VibFormer Vibration Translation for Bridge Live-load Displacement Monitoring

Murtuza Petladwala
Visual Intelligence Research Laboratories
NEC Corporation
Kawasaki, Japan
murtuza@nec.com

Takahiro Kumura
Visual Intelligence Research Laboratories
NEC Corporation
Kawasaki, Japan
t-kumura@nec.com

Chul-Woo Kim
Graduate School of Engineering
Kyoto University
Kyoto, Japan
kim.chulwoo.5u@kyoto-u.ac.jp

***Abstract*— This paper proposes a vibration translation model (termed as VibFormer) for bridge live-load displacement estimation from acceleration signals. The live-load displacement caused by passing vehicles over the bridge structure is a substantial physical quantity to determine structural or traffic properties such as structural damage, vehicle weight, traffic counting, and structural monitoring applications. Theoretically, the double integration of acceleration produces displacement. However, it is challenging because the dynamic elements in the acceleration signal influence the static component of an integrated signal due to time-critical integration boundaries that result in the accumulation of non-zero initial limits. In addition, continuous vehicle passages distort the integration results even if strict boundary conditions are applied. To overcome these challenges, we propose to split the vibration signal's static and dynamic components into each frequency band and use a transformer model to estimate the bridge displacement given acceleration as the input signal. We performed a field experiment on a bridge structure to measure acceleration and displacement signals and evaluate the proposed method. The results show that the proposed method achieved a 91% correlation between the actual and estimated displacements.**

Keywords*— Numerical integration, time-series transformer, live-load displacement, structural health monitoring.

I. INTRODUCTION

Bridge live-load displacements are critical indicators for bridge safety and design managers worldwide [1, 2]. Most bridges, specifically short- and medium-span bridges, are subject to the dynamic load caused by massive moving vehicles. These frequent occurrences of dynamic responses at relatively large amplitudes will cause fatigue damage [3] to the bridges, which can lead to the collapse of the whole structure. Thus live-load displacement monitoring becomes an essential aspect of the functionality and safety of bridges. In addition, traffic monitoring applications like vehicle weight-in-motion systems [4] and traffic counting [5] use live-load displacements.

Simultaneous measurements of the displacement response at multiple points are required to obtain the bridge live-load displacement. The measurement devices generally used are laser Doppler vibrometers [6] and linear variable differential transducers [7] that can measure the bridge displacement precisely with a fixed reference point for each installation at multiple bridge points. However, it is generally difficult to obtain a fixed reference point on an in-service bridge [8]. To overcome this problem, [9-11] proposed vision-based systems suitable for determining live-load displacements. However, vision-based systems also require height and camera angle calibration, which is often time-consuming and impractical in an in-service bridge. The alternative to the abovementioned methods, reference-free estimation using accelerometers [4, 12-14] and strain gauges [15-17], has been proposed. In [17], a feed-forward neural network predicts displacement signals; however, it requires multiple strain gauge sensors. Due to installation difficulties in strain gauges, accelerometers are preferred, which can be easily mounted to magnetic plates or brackets and have great potential to measure the responses even in in-service civil infrastructures [18-20].

Theoretically, the double integration of acceleration produces displacement. However, the dynamic elements in the acceleration influence the static components in an integrated signal due to time-critical integration boundaries that result in the accumulation of noise and non-zero initial limit, overall reducing the accuracy of the integrated displacement [21]. Furthermore, using the additional sensors to extract the strict boundary conditions, the passage of continuous vehicles contaminates the signal that distorts the integration results. Numerous methods in literature [4, 12-16] determine the initial integration limits. The displacement determined from the measured acceleration showed higher similarity with actual displacement in [14], which proposed a free vibration method (FVM) that estimates the bridge displacement by using the time instances extracted from free vibrations of the bridge as integration boundaries. However, method performance highly depends on the efficiency of selecting time instances that require additional sensor installation.

