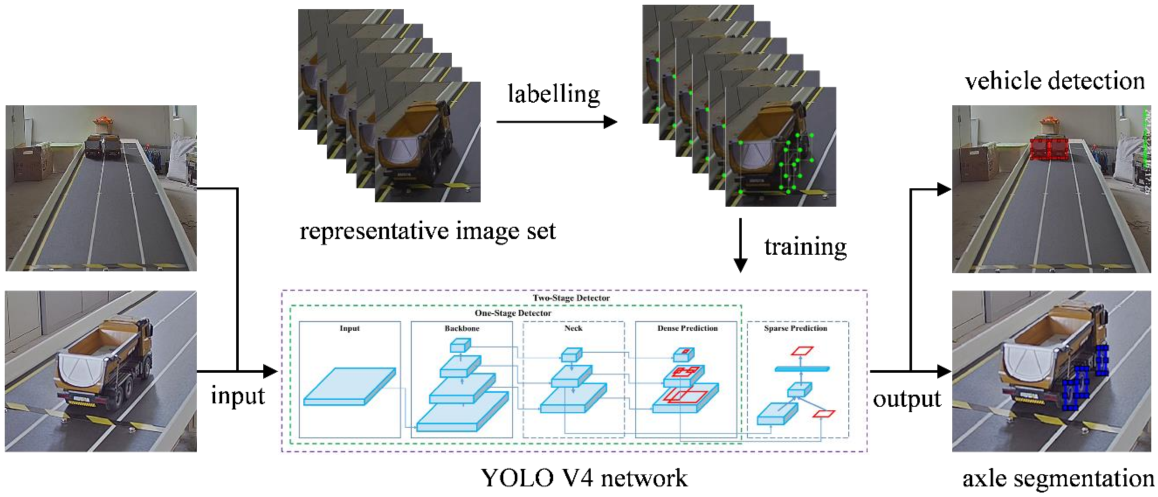


FIGURE 2 Flowchart of implementing the YOLO V4 network

(1) Formation of training data. The YOLO V4 finishes object detection tasks in an end-to-end way, which means only the objects YOLO V4 learns from the training data will be detected. As a consequence, training data in diversity must be prepared at first so that the YOLO V4 network can learn representative features of objects and detect them successfully. The application of BWIM is interested in vehicle position, axle number, and spacing. Accordingly, in the images of traffic videos, rears of vehicles are manually labeled for vehicle localization and wheels are annotated for axle detection. Both the labeling data and 200 images are then delivered to the YOLO V4 model for the training process.

(2) Training and testing. The training process of YOLO V4 is intended to find a set of neural network weights which minimize the error between the detection results and the expected results annotated in the training data. To initiate the training process, training parameters are set at first, including network input size, batch size, learning rate, class number of objects, number of convolution kernels, and number of iterations. Values of the training parameters in this study are listed in **TABLE 1** below. After the parameters are set, the training program starts. In this study, the training process took some 4 hours with a single NVIDIA 1080TI GPU, and the training loss curve is plotted in **FIGURE 3**. Upon the YOLO V4 network is trained well, vehicles and axles in the traffic video can be accurately and stably detected in real-time, regardless of the complicated traffic scenarios. Representative detection results are shown in **FIGURE 2**. As can be seen, the YOLO V4 outputs the detection results in the form of bounding boxes. The dimensions and pixel coordinates of such bounding boxes are collected to locate the position of vehicles further.

TABLE 1 Training parameters

Parameters	Value
Input size	416×416×3
Learning rate	0.001
Batch size	64
Classes of labeled objects	2 (truck and wheel)
Training set / testing set	8/2
Iterations	4000

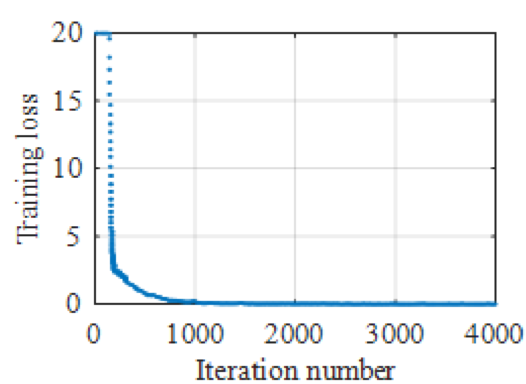


FIGURE 3 Training loss curve

2.2 Locating vehicles on the bridge

The detection results of YOLO V4 are in pixel coordinates format, which cannot be directly used for BWIM purposes because BWIM requires the vehicle's positions on the bridge. Hence a coordinate transformation is needed to transfer the two-dimensional (2D) pixel coordinates of detected vehicles and axles into three-dimensional (3D) coordinates in real space. Considering the disadvantages of the coordinate transformation method used in previous studies [30], this paper introduces a more accurate and convenient one and modifies it in the context of BWIM. Although computer vision scholars can be expected to be familiar with the transformation, this paper includes the brief description to ensure the accessibility to readers who are not familiar with the computer vision.

In order to describe the coordinate transformation mathematically, three coordinate systems are established. They are image coordinate system, camera coordinate system (in this coordinate system, the location of the camera is the coordinate origin), and world coordinate system. The three coordinate systems are described through the pinhole perspective model, as illustrated in **FIGURE 4** below.

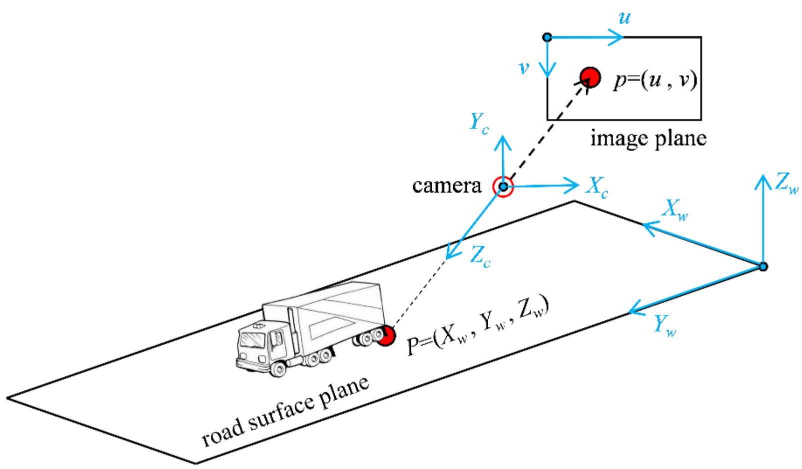


FIGURE 4 Diagram of the coordinate transformation for BWIM purposes

Let a point be denoted as $p=(u,v)$ in the image and the corresponding point be denoted as $P=(X_w,Y_w,Z_w)$ in the world coordinate system. Meantime, point P can be denoted as $P=(X_c,Y_c,Z_c)$ in the camera coordinate system. According to [36, 37], the mathematical relationship between point p and point P can be written as

$$Z_c\begin{bmatrix}u\\v\\1\end{bmatrix}=\begin{bmatrix}m_{11}&m_{12}&m_{13}&m_{14}\\m_{21}&m_{22}&m_{23}&m_{24}\\m_{31}&m_{32}&m_{33}&m_{34}\end{bmatrix}\begin{bmatrix}X_w\\Y_w\\Z_w\\1\end{bmatrix}=\mathbf{M}\begin{bmatrix}X_w\\Y_w\\Z_w\\1\end{bmatrix}\tag{1}$$

where $\mathbf{M}$ is the perspective matrix, m_{ij} is the element of $\mathbf{M}$.

In the context of BWIM, note that the vehicles move on the road surface which is usually an approximate plane, so all points on the plane share a common Z_w. Therefore, it is reasonable to describe the vehicle positions with 2D coordinates on road surface, instead of 3D coordinates in the space. In light of this planar assumption, a point on the road surface can be denoted as $P=(X_w,Y_w)$, and equation (1) turns into

$$Z_c\begin{bmatrix}u\\v\\1\end{bmatrix}=\begin{bmatrix}m_{11}&m_{12}&m_{13}\\m_{21}&m_{22}&m_{23}\\m_{31}&m_{32}&m_{33}\end{bmatrix}\begin{bmatrix}X_w\\Y_w\\1\end{bmatrix}=\mathbf{M}\begin{bmatrix}X_w\\Y_w\\1\end{bmatrix}\tag{2}$$

As can be seen from equation (2), the coordinate transformation can be conducted once upon the perspective matrix $\mathbf{M}$ is determined. To determine the perspective matrix $\mathbf{M}$, the direct linear transformation (DLT) method [38] is adopted in this study on account of its conciseness and operability. Based on equation (1), the DLT method derives two equations

$$\begin{cases}X_w m_{11}+Y_w m_{12}+m_{13}-uX_w m_{31}-uY_w m_{32}=um_{33}\\X_w m_{21}+Y_w m_{22}+m_{23}-vX_w m_{31}-vY_w m_{32}=vm_{33}\end{cases}\tag{3}$$

Given n points of which both the spatial coordinates $\left(X_{wi}, Y_{wi}\right)(i=1,2,\ldots,n)$ on the road surface and the pixel coordinates $\left(u_i, v_i\right)(i=1,2,\ldots,n)$ in the image are known, there will be $2n$ linear equations. Matrix form of the $2n$ equations can be expressed as

$$\begin{bmatrix} X_{w1} & Y_{w1} & 1 & 0 & 0 & 0 & -u_1 X_{w1} & -u_1 Y_{w1} \\ 0 & 0 & 0 & X_{w1} & Y_{w1} & 1 & -v_1 X_{w1} & -v_1 Y_{w1} \\ & & & \cdots & \cdots & \cdots & & \\ & & & \cdots & \cdots & \cdots & & \\ X_{wn} & Y_{wn} & 1 & 0 & 0 & 0 & -u_1 X_{wn} & -u_1 X_{wn} \\ 0 & 0 & 0 & X_{wn} & Y_{wn} & 1 & -v_1 X_{wn} & -v_1 Y_{wn} \end{bmatrix} \times \begin{bmatrix} m_{11} \\ m_{12} \\ m_{13} \\ m_{21} \\ m_{21} \\ m_{23} \\ m_{31} \\ m_{32} \end{bmatrix} = \mathbf{K} \times \mathbf{m} = \begin{bmatrix} u_1 m_{33} \\ v_1 m_{33} \\ \cdots \\ \cdots \\ u_n m_{33} \\ v_n m_{33} \end{bmatrix} = \mathbf{U} \times m_{33} \tag{4}$$

Since equation (4) is linear, it is reasonable to suppose $m_{33}=1$ so that equation (4) can be denoted as

$$\mathbf{Km} = \mathbf{U} \tag{5}$$

Equation (5) contains 8 unknowns that are the elements of vector **m**. In an effort to estimate **m**, the least square method can be used, provided there are at least 4 points of which the coordinates, $\left(X_{wi}, Y_{wi}\right)$ and $\left(u_i, v_i\right)$, are known. The least square solution [39] of **m** is given by

$$\mathbf{m} = \left(\mathbf{K}^T \mathbf{K}\right)^{-1} \mathbf{K}^T \mathbf{U} \tag{6}$$

The perspective matrix **M** is obtained after **m** is estimated. Next, according to equation (2), the coordinate, $\left(X_w, Y_w\right)$, of the vehicle on the road surface can be transformed from the pixel coordinate (u, v), by the equation

$$\left[X_w \quad Y_w \quad 1\right]^T = \mathbf{M}^{-1} Z_c \left[u \quad v \quad 1\right]^T \tag{7}$$

Even though Z_c is unknown, the coordinate transformation can still be completed under the constraint that the last element of the vector $\left[X_w \quad Y_w \quad 1\right]^T$ is 1. In this way, position of vehicles on the bridge and axle spacing required by BWIM are identified with the CV technique. The whole transformation does not require any prior knowledge about the camera, so it is rather practical and convenient.

3 Strain sensing

3.1 Preprocessing of bridge strain signals

As discussed earlier, the BWIM technique is built on the influence line theory [40], which describes the relationship among static load, load position, and static bridge responses. Put another way, the influence line theory is a static concept, so the raw strain data measured by strain sensors are not applicable to the influence line theory because bridges are subject to various dynamic loads in operation. To render the influence line theory applicable, it is necessary to preprocess the raw strain signals so that the static strain can be extracted. According to Jian et al. [30], the raw bridge strain signals contain mainly two components, which are non-vehicle component and vehicle-induced component, and the latter one can be further divided into vehicle-induced dynamic component and vehicle-induced static component. In order to extract the vehicle-induced static strain from the raw strain signal for BWIM purposes, the locally weighted regression (LOWESS) [41, 42] algorithm is employed in this study. **FIGURE 5** illustrates the process of strain preprocessing, and the details of deploying LOWESS can be found in [30]. The extracted static strain is subsequently used for the identification of strain influence surface and vehicle weight.

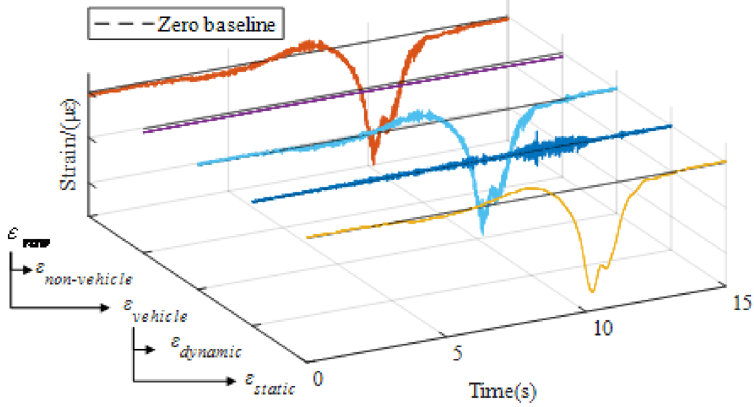


FIGURE 5 Process of extracting vehicle-induced static strain from raw strain signal

3.2 Identifying strain influence surface

Previous studies [18] on BWIM have shown the inability of influence line theory to realize BWIM in complicated traffic scenarios, as the influence line ignores the transverse location of the bridge. Therefore, the strain influence surface (SIS) is used in this study for its capacity to reflect both the longitudinal and transverse location of vehicle loads on the bridge. In an effort to identify the SIS of bridge structures, this study proposes a method that constructs the SIS in two steps. First, the strain influence lines (SIL) at different transverse locations are identified. Then the SIS is constructed on the basis of multiple identified SILs.

The first step starts with the influence line depicted in **FIGURE 6** below.

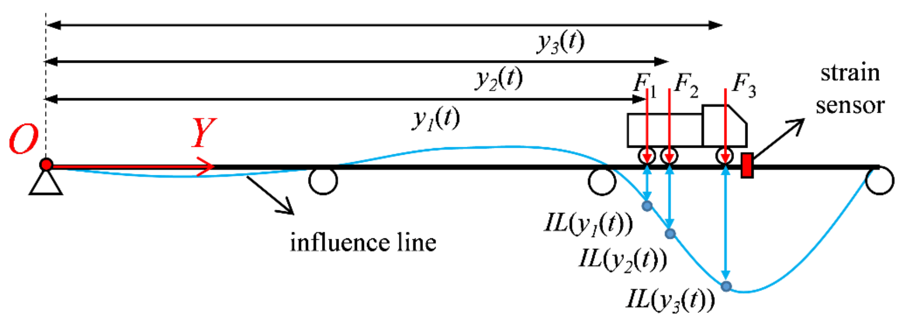


FIGURE 6 Diagram of the influence line

According to [40], when there are vehicles with N axles moving on the bridge, the influence line theory states

$$\varepsilon\left(t_p\right)=\frac{M\left(t_p\right)}{EW}=\frac{\sum_{k=1}^{K}F_k\cdot L_M\left[y_k\left(t_p\right)\right]}{EW}=\sum_{k=1}^{K}F_k\cdot L_\varepsilon\left[y_k\left(t_p\right)\right] \tag{8}$$

where $\varepsilon\left(t_p\right)$ is the vehicle-induced static strain at a discrete sampling point t_p, $M\left(t_p\right)$ is the bending moment of the bridge girder, E is the modulus of elasticity of the bridge material, W is the cross-section modulus of the bridge girder, F_k, k=1,2,3,…,K, is the k^{th} static axle load at the location $y_k\left(t_p\right)$ on the road surface, K is the number of axles on the bridge, $y_k\left(t_p\right)$ are the longitudinal coordinate along the bridge. $L_M\left(y\right)$ and $L_\varepsilon\left(y\right)$ are the value of the bending moment and strain influence line at location y, respectively.

Equation (8) suggests that the SIL is a prerequisite to calculate the axle load. So the first step for implementing BWIM is to identify the SIL by carrying out calibration tests where a vehicle with known axle weight F_k and location $y_k\left(t_p\right)$, which can be identified by CV technique, goes across the bridge. In this way $\varepsilon\left(t_p\right)$, F_k and $y_k\left(t_p\right)$ in equation (8) are all determined, but the $L_\varepsilon\left[y_k\left(t_p\right)\right]$ cannot be readily identified yet. This is because in the single equation (8) the unknowns $L_\varepsilon\left[y_k\left(t_p\right)\right]$ outnumber the equation, leading to an underdetermined problem.

To solve this problem, O'brien, Karoumi and Quilligan [18, 32, 43] proposed a widely accepted method calculating the bridge influence line from direct measurements by using many time scans. Their method, however, has three limitations. First, since there was no method to directly locate vehicles in real time, they estimated the critical vehicle positions by assuming that the vehicle velocity is constant or the vehicle is in the center of the lane. These assumptions are invalid when the vehicle changes its velocity or lane, which may thus make their method inapplicable or lead to a significant error. Second, their method requires to solve a large set of simultaneous equations expressed in matrix form, and the dimensions of the matrices will increase substantially when the bridges have longer spans or the vehicle used for identifying the influence line contains a large number of axles, justifying a need for improved computational efficiency. Although improving computability in the BWIM calibration process is not that urgent since the calibration of IL only needs to be done once for the BWIM system, it is helpful to find a method with better computability. Third, as Ieng [44] points out, this method is not robust enough to provide an accurate IL of the bridge, especially when the bridge strain signals are heavily polluted by noises or dynamic effects of vehicle load.

In order to improve the previous methods for SIL identification, this paper proposes a new method that avoids not only the constant-speed assumption but also large-scale matrices and is robust to noises. **FIGURE 7** illustrates the main steps of the proposed method, and elaborations are given as follows.

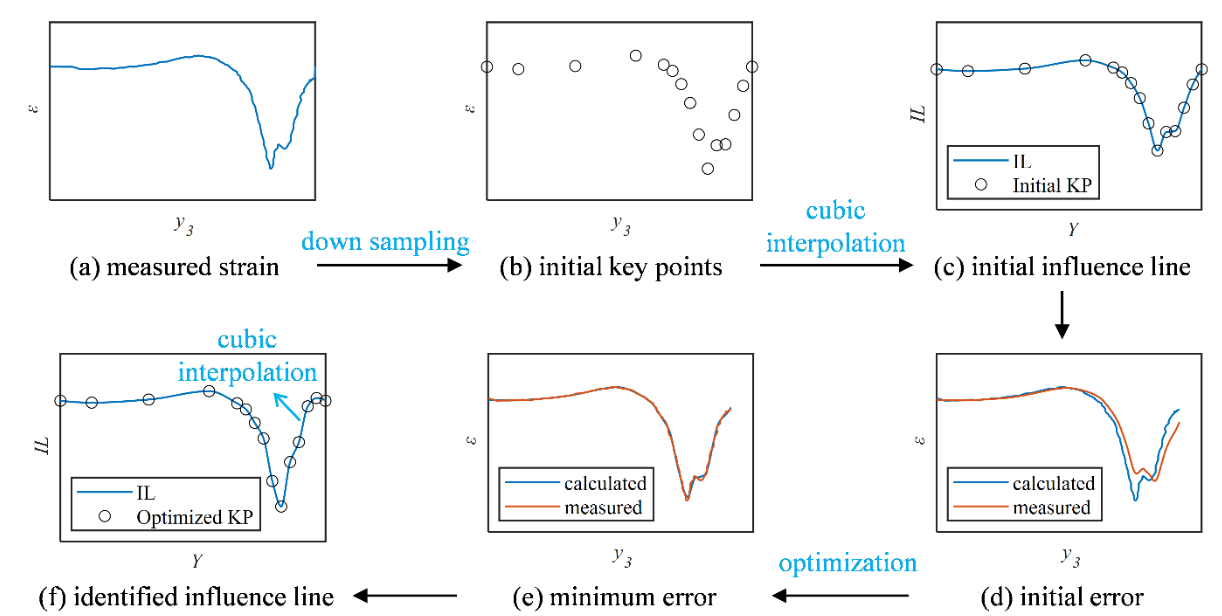


FIGURE 7 Diagram of the identification for the strain influence line

1) Estimating an SIL.