In this paper, we propose a vibration translation method termed VibFormer, a transformer-based [22] approach to estimate the bridge live-load displacement from a single accelerometer response. Theoretically, double integration of acceleration strongly correlates with actual displacement, which means an inherent relationship between their static and dynamic components [23]. For this reason, we propose to split the static and dynamic components of both the signals into each frequency band and train a transformer model by providing acceleration and displacement to the encoder and decoder network, respectively. The transformer model can efficiently capture long-range dependencies in the time-series sequences compared to the convolutional networks [19]. We performed a real-world bridge experiment on a national highway in Japan to measure acceleration and displacement signals and prepared single and continuous vehicle event datasets by reference to camera recordings. The evaluation of our proposed method on this experiment dataset proved the effectiveness of our approach, even in the continuous vehicle passages.

II. Related Work

A. Free vibration method (FVM)

FVM determines the displacement of an in-service bridge from its acceleration based on its free and forced vibration regions [14]. At first, the initial and terminal limits for the integration are determined, assuming that before vehicle entry and after vehicle exit, the bridge is vibrating with sinusoidal oscillations about the zero-axis at its free vibration frequency. Second, the forced acceleration component of the bridge is numerically double integrated to obtain the forced displacement known as live-load on the bridge. Finally, the displacement curve is determined by summing the free vibration displacement and the forced displacement after subtracting the drift component from the integration result. The FVM requires multiple sensor installations to extract vehicle entry and exit time instances located at bridge edges, as FVM performance highly depends on integration boundaries. As shown in Fig. 1., the integration-based FVM result for a single vehicle is distorted from the actual displacement signal. The dynamic bias shift in the integrated signal occurs from the accumulation of non-zero boundaries and the inaccurate selection of time instances during the integration operation. Furthermore, the FVM performance on continuous vehicle passages degrades due to difficulty finding each vehicle's free vibration region.

B. Transformer model

The transformer model first proposed in [22] consists of an encoder-decoder structure. Both the encoder and decoder networks are composed of multiple identical blocks. Each encoder block consists of a multi-head self-attention module and a position-wise feed-forward network. The decoder block places cross-attention between the multi-head self-attention module and the position-wise feed-forward network. The three modules, absolute positional encoding, multi-head attention, and position-wise feed-forward network, are explained as follows.

The first module, i.e., the absolute positional encoding layer, models the sequence information that describes the position of a value in that sequence, where each position is assigned a unique representation. For each position index t in time-series, the encoding vector of size d_{model} is given by

$$PE(t,2i) = \sin\left(t/10000^{2i/d_{model}}\right)$$
$$PE(t,2i+1) = cos\left(t/10000^{2i/d_{model}}\right) \tag{1}$$

where i is the dimension for each input position index t. The second module, i.e., multi-head attention, works as an attention mechanism with Query-Key-Value (QKV) model. The scaled dot-product attention is given by

$$Attention(Q,K,V) = softmax\left(\frac{QK^T}{\sqrt{D_k}}\right)V \tag{2}$$

where queries $Q \in R^{N\times D_k}$, keys $K \in R^{M\times D_k}$, values $V \in R^{M\times D_v}$ and N, M denote the lengths of queries and keys, D_k, D_v denote the dimension of keys and values, respectively. The H different attention outputs are concatenated and linearly projected into the required dimension as

$$MultiHeadAttn(Q,K,V) = Concat(head_1, \ldots, head_H)W^0 \tag{3}$$

where $head_i = Attention\left(QW_i^Q, KW_i^K, VW_i^V\right)$. The third module, i.e., position-wise feed-forward network, is a fully connected layer expressed as

$$FFN(H') = ReLU(H'W^1 + b^1)W^2 + b^2 \tag{4}$$

where H' is output of previous layer of dimension d_{model}, $W^1 \in R^{d_{model}\times d_{ff}}$, $W^2 \in R^{d_{ff}\times d_{model}}$, $b^1 \in R^{d_{ff}}$, $b^2 \in R^{d_{model}}$ are trainable parameters. In addition, the Layer Normalization module is placed after each module, i.e.,

$$H' = LayerNorm(SelfAttn(X) + X) \tag{5}$$

$$H = LayerNorm(FFN(H') + H') \tag{6}$$

where $SelfAttn(.)$ denotes self-attention module and $LayerNorm(.)$ denotes the layer normalization operation.