Based on the theory of structures [40], it is easy to roughly estimate the SIL of a beam-like structure by cubic interpolation on a set of key points (KP), $\left[y_{kp}, L_{kp}\right]$, selected from the strain curve in a calibration test. As shown in **FIGURE 7** (a), (b) and (c), when a vehicle goes across a three-span beam like **FIGURE 6** depicts, the SIL of a fixed point on the chosen beam cross-section can be roughly estimated with.

$$L_{kp} = \frac{\varepsilon\left(y_{kp}\right)}{\sum_{k=1}^{K} F_k} \tag{9}$$

It should be noted that L_{kp} in equation (9) is not the final IL used for BWIM. The objective of equation (9) is to provide an initial value for the subsequent optimization problem, so it neglects the axle distances that will be considered later. In addition to that, although equation (9) includes an approximation that introduces errors, the errors will be mathematically minimized through the optimization, which means equation (9) does not affect the weighing accuracy.

2) Defining an error function.

Obviously, the rough estimation for an SIL in the previous step is far from accurate. This inaccuracy is demonstrated by **FIGURE 7** (d) where the strain calculated with equation (8) is compared with the strain measured in the calibration test, and a significant distinction can be observed between those two curves. To quantify this distinction, in this study an error function is defined as

$$f\left(L_{kp}\right)=\sum_{p=1}^{P}\left\{\varepsilon\left(t_p\right)-\sum_{k=1}^{K}F_k\cdot L_{\varepsilon}^{*}\left[y_k\left(t_p\right)\right]\right\}^2 \tag{10}$$

where P is the total number of discrete sampling points, $L_{\varepsilon}^{*}(y)$ is the roughly estimated SIL obtained by interpolating a set of L_{kp}.

3) Optimizing the estimated SIL.

After the error function is defined in equation (10), the identification of SIL turns into a multivariable optimization problem that aims to find a set of key points $\left[y_{kp}, L_{kp}\right]$ to minimize the defined error function, and the quasi newton method (QNM) [45] is adopted herein to solve the optimization problem. The iterative QNM is often used to find the global minimum of a function that is twice-differentiable, as expressed in equation (10). Procedures for deploying QNM are as follows:

i) Choosing and interpolating a set of starting L_{kp}^{0} to estimate an initial $L_{\varepsilon}^{*0}(y)$.The L_{kp}^{0} can be calculated using equation (9).

ii) Calculating search direction by approximating the Hessian matrix $\mathbf{H}^{-1}$. Details of the approximation can be found in [46].

iii) Calculating the change in L_{kp} using the following equation

$$L_{kp}^{j+1}=L_{kp}^{j}-\left[\mathbf{H}^{-1}\right]^{j}\cdot\nabla\left(L_{kp}^{j}\right) \tag{11}$$

iv) Determining if method has converged using convergence criteria that is defined as

$$\nabla\left(L_{kp}^{j+1}\right)<\text{threshold} \tag{12}$$

v) Repeating from step 2 if not converged. Outputting a set of optimized L_{kp} if converged.

By interpolating the optimized L_{kp}, the SIL is identified. **FIGURE 7** (f) gives an example of identified SIL with the QNM, and the strain calculated with the optimized SIL is compared with the strain measured in the calibration test in **FIGURE 7** (e). Since these two curves match very well, it can be concluded that the SIL is successfully identified.

As previously mentioned, this study uses the SIS instead of SIL to implement BWIM, so the next step is to construct the SIS from multiple identified SILs of which the transverse locations are different. Provided a calibration vehicle is arranged to go across the bridge on the left, middle, and right side, the construction can be simply achieved by transverse cubic interpolation on three identified SILs located at the left, middle and right side of the bridge, as shown in **FIGURE 8**.

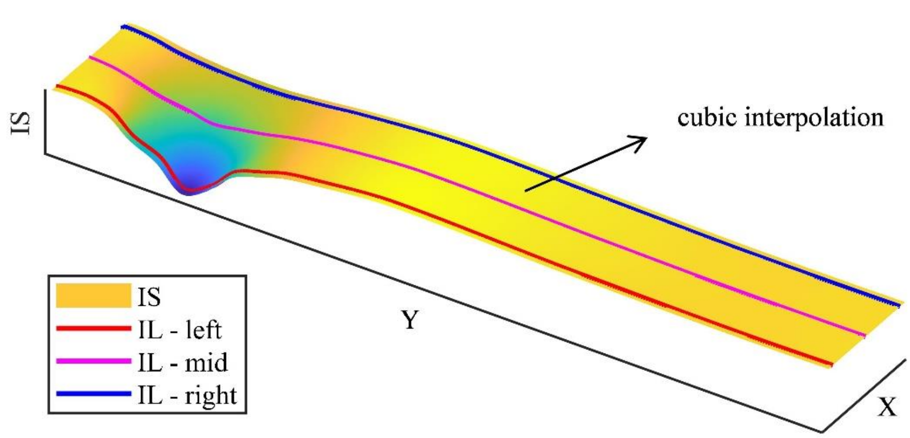


FIGURE 8 Diagram of constructing the strain influence surface

3.3 Identifying vehicle weight

Once the SIS has been identified with calibration tests, the influence surface theory will be applicable so that the bridge strain is written as

$$\varepsilon\left(t_p\right)=\sum_{k=1}^{K} F_k \cdot S_{\varepsilon}\left[x_k\left(t_p\right), y_k\left(t_p\right)\right] \tag{13}$$

where $x_k\left(t_p\right)$ and $y_k\left(t_p\right)$ are the transverse and longitudinal coordinate where the k^{th} axle locates, respectively. $S_{\varepsilon}\left(x,y\right)$ are the value of the SIS at location (x,y). It should be noted that in this paper, $x_k\left(t_p\right)$ and $y_k\left(t_p\right)$ can be accurately identified by the CV technique in a real-time manner no matter how the vehicle moves, but previous studies can only estimate them by assuming that the vehicle moves at a constant speed and a fixed transverse location.

Now that every variable is known except the static axle load F, equation (13) can be used to identify the vehicle weight. This single equation, however, is difficult to solve as the unknowns are more than one. It is thus necessary to transform this underdetermined problem into a determined or overdetermined one, and the transformation can be realized from two dimensions: time and space.

Assuming there are Q strain sensors mounted on the bridge girder, and P discrete strain data points are recorded when all axles move on the bridge. Here, 'time dimension' means that P sampling points of the q^{th} strain sensor are used to form an overdetermined equation set, such that

$$\begin{cases}\varepsilon_q\left(t_1\right)=\sum_{k=1}^{K} F_k \cdot S_{\varepsilon q}\left[x_k\left(t_1\right), y_k\left(t_1\right)\right] \\ \varepsilon_q\left(t_2\right)=\sum_{k=1}^{K} F_k \cdot S_{\varepsilon q}\left[x_k\left(t_2\right), y_k\left(t_2\right)\right] \\ \quad\vdots \\ \varepsilon_q\left(t_P\right)=\sum_{k=1}^{K} F_k \cdot S_{\varepsilon q}\left[x_k\left(t_P\right), y_k\left(t_P\right)\right]\end{cases} \tag{14}$$

where P is the number of strain's discrete sampling points when all axles are on the bridge.

Correspondingly, 'space dimension' means that data of Q strain sensors at the p^{th} sampling moment are used to form an overdetermined equation set that denotes

$$\begin{cases}\varepsilon_1\left(t_p\right)=\sum_{k=1}^{K} F_k \cdot S_{\varepsilon 1}\left[x_k\left(t_p\right), y_k\left(t_p\right)\right] \\ \varepsilon_2\left(t_p\right)=\sum_{k=1}^{K} F_k \cdot S_{\varepsilon 2}\left[x_k\left(t_p\right), y_k\left(t_p\right)\right] \\ \quad\vdots \\ \varepsilon_Q\left(t_p\right)=\sum_{k=1}^{K} F_k \cdot S_{\varepsilon Q}\left[x_k\left(t_p\right), y_k\left(t_p\right)\right]\end{cases} \tag{15}$$

where Q is the number of strain sensors installed on the bridge.

Equation (14) and (15) can be rewrote in the matrix form as

$$\boldsymbol{\varepsilon}_q=\begin{bmatrix}\varepsilon_q\left(t_1\right) \\ \varepsilon_q\left(t_2\right) \\ \vdots \\ \varepsilon_q\left(t_P\right)\end{bmatrix}_{P\times 1}=\begin{bmatrix}S_{\varepsilon q}\left[x_1\left(t_1\right), y_1\left(t_1\right)\right] & S_{\varepsilon q}\left[x_2\left(t_1\right), y_2\left(t_1\right)\right] & \cdots & S_{\varepsilon q}\left[x_K\left(t_1\right), y_K\left(t_1\right)\right] \\ S_{\varepsilon q}\left[x_1\left(t_2\right), y_1\left(t_2\right)\right] & S_{\varepsilon q}\left[x_2\left(t_2\right), y_2\left(t_2\right)\right] & \cdots & S_{\varepsilon q}\left[x_K\left(t_2\right), y_K\left(t_2\right)\right] \\ \vdots & \vdots & \ddots & \vdots \\ S_{\varepsilon q}\left[x_1\left(t_P\right), y_1\left(t_P\right)\right] & S_{\varepsilon q}\left[x_2\left(t_P\right), y_2\left(t_P\right)\right] & \cdots & S_{\varepsilon q}\left[x_K\left(t_P\right), y_K\left(t_P\right)\right]\end{bmatrix}_{P\times K}\times\begin{bmatrix}F_1 \\ F_2 \\ \vdots \\ F_K\end{bmatrix}_{K\times 1}=\mathbf{S}_q\times\mathbf{F} \tag{16}$$

$$\boldsymbol{\varepsilon}\left(t_p\right)=\begin{bmatrix}\varepsilon_1\left(t_p\right)\\ \varepsilon_2\left(t_p\right)\\ \vdots\\ \varepsilon_Q\left(t_p\right)\end{bmatrix}_{Q\times 1}=\begin{bmatrix}S_{\varepsilon 1}\left[x_1\left(t_p\right),y_1\left(t_p\right)\right] & S_{\varepsilon 1}\left[x_2\left(t_p\right),y_2\left(t_p\right)\right] & \cdots & S_{\varepsilon 1}\left[x_K\left(t_p\right),y_K\left(t_p\right)\right]\\ S_{\varepsilon 2}\left[x_1\left(t_p\right),y_1\left(t_p\right)\right] & S_{\varepsilon 1}\left[x_2\left(t_p\right),y_2\left(t_p\right)\right] & \cdots & S_{\varepsilon 2}\left[x_K\left(t_p\right),y_K\left(t_p\right)\right]\\ \vdots & \vdots & \ddots & \vdots\\ S_{\varepsilon Q}\left[x_1\left(t_p\right),y_1\left(t_p\right)\right] & S_{\varepsilon Q}\left[x_2\left(t_p\right),y_2\left(t_p\right)\right] & \cdots & S_{\varepsilon Q}\left[x_K\left(t_p\right),y_K\left(t_p\right)\right]\end{bmatrix}_{Q\times K}\times\begin{bmatrix}F_1\\ F_2\\ \vdots\\ F_K\end{bmatrix}_{K\times 1}=\mathbf{S}\left(t_p\right)\times\mathbf{F} \tag{17}$$

Equation (16) and (17) can be further combined such that

$$\mathbf{S}\times\mathbf{F}=\boldsymbol{\varepsilon} \tag{18}$$

where the strain vector $\boldsymbol{\varepsilon}$, the axle load vector **F**, and the influence surface matrix **S** are denoted as:

$$\boldsymbol{\varepsilon}=\left[\varepsilon_1\left(t_1\right)\ \cdots\ \varepsilon_1\left(t_P\right)\ \ \varepsilon_2\left(t_1\right)\ \cdots\ \varepsilon_2\left(t_P\right)\ \cdots\ \varepsilon_Q\left(t_1\right)\ \cdots\ \varepsilon_Q\left(t_P\right)\right]^T_{1\times(P\times Q)} \tag{19}$$

$$\mathbf{F}=\left[F_1\ \ F_2\ \ \cdots\ \ F_K\right]^T_{1\times K} \tag{20}$$

$$\mathbf{S}=\begin{bmatrix}S_{\varepsilon 1}\left[x_1\left(t_1\right),y_1\left(t_1\right)\right] & S_{\varepsilon 1}\left[x_2\left(t_1\right),y_2\left(t_1\right)\right] & \cdots & S_{\varepsilon 1}\left[x_K\left(t_1\right),y_K\left(t_1\right)\right]\\ \vdots & \vdots & \ddots & \vdots\\ S_{\varepsilon 1}\left[x_1\left(t_P\right),y_1\left(t_P\right)\right] & S_{\varepsilon 1}\left[x_2\left(t_P\right),y_2\left(t_P\right)\right] & \cdots & S_{\varepsilon 1}\left[x_K\left(t_P\right),y_K\left(t_P\right)\right]\\ S_{\varepsilon 2}\left[x_1\left(t_1\right),y_1\left(t_1\right)\right] & S_{\varepsilon 2}\left[x_2\left(t_1\right),y_2\left(t_1\right)\right] & \cdots & S_{\varepsilon 2}\left[x_K\left(t_1\right),y_K\left(t_1\right)\right]\\ \vdots & \vdots & \ddots & \vdots\\ S_{\varepsilon 2}\left[x_1\left(t_P\right),y_1\left(t_P\right)\right] & S_{\varepsilon 2}\left[x_2\left(t_P\right),y_2\left(t_P\right)\right] & \cdots & S_{\varepsilon 2}\left[x_K\left(t_P\right),y_K\left(t_P\right)\right]\\ \vdots & \vdots & \vdots & \vdots\\ S_{\varepsilon Q}\left[x_1\left(t_1\right),y_1\left(t_1\right)\right] & S_{\varepsilon Q}\left[x_2\left(t_1\right),y_2\left(t_1\right)\right] & \cdots & S_{\varepsilon Q}\left[x_K\left(t_1\right),y_K\left(t_1\right)\right]\\ \vdots & \vdots & \ddots & \vdots\\ S_{\varepsilon Q}\left[x_1\left(t_P\right),y_1\left(t_P\right)\right] & S_{\varepsilon Q}\left[x_2\left(t_P\right),y_2\left(t_P\right)\right] & \cdots & S_{\varepsilon Q}\left[x_K\left(t_P\right),y_K\left(t_P\right)\right]\end{bmatrix}_{(P\times Q)\times K} \tag{21}$$

As a result, the axle loads **F** can be obtained by solving the linear matrix equation (18). Note that equation (18) is an overdetermined equation set where the equations significantly outnumber the unknowns, a good identification accuracy can thus be expected when using this equation to solve the axle loads **F**. The traditional least square method can be employed to solve equation (18), under the constraint **F**>0. More details about solving nonnegative linear least-squares problem can be found in [39].

At last, the axle weight AW and gross vehicle weight GVW can be calculated by

$$AW_k=F_k/g \tag{22}$$

$$GVW=\sum_{k=1}^{K}AW_k \tag{23}$$

where g is the gravitational acceleration known generally as 9.8 m/s^2. It is also worth mentioning that, in the International Measurement System, the weight is defined as force (unit: N) instead of mass (unit: kg). However, in this study the vehicle weight is normalized with the acceleration of gravity, and in such case the measurement unit of the vehicle weight is kilogram force (unit: kg). Because this definition of vehicle weight is a widely accepted convention in the field of BWIM.

4 Scale model experiment

4.1 Model description and instrumentation

To investigate the reliability and accuracy of the proposed BWIM method in practice, a scale vehicle-bridge model with high fidelity is utilized. The 1/20-scale bridge model is made of perspex, with static and dynamic similarity to a real concrete box girder bridge located in Shanghai Urban Expressway, China. In order to measure the normal strain of the bridge for BWIM purposes, three resistance-type strain sensors are mounted at the bottom of the third mid-span cross-section. For illustration, the configuration and dimension of the bridge model are shown in **FIGURE 9**.

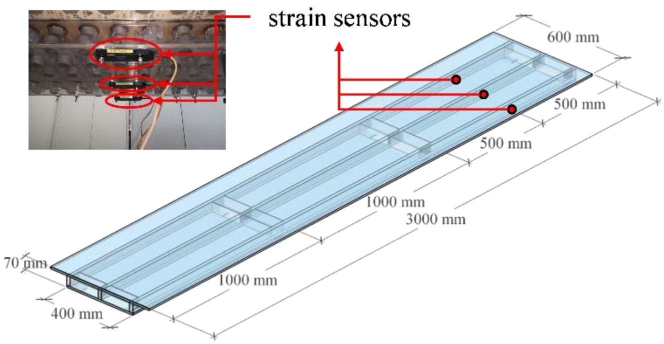


FIGURE 9 Configuration, dimension and instrumentation of the bridge model

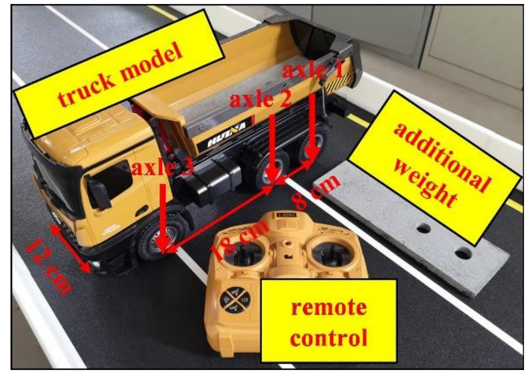


FIGURE 10 Picture of the truck model

In terms of vehicles, remote-control truck models are used in the experiments to simulate vehicle loads. As shown in **FIGURE 10**, the appearance of the three-axle truck model is rather similar to a real dump truck so that the feature-based CV method is applicable. The vehicle is able to forward, back and turn by means of the remote control, at a maximum speed of up to 30 cm/s. The gross weight of the truck is 2.64kg, and there are two additional 1.46 kg weights that can be added onto the vehicle to change the vehicle weight. The gross weight of the scale bridge model is 170 kg, so the mass ratio between the vehicle and the bridge is approximately 2%~3%, which is reasonable and realistic.

FIGURE 11 shows the on-site situation of the experiments. As can be observed, near the scale model two cameras are deployed, among which camera 1 is used to identify the axle configuration of vehicles and camera 2 is used to locate the vehicles. Another reason to use two cameras is to solve the possible obstruction problem between multiple vehicles, and this problem will be elaborated later. The pavement of the bridge model is simulated by abrasive paper, on which pavement lines are painted to simulate various traffic scenarios. The sampling frequency of the three strain sensors is 200 Hz, and the frame rate of the two cameras is 25 frames per second. It should be noticed that the cameras and strain sensors are strictly time-synchronized in the experiments. Otherwise the data fusion between the strain and axle location will be unavailable.