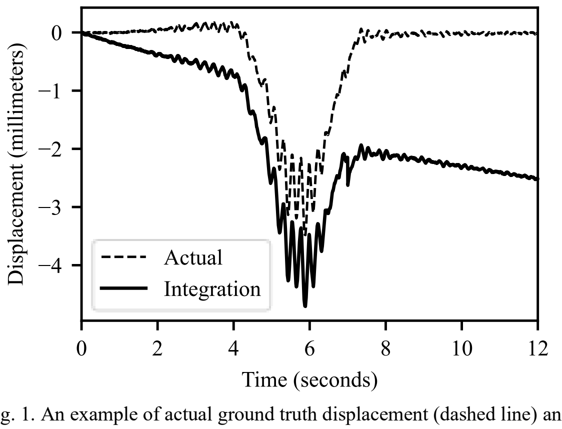

Fig. 1. An example of actual ground truth displacement (dashed line) and displacement obtained from acceleration signal by FVM (solid line).

The time-series modeling for classification [24, 25], anomaly detection [26, 27], and forecasting [28-31] tasks apply the transformer approaches. Time-series forecasting is an essential application of time-series analysis and motivation of our vibration translation method. Reference [28] proposed LogSparse transformer by using causal convolutions to generate queries and keys in the self-attention layer that reduced computational complexity. Another simple seasonal trend decomposition architecture with an auto-correlation mechanism was proposed in [30] as an attention module, which measures the time-delay similarity between inputs and aggregates the top-k similar sub-series to produce the output with reduced complexity. On the contrary, FEDformer model in [31] proposed to apply attention operation in the frequency domain with Fourier and wavelet transform. It achieved a linear complexity by randomly selecting a fixed-size subset of frequency. These transformer models have shown superior results in time-series forecasting applications. Another aspect of signal modeling is signal translation application [32-34], where an input signal translates to another signal or text. Reference [34] proposed WaveTransformer focusing on learning long-term temporal and time-frequency information from audio and expressing it into text using the transformer model. These transformer models can learn long-term sequence complexity better than RNN and CNN models, as reported in [22]. So, the proposed method includes a transformer model to translate one form of vibration to another form by a single time series, explained in the next section.

III. Proposed Method

A combination of signal processing and deep learning techniques is proposed here as a vibration translation method. The proposed method consists of two main parts, i.e., vibration feature extraction and translation. The vibration feature extraction splits two vibration signals into each frequency band, its static and dynamic components. The vibration translation trains a transformer model with features that can translate one form of vibration into another. Fig. 2. illustrates a combined block diagram of training and estimation phase data flow, including two parts of the proposed method. For simplicity, acceleration to displacement translation is used here for the explanation.

A. Vibration feature extraction

In vibration engineering, three types of vibration, displacement, velocity, and acceleration, are related by differential and integral operation, and their frequencies relate to sine or cosine functions that satisfy the differential equations for simple motions of a bridge structure as expressed in (7) to (9).

$$x = A\cos(\omega t + \varphi) \tag{7}$$

where, x=displacement, A=amplitude, ω=frequency, t=time and φ=phase. The time derivative of displacement is velocity,

$$v = \frac{dx}{dt} = -\omega A\sin(\omega t + \varphi) \tag{8}$$

where v=velocity. The double derivative of displacement is acceleration,

$$a = \frac{d^2x}{dt^2} = -\omega^2 A\cos(\omega t + \varphi) \tag{9}$$

where a=acceleration. From (9), the double integration of acceleration will produce back to displacement, which means a strong correlation between integrated acceleration and the actual displacement. Thus, a strong correlation between static and dynamic frequency components. The feature extraction part assumes that a vibration signal $y(t)$ can be expressed as Fourier expansion, as stated in (10), that consists of frequency and phase information from zero to Nyquist frequency obtained by the FFT method,