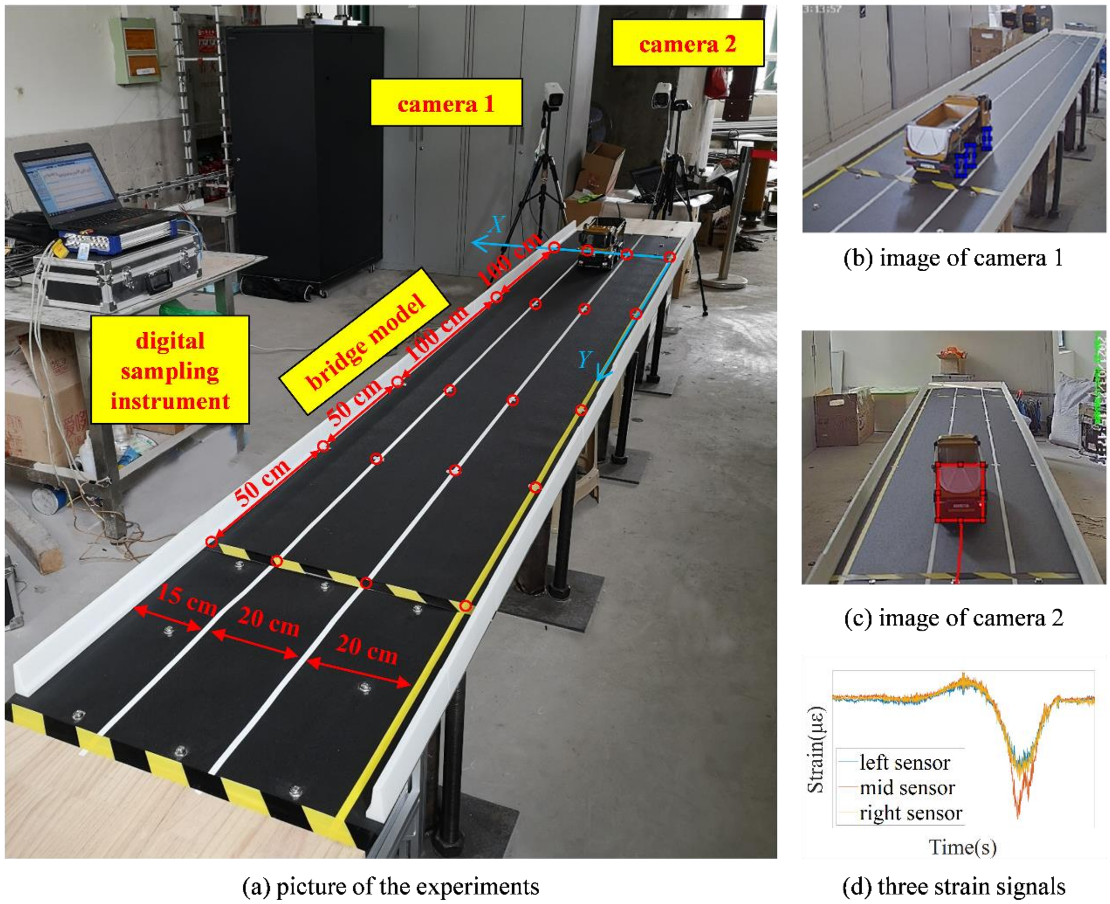
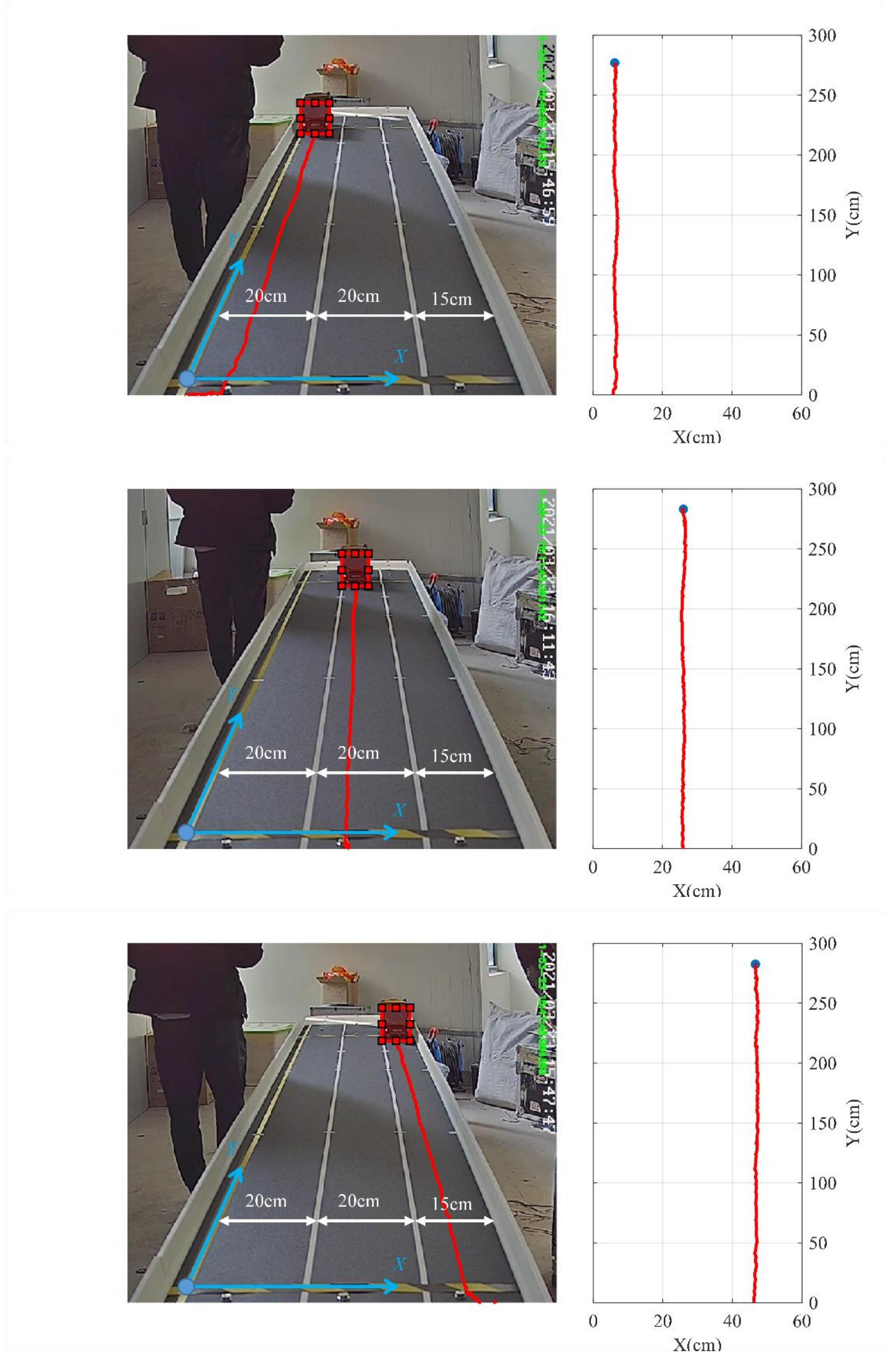


FIGURE 11 Site situation of the experiment

4.2 Results of locating vehicles

In the experiments, tasks of locating axles on the bridge are divided into two stages: i) when vehicles start entering the bridge, detecting axles in the image of camera 1, as shown in **FIGURE 11**(b), and measuring the spacing between each axle. ii) when vehicles fully enter the bridge, detecting the back of vehicles in the image of camera 2, as shown in **FIGURE 11**(c), so as to locate every vehicle on the bridge. The purpose of setting two identification stages is to improve the localization accuracy and reliability by avoiding possible obscuration in images. The theory behind the object detection and length measurement is introduced in the aforementioned visual sensing section. It is worth mentioning that a total of 16 calibration points, of which the locations on the pavement are known, are used in the process of coordinate transformation. Those calibration points are marked on the bridge model in advance and can be observed in images of both two cameras, as shown in **FIGURE 11**(a), (b), and (c).

To evaluate the accuracy of locating vehicles, three calibration experiments are carried out where a single vehicle goes straight on the bridge with its left wheels placed on the pavement line so that the paths of vehicles are known as references. **FIGURE 12** illustrates the three calibration experiments and the identified vehicle paths, and the localization accuracy can be evaluated by comparing the identified path with the pavement line. As can be seen in the following figures, the pixel coordinate of the detection box's bottom midpoint is transferred into the pavement plane coordinate system to locate the rear axle of the vehicle, and the remaining axles are subsequently located by adding the axles spacing identified in the image of camera 2. The three paths' mean lateral distances to the nearest pavement line are 6.45 cm, 26.21 cm, and 46.95 cm, and the variance of the three paths' X coordinates are 0.08 cm, 0.41 cm and 0.07 cm, respectively. Considering the half lateral length of the vehicle is 6cm, it can be concluded the localization accuracy and reliability are rather satisfactory.



(a) calibration experiment 1

(b) calibration experiment 2

(c) calibration experiment 3

FIGURE 12 Diagram of calibration experiments

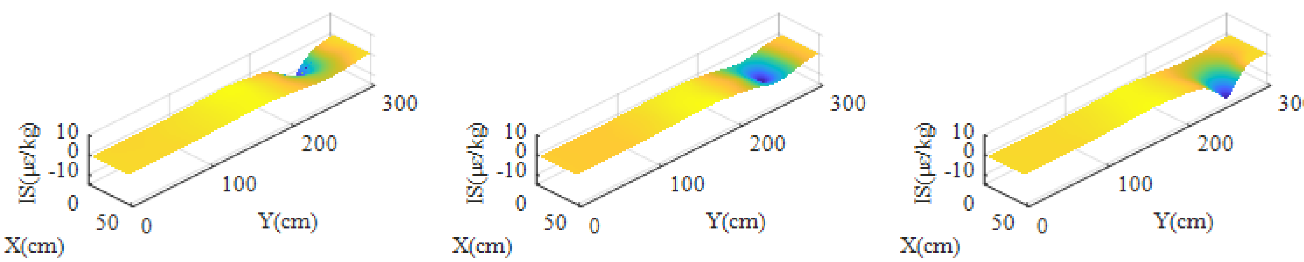
Also worth mentioning is the obstruction problem encountered in this study. For example, when two trucks travel in the same lane, the rear truck may obstruct the view of the front truck in the camera, then using the CV to locate vehicles may be affected. To solve this problem, this study takes two measures. First, when the front truck is partly obstructed by the rear truck, a YOLO model that is specially trained for this case can still detect and track the front truck because of its pleasing robustness as shown in **FIGURE 13** (a). Second, when the front vehicle is completely obstructed by the rear vehicle as shown in **FIGURE 13** (b), even a human cannot detect the front truck, let alone the YOLO model. In this case, it is necessary to deploy another camera with a different angle where the front vehicle is not completely obstructed so that the well-trained YOLO can detect the partly obstructed vehicle as shown in **FIGURE 13** (c). This is one of the reasons why two cameras are used in this study. In addition to the obstruction, illumination change also affects the YOLO in terms of object detection. Similar to the obstruction problem, the YOLO model can still accurately detect all the vehicles regardless of slight illumination change caused by the weather. When there is completely no illumination at night, however, an additional light source or an infrared camera are needed to enable the YOLO model to detect vehicles and axles in the image.

(a) Partial obstruction (b) Complete obstruction (c) Another view of (b), partial obstruction

FIGURE 13 Examples of the obstruction between two vehicles

4.3 Results of identifying strain influence surface

Now that the axle weight, axle spacing, and vehicle path in the three calibration experiments mentioned above have already been known, those experiments are meantime used to identify the SISs belonging to three strain sensors displayed in **FIGURE 9**, with the identification method proposed in section 3.2. The identified SISs are visualized in **FIGURE 14**, which indicates the shapes of these SISs are in good agreement with the influence surface theory despite the fact that they are identified in a model-free approach. Moreover, the quantitative identification accuracy of the SIS will be discussed in the section of identifying vehicle weights.