$$y(t) = A_i + \sum_{n=0}^{f_s/2} A_n \sin(2\pi f_n t + \varphi_n) \tag{10}$$

where, A_i =initial amplitude, f_s =sampling frequency, f_n=signal frequency, t=time and φ_n=phase and n=frequency number up to Nyquist frequency. Equation (11) shows the split version of (10) representing each frequency component,

$$\begin{gathered} y(t) = A_0\sin(2\pi f_0 t + \varphi_0) + \\ A_1\sin(2\pi f_1 t + \varphi_1) + \ldots + A_n\sin(2\pi f_n t + \varphi_n) \end{gathered} \tag{11}$$

where, n =number of frequencies. Our proposed method inverses each FFT component independently to obtain its corresponding time series and concatenates to form an n-dimensional matrix representing static and dynamic vibrations. After the model estimations, these vibration features can be reconstructed to their original 1-dimensional time series by a simple sum operation.

B. Vibration translation

We present vibration translation as a supervised machine learning task, given a first vibration signal X as input and the second vibration signal as output Y. Each data point in X and Y are a vector containing vibration features. The vibration translation part follows the original Transformer architecture [22] that consists of encoder and decoder blocks, as shown in Fig. 3. The encoder block is composed of a linear input layer, a positional encoding layer, and a stack of six identical encoder layers. The input layer maps the input time series n-dimensional features to a d_{model} dimension vector through a fully connected network. The Positional encoding uses sine and cosine functions to encode sequential information and applies element-wise addition of the input vector with a positional encoding vector and feed to six encoder layers. Each encoder layer consists of a self-attention sub-layer and a fully connected feed-forward sub-layer, followed by a normalization layer after each sub-layer. The encoder block generates a d_{model}-dimensional vector to input to the decoder block. The decoder block comprises a linear input layer, a positional encoding layer, six identical decoder layers, and an output layer. The input layer maps the decoder input to a

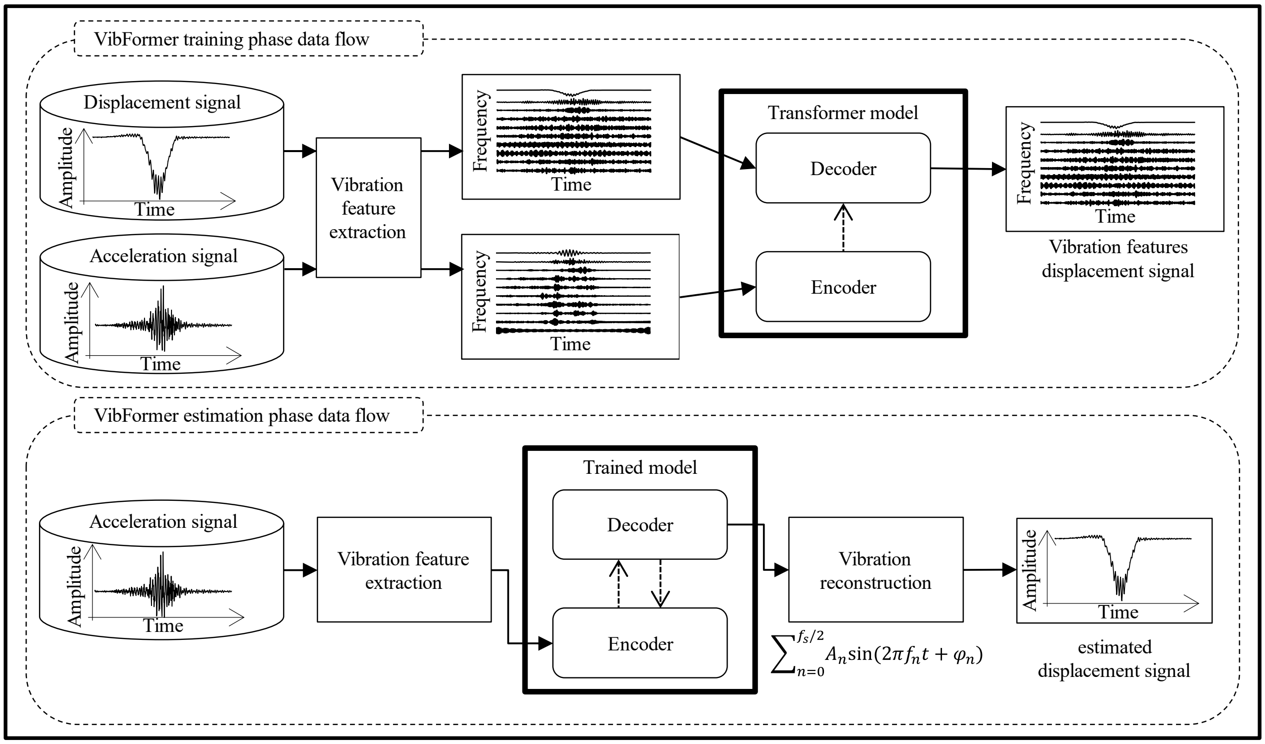
VibFormer training phase data flow
Displacement signal
Amplitude
Time
Acceleration signal
Amplitude
Time
Vibration feature extraction
Frequency
Time
Frequency
Time
Transformer model
Decoder
Encoder
Frequency
Time
Vibration features displacement signal
VibFormer estimation phase data flow
Acceleration signal
Amplitude
Time
Vibration feature extraction
Trained model
Decoder
Encoder
Vibration reconstruction
$\sum_{n=0}^{f_s/2} A_n \sin(2\pi f_n t + \varphi_n)$
Amplitude
Time
estimated displacement signal

Fig. 2. Block diagram of the proposed VibFormer method with the data flow during training and estimation phase.

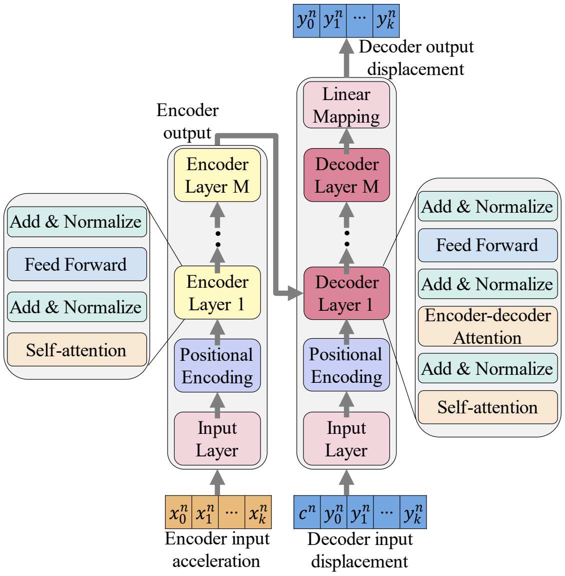

Fig. 3. Architecture of Transformer-based vibration translation model.

d_{model}-dimensional vector followed by positional encoding layer. The decoder layers have two sub-layers like encoder layers and insert a third sub-layer to apply self-attention mechanisms over the encoder output. Finally, there is an output layer that maps the output of the last decoder layer to the target n-dimensional time-series features. A look-ahead mask approach is used here with a one-position shift between the decoder input and output. For this reason, a constant value is inserted at the initial time instant of each decoder input vector to ensure the time-series translation follows the same time sequence as the encoder input.

IV. Evaluations

A. Dataset processing and model training

A bridge experiment was carried out on a national highway in Japan to measure acceleration and displacement. We selected the event dataset by counting the number of peak displacements greater than 2mm, 20% of the max peak. We sliced 500 single and 275 continuous vehicle events into 10- and 20-second sequences, where single events consist of a single peak and continuous events more than one peak. This evaluation shows a comparison between the FVM method and the VibFormer. We selected the integration boundaries in FVM by obtaining the time instant of zero-point in displacement and then mapped it to the acceleration. Secondly, for VibFormer training, we concatenated single events and used a fixed-length sliding time window of 250ms to construct 16k samples of X, Y pairs. Before applying the sliding window, we perform max normalization on all the data with maximum value of training dataset and apply standardization after obtaining n-dimensional vibration features, N=20 in this evaluation. The training and evaluation dataset ratio was 7:3.