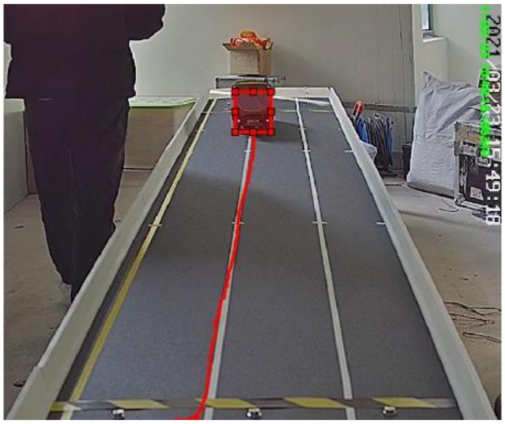


(a) SIS of the left strain sensor (b) SIS of the middle strain sensor (c) SIS of the right strain sensor

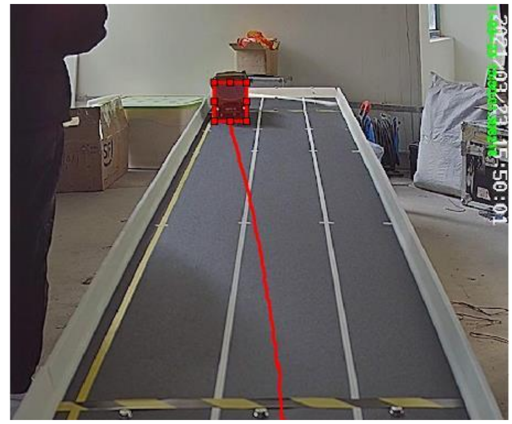
FIGURE 14 Identification results of the strain influence surfaces belonging to three strain sensors

4.4 Results of identifying vehicle weight

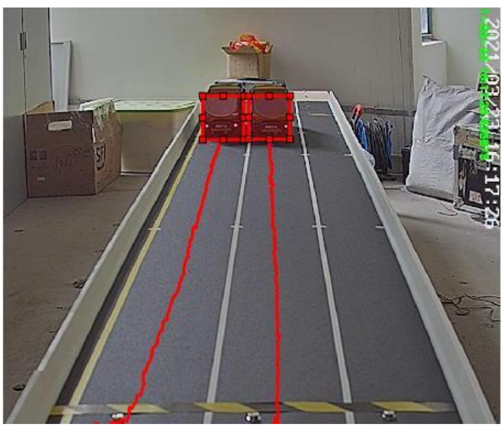
Since the objective of this study is to investigate the accuracy and reliability of the proposed BWIM method in various complicated traffic scenarios, six basic traffic patterns that can be combined into a variety of traffic scenarios are simulated in the experiments. The six basic patterns are ‘single vehicle – go straight’, ‘single vehicle – change lane’, ‘double vehicles – go straight’, ‘double vehicles – change lane’, ‘four vehicles – go straight’, and ‘four vehicles – change lane’, as illustrated in **FIGURE 15**. The ‘double vehicles’ and ‘four vehicles’ scenarios contain two sub-patterns, which are ‘side-by-side’ pattern shown in **FIGURE 15** (a) and ‘one-by-one’ pattern shown in **FIGURE 13** (a).



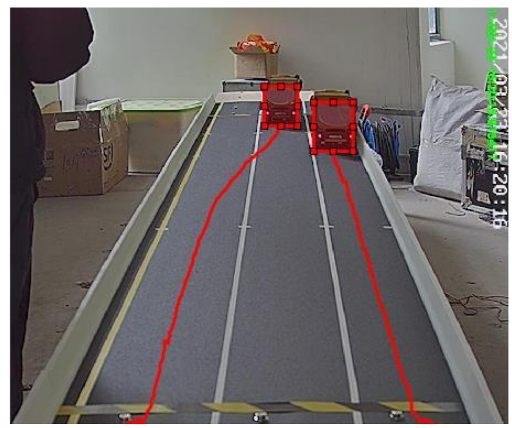
(a) single vehicle – go straight



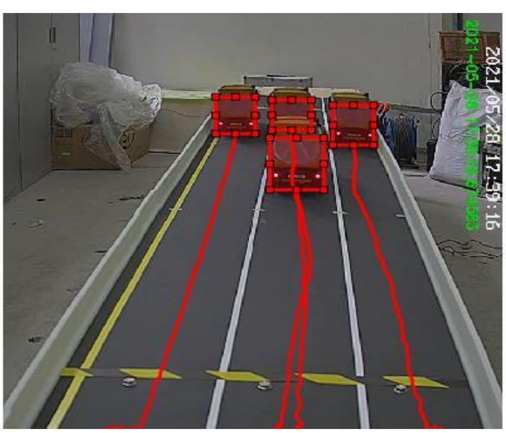
(b) single vehicle – change lane



(c) double vehicles – go straight



(d) double vehicles – change lane



(e) four vehicles – go straight

(f) four vehicles – change lane

FIGURE 15 Diagram of basic traffic patterns

Considering that the weight of each vehicle is different in reality, three different vehicle weights are set by adding additional weights onto the vehicle model, of which the axle weights (AW) and gross vehicle weights (GVW) are shown in **TABLE 2** below. The axle number can be found in **FIGURE 10**, among which axle 1 and axle 2 are regarded as a group because they are too close to weigh separately.

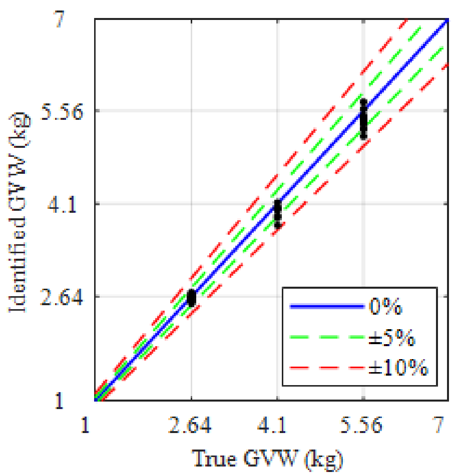
TABLE 2 Weights of vehicles deployed in the experiments

No.	Weight of axle 1 + axle 2 (kg)	Weight of axle 3 (kg)	Gross vehicle weight (kg)
1	1.17	1.47	2.64
2	2.56	1.54	4.10

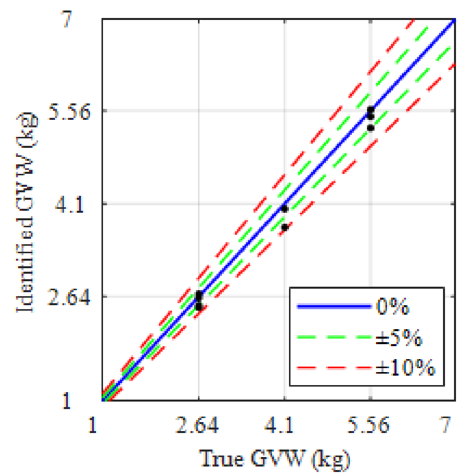
3	3.97	1.59	5.56

Subsequently, 74 experiments are conducted and a total of 116 vehicles are weighed. In each experiment, the path and velocity of vehicles are neither the same nor constant so as to simulate the traffic as real as possible. Experiments in which vehicles stop on the bridge are also conducted to simulate the traffic congestion. The bridge normal bending strains are measured by the three strain sensors with the sampling frequency of 200 Hz, and the vehicle locations along with the axle spacing are identified with the CV technique from the videos recorded by two cameras whose frame rates are 25 frames per second. These time-synchronized measurements are then delivered into the proposed BWIM method to identify the vehicle weights.

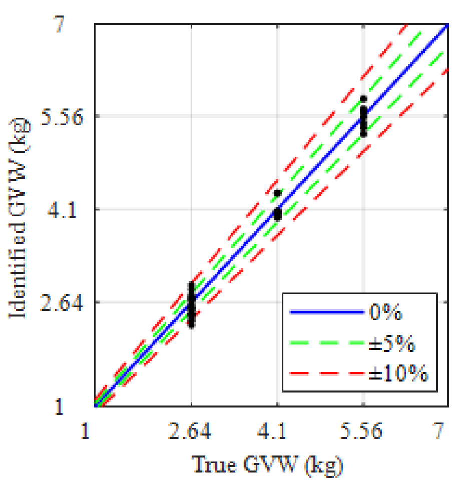
The identification results of GVWs are visualized in **FIGURE 16**, in which each point corresponds to the weighing result of a vehicle. In this figure, the further away the point is from the baseline ±0%, the larger the weighing error is. Four extra reference lines that represent ±5% and ±10% relative error are also plotted in the figure to further evaluate the identification accuracy and reliability. Among the 100 identified GVWs, relative errors of 75 GVWs are less than ±5%, and relative errors of 103 GVWs are less than ±10%.



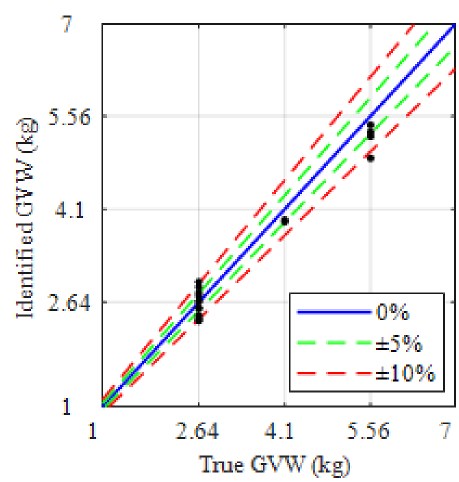
(a) single vehicle – go straight



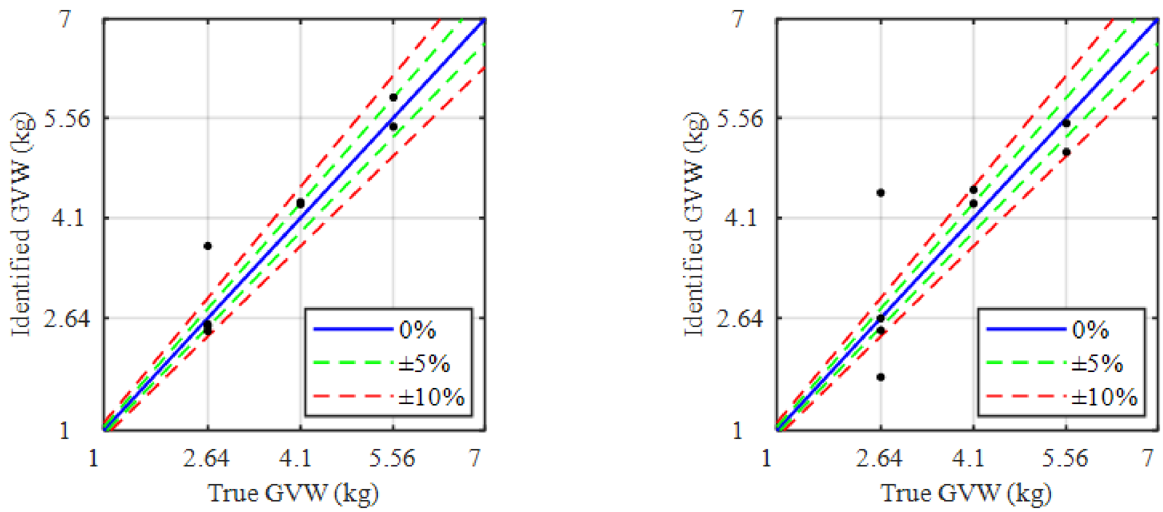
(b) single vehicle – change lane



(c) double vehicles – go straight



(d) double vehicles – change lane



(e) four vehicles – go straight (f) four vehicles – change lane

FIGURE 16 Results of identifying gross vehicle weight

For the purpose of quantitative investigation on the identification accuracy and reliability, relative weighing errors of the identified GVWs are calculated by

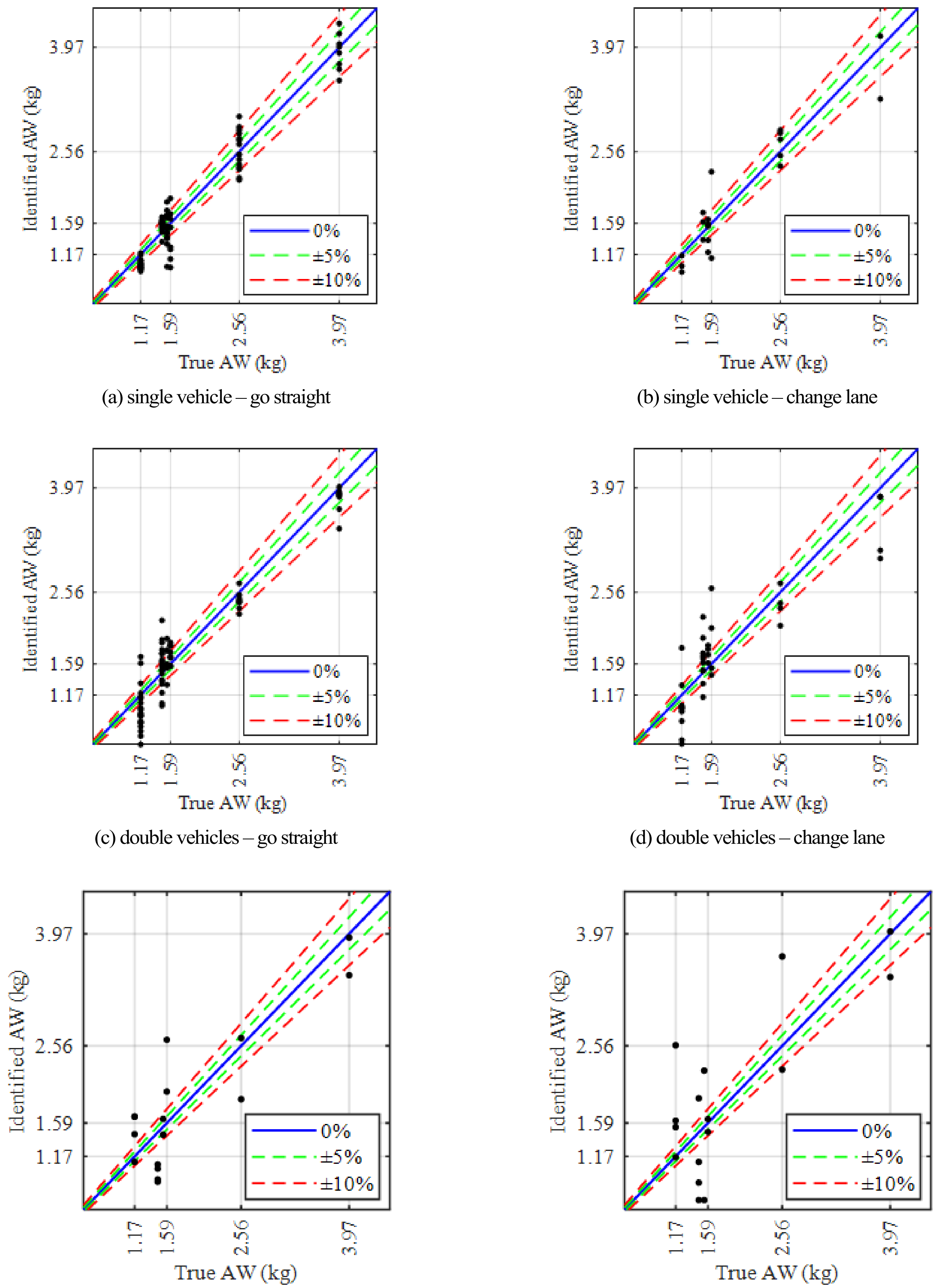
$$error = \frac{\mathrm{GVW}_{\mathrm{identified}} - \mathrm{GVW}_{\mathrm{true}}}{\mathrm{GVW}_{\mathrm{true}}} \times 100\% \tag{24}$$

Statistics of the relative weighing errors in the four traffic scenarios are listed in **TABLE 3** below. The statistics demonstrate a pleasing accuracy and reliability of the BWIM results regardless of the presence of various complicated traffic scenarios. Besides, an accuracy degradation can be found when the traffic scenario gets more complicated, confirming the aforementioned limitation of the BWIM technique when it encounters complicate traffic patterns. Apart from those conclusions about BWIM, the accuracy of weighing vehicles also verifies the successful identification of the bridge structure's strain influence surface.

TABLE 3 Statistics of the relative GVW errors

Scenario	Number of vehicles	Mean (%)	Standard Deviation (%)	Maximum Absolute (%)
single vehicle – go straight	36	-1.63	2.68	8.17
single vehicle – change lane	10	-2.95	3.56	9.07
single vehicle	46	-1.89	2.88	9.07
double vehicles – go straight	36	-1.91	5.44	13.82
double vehicles – change lane	18	-2.57	6.96	13.06
double vehicles	54	-2.13	5.93	13.82
four vehicles – go straight	8	+4.77	15.00	39.77
four vehicles – change lane	8	+4.32	29.20	69.32
four vehicles	16	+4.55	22.43	69.32
overall	116	-1.23	9.52	69.32

Similar to **FIGURE 16**, the identification results of AWs are visualized in **FIGURE 17**, in which each point corresponds to the weighing result of a single axle or a tandem axle. Among the 232 identified AWs (116 single axles and 116 tandem axles), relative errors of 57 AWs are less than ±5%, and relative errors of 108 AWs are less than ±10%.



(a) single vehicle – go straight (b) single vehicle – change lane

(c) double vehicles – go straight (d) double vehicles – change lane

(e) four vehicles – go straight (f) four vehicles – change lane

FIGURE 17 Results of identifying axle weight

For the purpose of quantitative investigation on the identification accuracy and reliability, relative weighing errors of the identified AWs are calculated by

$$error = \frac{\mathrm{AW}_{\mathrm{identified}} - \mathrm{AW}_{\mathrm{true}}}{\mathrm{AW}_{\mathrm{true}}} \times 100\% \tag{25}$$

Statistics of the relative weighing errors in the four traffic scenarios are listed in **TABLE 4** below. Unsurprisingly, just as the existing BWIM methods behave [47], the identification accuracy and reliability of AWs are significantly inferior to those of GVWs. Also can be observed is the phenomenon that the more complicated the traffic scenarios are, the larger the errors are, especially when there are more vehicles on the bridge.

TABLE 4 Statistics of the relative AW errors

Scenario	Number of axles	Mean (%)	Standard Deviation (%)	Maximum Absolute (%)
single vehicle – go straight	72	-2.23	12.51	37.46
single vehicle – change lane	20	-1.00	16.50	43.86
single vehicle	92	-1.96	13.39	43.86
double vehicles – go straight	72	-2.23	19.93	57.27
double vehicles – change lane	36	0.42	26.00	64.43
double vehicles	108	-1.34	22.05	64.43
four vehicles – go straight	16	+1.08	31.87	65.86
four vehicles – change lane	16	+5.87	45.38	119.56
four vehicles	32	+3.48	38.65	119.56
overall	232	-0.93	22.35	119.56

4.5 Error analysis

An error analysis is meaningful for further improvement of the identification performance. For the proposed method, there are three main sources of errors.

1) Errors in the identification of the SIS. As stated previously, the accuracy of BWIM heavily depends on the accuracy of SIS identification. Though the proposed method can identify the SILs of bridges with high accuracy, errors are inevitable when transversely interpolating multiple SILs to construct the SIS. That explains why the errors of vehicle-change-lane cases are larger than those of vehicle-go-straight cases. However, it is difficult to use the optimization-based method to reduce the error, because the calibration vehicle cannot move on the bridge transversely. Moreover, the transverse width of vehicles is ignored in the process of identifying SILs, which may also lead to errors. For example, the actual trucks to be weighed will likely have different width from the IL calibration truck, and this variation of vehicle width will result in errors. Nonetheless, errors caused by the variation of vehicle width are not supposed to be significant, because the variation range of vehicle width, which is usually from 2.0 m to 2.4 m, is rather tight when compared with the width of the bridge. Anyhow, it is worthy to address the errors induced by the SIS in the future study.

2) Number of vehicles. Many studies on BWIM have shown that the accuracy of existing BWIM systems is strongly affected by the number of vehicles present on the bridge during measurement. Because the more the vehicles are on the bridge, the more difficult it is to accurately separate the individual contribution of each axle from the measured overall bridge responses, and the larger the weighing errors will be. Results of this study validate this point once again because the identification accuracy of AWs is much lower than that of GVWs, and a significant accuracy degradation can be observed as the number of vehicles increases. Nevertheless, the identified total weight of all vehicles on the bridge is still accurate since the mean identification errors are small even if there are multiple vehicles.

3) Time lag between the strain signals and the videos. The proposed method requires strictly time-synchronized strain signals and videos, otherwise errors will occur due to the mismatching between the vehicle positions and bridge responses. In the experiments, however, inevitable time lags are observed between the strain signals and the videos because the time computers spend on saving each frame of videos are longer than that of strain signals. Although most of the time lags can be eliminated by post-processing, this study fails to avoid them completely.

At last, concerns may arise over the influence of vehicle velocity on the weighing accuracy, but the experimental results manifest that the variation of vehicle velocity has little influence on the identification accuracy. This is because the vehicle velocity can affect the weighing process by exciting the bridge structure, but the dynamic responses of the bridge are eliminated with the LOWESS algorithm mentioned in Section 3.1

above.