The VibFormer model is trained to estimate the corresponding sequence from its input sequence, where each sequence had 20Hz sampling frequency and 8 seconds long. As shown in Fig. 3., given the encoder input $x_0^n, x_1^n, \dots, x_k^n$, and the decoder input $c^n y_0^n, y_1^n, \dots, y_k^n$, the decoder aims to output $y_0^n, y_1^n, \dots, y_k^n$, where k=160, time samples, and, c=1, a constant used in this evaluation. We used the Adam optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.98$, $\epsilon = 10^{-9}$ and a custom learning rate as in [22] with a minibatch of size 32. Each sub-layer of encoder-decoder blocks used a dropout rate of 0.2. Since the signal translation task is similar to a regression problem, the root mean square error is used as the loss function in the model training. Finally, a greedy decoding method generates the target displacement signals in the model estimation phase.

B. Results

Two metrics, including cross-correlation coefficient and mean-absolute errors (MAE), are computed to measure the performance between the actual and estimated displacement. The cross-correlation coefficient shows the similarity between the time series, indicating better performance with a larger value. This evaluation analyzes the efficiency of estimating the long-term complex structure in time series by cross-correlation coefficient, which can represent the overall similarity between actual and estimated displacement. The second metric, MAE shows the error between paired observations. A lower value indicates better performance. On applying acceleration to displacement translation, we focused on the efficiency of estimated bridge displacement values, especially the maximum deflection point. For each sequence, the maximum displacement is selected, and MAE is calculated.

The VibFormer model performance, i.e., the median of the cross-correlation coefficient and MAE on the training dataset, is 98% and 7%, respectively. Table I summarizes the median of both metrics for existing FVM and the proposed VibFormer method, calculated from the evaluation dataset not shown in the training phase. The comparison suggests that the VibFormer method outperformed FVM for correlation and MAE metrics. The cross-correlation coefficient of the VibFormer method for single events is slightly higher than FVM, although, in the continuous event category, it is a drastic improvement. For MAE, the VibFormer shows lower errors in both event categories. These results suggest that VibFormer can capture complex temporal patterns even in continuous vehicle events. Furthermore, Fig. 4. illustrates a continuous vehicle event from the evaluation dataset, which shows a good similarity and accurate peak displacements.

TABLE I. Results on evaluation dataset. Boldface fonts indicate the best values for each metric

Evaluation method	**Single vehicle event**		**Continuous vehicle event**	
	Cross-correlation	***MAE***	***Cross-correlation***	***MAE***
FVM Integration	84%	15%	58%	21%
Proposed VibFormer	**91%**	**6%**	**87%**	**9%**

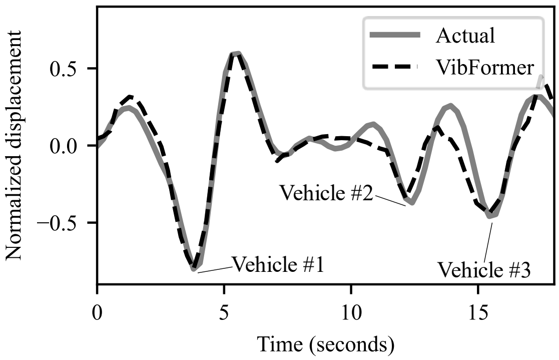

Fig. 4. Displacement signals of three continuous vehicles, ground-truth (solid line) and estimated from VibFormer method (dashed line).

V. Conclusion

A vibration translation method, VibFormer, for bridge live-load displacement estimation from acceleration has been proposed. Vibration feature extraction by splitting static and dynamic frequency components simplified the complex long-term structure in time-series sequences, improving the overall translation efficiency. It has been shown that the proposed method outperformed the existing method in an experimental dataset of a real bridge. We further plan to increase the range of the training dataset to include more vibration signals of multiple bridge structures.

References

[1] L. Yao, X. Li, J. Li, and C. Wang, “Application of data acquisition and intelligent analysis in bridge operation safety monitoring,” *2022 4th IEEE International Conference on Intelligent Control, Measurement and Signal Processing (ICMSP)*. Jul. 08, 2022.