5 Conclusion and future work

In this study, a novel BWIM method is proposed with an extra focus on complicated traffic scenarios. The proposed method is able to identify strain influence surfaces of the bridge structure and vehicle weights by integrating the bridge influence surface theory and computer vision technique. A scale vehicle-bridge model with high fidelity is utilized. Extensive experimental studies are performed to investigate the accuracy and reliability of the proposed method on weighing vehicles in various traffic patterns. The conclusions derived from this study are now summarised as follows:

1) A total of 116 vehicles are weighed with the proposed BWIM method in 74 model experiments (40 single-vehicle cases,30 double-vehicles cases, and 4 four-vehicle cases) where a variety of traffic scenarios, including changing lane and speed, are simulated. Those experiments yield statistically satisfactory weighing results. For the identification of gross vehicle weights, the mean and standard deviation of the overall relative errors are -1.23% and 9.52%, respectively. For the identification of axle weights, the mean and standard deviation of the overall relative errors are -0.93% and 22.35%, respectively. These statistics verify the pleasing accuracy and reliability of the proposed method, but an acceptable degradation of accuracy can still be observed as the traffic scenarios get more and more complicated.

2) In addition to BWIM purposes, the proposed method is also able to identify the strain influence surface of bridge structures in a model-free and practical way, and BWIM results have verified the identification accuracy. The identified bridge influence surface may be further used for bridge condition assessment or other structural identification tasks. Meanwhile, identifying the strain influence surface of bridge structures in a model-based approach is worth further investigation as well, because a model-based strain influence surface may make results less sensitive to errors and help to perform sensitivity analyses and to make considerations on the uniqueness of the solution.

Although the gross vehicle weight can be successfully identified with the proposed method, the accuracy and reliability of identifying individual axle weight remain unsatisfactory. Investigation on improving the identification accuracy of axle weights is left for future studies. Though this study has exhausted all the possible complicated scenarios of two trucks, further studies should also pay attention to traffic scenarios that are very complicated. For example, it is supposed to be rather tough to identify vehicle weights in the scenarios where four vehicles move on the bridge simultaneously with the possibility of changing lanes.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

REFERENCES

1. Lydon, M., et al., *Recent developments in bridge weigh in motion (B-WIM).* Journal of Civil Structural Health Monitoring, 2016. **6**(1): p. 69-81.
2. Yu, Y., C.S. Cai, and L. Deng, *State-of-the-art review on bridge weigh-in-motion technology.* Advances in Structural Engineering, 2016. **19**(9): p. 1514-1530.
3. Fred Moses, M.G., *Instrumentation for weighing truck-in-motion for highway bridge loads*. 1983, Case Western Reserve University: Cleveland.
4. Moses, F., *Weigh-in-motion system using instrumented bridges.* Teansportation Engineering Journal of ASCE, 1979. **105**(3): p. 233-249.
5. Thillainath, S. and R. Hood. *An improved method of CULWAY calibration*. in *Australian Road Research Board (ARRB) Conference, 15th, 1990, Darwin, Northern Territory*. 1990.
6. Snyder, R.E. and F. Moses, *Application of in-motion weighing using instrumented bridges.* Transportation Research Record, 1985. **1048**: p. 83-88.
7. Chatterjee, P., et al., *Wavelet domain analysis for identification of vehicle axles from bridge measurements.* Computers & Structures, 2006. **84**(28): p. 1792-1801.
8. Yu, Y., C. Cai, and L. Deng, *Vehicle axle identification using wavelet analysis of bridge global responses.* Journal of Vibration and Control, 2017. **23**(17): p. 2830-2840.
9. Lechner, B., et al., *A WAVELET-BASED BRIDGE WEIGH-IN-MOTION SYSTEM.* International Journal on Smart Sensing & Intelligent Systems, 2017. **3**(4).
10. Obrien, E.J., et al. *Strategies for Axle Detection in Bridge Weigh-in-Motion Systems*. in *6th International Conference on Weigh-in-Motion*. 2012.
11. Deng, L., et al., *Equivalent Shear Force Method for Detecting the Speed and Axles of Moving Vehicles on Bridges.* Journal of Bridge Engineering, 2018. **23**(8): p. 13.
12. He, W., et al., *Virtual Axle Method for Bridge Weigh-in-Motion Systems Requiring No Axle Detector.* Journal of Bridge Engineering, 2019. **24**(9): p. 04019086.
13. Xiao, Z.-G., et al., *Measurement of truck axle weights by instrumenting longitudinal ribs of orthotropic bridge.* Journal of Bridge Engineering, 2006. **11**(5): p. 526-532.

14. Yamaguchi, E., et al., *Bridge-Weigh-in-Motion by Two-Span Continuous Bridge with Skew and Heavy-Truck Flow in Fukuoka Area, Japan.* Advances in Structural Engineering, 2009. **12**(1): p. 115-125.

15. Kalhori, H., et al., *Nothing-on-Road Axle Detection Strategies in Bridge-Weigh-in-Motion for a Cable-Stayed Bridge: Case Study.* Journal of Bridge Engineering, 2018. **23**(8).

16. Zhou, Y., et al., *Novel methodology for identifying the weight of moving vehicles on bridges using structural response pattern extraction and deep learning algorithms.* Measurement, 2021. **168**: p. 17.

17. Yang, S.K., T.S. Liu, and Y.C. Cheng, *Automatic measurement of payload for heavy vehicles using strain gages.* Measurement, 2008. **41**(5): p. 491-502.

18. Quilligan, M., R. Karoumi, and E.J. Obrien. *Development and Testing of a 2-Dimensional Multi-Vehicle Bridge-WIM Algorithm*. in *International Conference on Weigh-in-motion*. 2002.

19. Žnidarič, A., et al. *Using Strips to Mitigate the Multiple-Presence Problem of BWIM Systems*. in *6th International Conference on Weigh-In-Motion (ICWIM 6)*. 2012.

20. Yu, Y., C.S. Cai, and L. Deng, *Nothing-on-road bridge weigh-in-motion considering the transverse position of the vehicle.* Structure and Infrastructure Engineering, 2018. **14**(8): p. 1108-1122.

21. Chen, S.-Z., G. Wu, and D.-C. Feng, *Development of a bridge weigh-in-motion method considering the presence of multiple vehicles.* Engineering Structures, 2019. **191**: p. 724-739.

22. Chen, Z., et al., *Identification of spatio-temporal distribution of vehicle loads on long-span bridges using computer vision technology.* Structural Control & Health Monitoring, 2016. **23**(3): p. 517-534.

23. Dan, D.H., L.F. Ge, and X.F. Yan, *Identification of moving loads based on the information fusion of weigh-in-motion system and multiple camera machine vision.* Measurement, 2019. **144**: p. 155-166.

24. Zhang, B., L. Zhou, and J. Zhang, *A methodology for obtaining spatiotemporal information of the vehicles on bridges based on computer vision.* Computer‐Aided Civil and Infrastructure Engineering, 2019. **34**(6): p. 471-487.

25. Ge, L., D. Dan, and H. Li, *An accurate and robust monitoring method of full-bridge traffic load distribution based on YOLO-v3 machine vision.* Structural Control and Health Monitoring, 2020. **n/a**(n/a): p. e2636.

26. Khuc, T. and F.N. Catbas, *Structural Identification Using Computer Vision–Based Bridge Health Monitoring.* Journal of Structural Engineering, 2017. **144**(2): p. 04017202.1-04017202.13.

27. Zhou, Y., et al., *Vehicle weight identification system for spatiotemporal load distribution on bridges based on non-contact machine vision technology and deep learning algorithms.* Measurement, 2020. **159**: p. 107801.

28. Ojio, T., et al., *Contactless Bridge Weigh-in-Motion.* Journal of Bridge Engineering, 2016. **21**(7).

29. Voulodimos, A., et al., *Deep learning for computer vision: A brief review.* Computational intelligence and neuroscience, 2018. **2018**.

30. Jian, X., et al., *Traffic Sensing Methodology Combining Influence Line Theory and Computer Vision Techniques for Girder Bridges.* Journal of Sensors, 2019. **2019**: p. 15.

31. Ye Xia, X.J., Bin Yan, Dan Su, *Infrastructure Safety Oriented Traffic Load Monitoring Using Multi-Sensor and Single Camera for Short and Medium Span Bridges.* Remote Sensing, 2019: p. 21.

32. Quilligan, M., *Bridge weigh-in motion: Development of a 2-D multi-vehicle algorithm*, in *Structural Design and Bridge Division*. 2003, Royal Institute of Technology: Stockolm.

33. Wu, X., D. Sahoo, and S.C. Hoi, *Recent advances in deep learning for object detection.* Neurocomputing, 2020.

34. Jiao, L., et al., *A survey of deep learning-based object detection.* IEEE Access, 2019. **7**: p. 128837-128868.

35. Bochkovskiy, A., C.-Y. Wang, and H.-Y.M. Liao, *YOLOv4: Optimal Speed and Accuracy of Object Detection.* arXiv preprint arXiv:2004.10934, 2020.

36. Faugeras, O. and O.A. FAUGERAS, *Three-dimensional computer vision: a geometric viewpoint*. 1993: MIT press.

37. Forsyth, D.A. and J. Ponce, *Computer vision: a modern approach*. 2002: Prentice Hall Professional Technical Reference.

38. Abdel-Aziz, Y.I. and H.M. Karara, *Direct Linear Transformation from Comparator Coordinates into Object Space Coordinates in Close-Range Photogrammetry.* Photogrammetric Engineering & Remote Sensing, 2015. **81**(2): p. 103-107.

39. Lawson, C.L. and R.J. Hanson, *Solving least squares problems*, in *Solving least squares problems*. 1974, Prentice Hall.: Upper Saddle River, NJ. p. 161.

40. Timoshenko, S.P. and D.H. Young, *Theory of structures*. 2nd ed. 1965, New York: McGraw-Hill. 629.

41. Cleveland, W.S., *LOWESS: A Program for Smoothing Scatterplots by Robust Locally Weighted Regression.* American Statistician, 1981. **35**(1): p. 54-54.

42. Cleveland, W.S. and S.J. Devlin, *Locally Weighted Regression: An Approach to Regression Analysis by Local Fitting.* Publications of the American Statistical Association,

1988. **83**(403): p. 596-610.

43. O'Brien, E.J., M. Quilligan, and R. Karoumi, *Calculating an influence line from direct measurements.* Bridge Engineering, Proceedings of the Institution of Civil Engineers, 2006. **159**(BE1): p. 31-34.

44. Ieng, S.-S., *Bridge Influence Line Estimation for Bridge Weigh-in-Motion System.* Journal of Computing in Civil Engineering, 2015. **29**(1).

45. Gill, P.E. and W. Murray, *Quasi-Newton methods for unconstrained optimization.* IMA Journal of Applied Mathematics, 1972. **9**(1): p. 91-108.

46. Fletcher, R., *Practical methods of optimization*. 2013: John Wiley & Sons.

47. O'Brien, E.J., et al., *A regularised solution to the bridge weigh-in-motion equations.* International Journal of Heavy Vehicle Systems, 2009. **16**(3): p. 310-.