[2] X. Li and F. Li, “Displacement monitoring requirements and laser displacement monitoring technology of bridges with short and medium spans,” *Applied Sciences*, vol. 12, no. 19. MDPI AG, p. 9663, Sep. 26, 2022.

[3] H. Sekiya, O. Maruyama, and C. Miki, “Visualization system for bridge deformations under live load based on multipoint simultaneous measurements of displacement and rotational response using MEMS sensors,” *Engineering Structures*, vol. 146. Elsevier BV, pp. 43–53, Sep. 2017.

[4] H. Sekiya, K. Kubota, and C. Miki, “Simplified portable bridge weigh-in-motion system using accelerometers,” *Journal of Bridge Engineering*, vol. 23, no. 1. American Society of Civil Engineers (ASCE), Jan. 2018.

[5] Z. Aliansyah, K. Shimasaki, T. Senoo, I. Ishii, and S. Umemoto, “Single-camera-based bridge structural displacement measurement with traffic counting,” *Sensors*, vol.21, no.13. MDPI AG, p.4517, 2021.

[6] H. H. Nassif, M. Gindy, and J. Davis, “Comparison of laser Doppler vibrometer with contact sensors for monitoring bridge deflection and vibration,” *NDT & E International*, vol. 38, no. 3. Elsevier BV, pp. 213–218, Apr. 2005.

[7] M. Gindy, R. Vaccaro, H. Nassif, and J. Velde, “A state-space approach for deriving bridge displacement from acceleration,” *Computer-Aided Civil and Infrastructure Engineering*, vol. 23, no. 4. Wiley, pp. 281–290, May 2008.

[8] K. Helmi, T. Taylor, A. Zarafshan, and F. Ansari, “Reference free method for real time monitoring of bridge deflections,” *Engineering Structures*, vol. 103. Elsevier BV, pp. 116–124, Nov. 2015.

[9] S. Yoneyama and H. Ueda, “Bridge deflection measurement using digital image correlation with camera movement correction,” *Materials Transactions*, vol. 53, no. 2. Japan Institute of Metals, pp. 285–290, 2012.

[10] J. J. Lee and M. Shinozuka, “A vision-based system for remote sensing of bridge displacement,” *NDT & E International*, vol. 39, no. 5. Elsevier BV, pp. 425–431, Jul. 2006.

[11] S. Yoneyama, A. Kitagawa, S. Iwata, K. Tani, and H. Kikuta, “Bridge deflection measurement using digital image correlation,” *Experimental Techniques*, vol. 31, pp. 34–40, Jan. 2007.

[12] M. Gindy, H. H. Nassif, and J. Velde, “Bridge displacement estimates from measured acceleration records,” *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2028, no. 1. pp. 136–145, Jan. 2007.

[13] J. W. Park, S. H. Sim, H. J. Jung, and B. Jr., “Development of a wireless displacement measurement system using acceleration responses,” *Sensors*, vol. 13, no. 7. MDPI AG, pp. 8377–8392, Jul. 01, 2013.

[14] H. Sekiya, K. Kimura, and C. Miki, “Technique for determining bridge displacement response using MEMS accelerometers,” *Sensors*, vol. 16, no. 2. MDPI AG, p. 257, Feb. 19, 2016.

[15] S. Cho, S. H. Sim, J. W. Park, and J. Lee, “Extension of indirect displacement estimation method using acceleration and strain to various types of beam structures,” *Smart Structures and Systems*, vol. 14, no. 4, pp. 699–718, Oct. 2014.

[16] F. Huseynov, D. Hester, E. J. O’Brien, C. McGeown, C. W. Kim, K. Chang, and V. Pakrashi, “Monitoring the condition of narrow bridges using data from rotation-based and strain-based bridge weigh-in-motion systems,” *Journal of Bridge Engineering*, vol. 27, no. 7. *American Society of Civil Engineers (ASCE)*, Jul. 2022.

[17] H. S. Moon, S. Ok, P. Chun, and Y. M. Lim, “Artificial neural network for vertical displacement prediction of a bridge from strains (part 1): girder bridge under moving vehicles,” Applied Sciences, vol. 9, no. 14, p. 2881, Jul. 2019, doi: 10.3390/app9142881.

[18] S. Morichika, H. Sekiya, Y. Zhu, S. Hirano and O. Maruyama, “Estimation of displacement response in steel plate girder bridge using a single MEMS accelerometer,” in *IEEE Sensors Journal*, vol. 21, no. 6, pp. 8204-8208, 2021.

[19] T. Kawakatsu, K. Aihara, A. Takasu, J. Adachi, H. Wang and T. Nagayama, “Fully-neural approach to vehicle weighing and strain prediction on bridges using wireless accelerometers,” *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 8027-8031, 2021.

[20] Y. Zhu, H. Sekiya, T. Okatani, I. Yoshida and S. Hirano, “Acceleration-based deep learning method for vehicle monitoring,” in *IEEE Sensors Journal*, vol. 21, no. 15, pp. 17154-17161, 2021.

[21] H. Zhu, Y. Zhou, and Y. Hu, “Displacement reconstruction from measured accelerations and accuracy control of integration based on a low-frequency attenuation algorithm,” *Soil Dynamics and Earthquake Engineering*, vol. 133. Elsevier BV, p. 106122, Jun. 2020.

[22] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” Advances in Neural Information Processing Systems (NeurIPS), 30, 2017.

[23] J. Yu, Z. Fang, X. Meng, Y. Xie, and Q. Fan, “Measurement of quasi-static and dynamic displacements of footbridges using the composite instrument of a smartstation and an accelerometer: case studies,” *Remote Sensing*, vol. 12, no. 16. MDPI AG, p. 2635, Aug. 15, 2020.

[24] G. Zerveas, S. Jayaraman, D. Patel, A. Bhamidipaty, and C. Eickhoff, “A transformer-based framework for multivariate time series representation learning,” *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, ACM*, 2021.

[25] C. H. H. Yang, Y. Y. Tsai, and P. Y. Chen, “Voice2series: reprogramming acoustic models for time series classification,” *International Conference on Machine Learning (ICML)*, pp. 11808-11819, 2021.

[26] J. Xu, H. Wu, J. Wang and M. Long, “Anomaly transformer: time series anomaly detection with association discrepancy,” *arXiv preprint* arXiv:2110.02642, 2021.

[27] S. Tuli, G. Casale, and N. R. Jennings, “Tranad: deep transformer networks for anomaly detection in multivariate time series data,” *arXiv preprint* arXiv:2201.07284, 2022.

[28] S. Li, X. Jin, Y. Xuan, X. Zhou, W. Chen, Y.-X. Wang, and X. Yan., “Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting,” *Advances in neural information processing systems (NeurIPS)*, 32, 2019.

[29] H. Zhou, S. Zhang, J. Peng, S. Zhang, J. Li, H. Xiong, and W. Zhang, “Informer: beyond efficient transformer for long sequence time-series forecasting,” In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 35, No. 12, pp. 11106-11115, 2021.

[30] H. Wu, J. Xu, J. Wang, and M. Long, “Autoformer: decomposition transformers with auto-correlation for long-term series forecasting,” *Advances in Neural Information Processing Systems (NeurIPS)*, 34, pp. 22419-22430, 2021.

[31] T. Zhou, Z. Ma, Q. Wen, X. Wang, L. Sun, and R. Jin, “Fedformer: frequency enhanced decomposed transformer for long-term series forecasting,” *International Conference on Machine Learning (ICML)*, pp. 27268-27286, 2022.

[32] I. Sutskever, O. Vinyals and Q. V. Le, “Sequence to sequence learning with neural networks,” *Advances in neural information processing systems (NeurIPS)*, 27, 2014.

[33] S. Kim, H. Lee, J. Han, and J. H. Kim, “Sig2Sig: signal translation networks to take the remains of the past,” *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021.

[34] A. Tran, K. Drossos, and T. Virtanen, “WaveTransformer: an architecture for audio captioning based on learning temporal and time-frequency information,” *IEEE 29th European Signal Processing Conference (EUSIPCO)*, 2021.