Ocean Engineering 310 (2024) 118639

Contents lists available at ScienceDirect

Ocean Engineering

journal homepage: www.elsevier.com/locate/oceaneng

Research paper

# Variational autoencoders and transformers for multivariate time-series generative modeling and forecasting: Applications to vortex-induced vibrations

Andreas P. Mentzelopoulos [a,*], Dixia Fan [b], Themistoklis P. Sapsis [a], Michael S. Triantafyllou [a]

[a] Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA, USA
[b] School of Engineering, Westlake University, Hangzhou, China

ARTICLE INFO

*Keywords:*
Variational autoencoders
Generative modeling
Generative-AI
Transformers
Forecasting
Time-series generation
Time-series forecasting
LSTM
Deep Neural Networks
Deep Learning

ABSTRACT

This study employs a data-driven approach to studying physical system vibrations, focusing on two main aspects: using variational autoencoders (VAEs) to generate physical data (i.e. data "similar" to those obtained via real-world processes) and using transformers in order to continuously forecast flexible body nonstationary vibrations (2D time-series) in time–space using information from sparse sensors on the body (observers). A VAE is trained on vortex-induced vibrations (VIV) data collected from experiments conducted by the authors and is then tasked with generating synthetic VIV data similar to the experimental. The synthetic data are then used to train a transformer architecture whose objective is to continuously forecast the vibrations in time–space using sparse observations. The transformer (which has never seen real data) is tested against real experiments and its performance is compared to that of the same architecture trained on real data. In doing so, the ability of VAEs to generate data which preserve their training data's intrinsic properties (i.e. physicality) is evaluated. Finally a comparison between the forecasting performance of the transformer architecture, an LSTM, and a DNN is presented.

## 1. Introduction

### 1.1. Vortex-induced vibrations (VIV)

Vortex-induced vibrations (VIV) are vibrations that affect bluff bodies in the presence of currents. VIV are driven by the periodic formation and shedding of vortices in the bodies' wakes which create an alternating pressure variation causing persistent vibrations (Triantafyllou et al., 2016). The vibration amplitude is typically moderate, not exceeding about one to two body diameters (Bernitsas et al., 2019). For flexible bodies, VIV are not uniform along the body's length (termed the span in literature) but rather different points along the body vibrate with different amplitudes and phases (visually resembling a taut string), as shown if Fig. 1.

Today, VIV have become a problem of interest to both theoreticians, due to the complex underlying mechanisms involved, and engineers, due to the practical significance of mitigating the fatigue damage VIV can cause to offshore structures and equipment such as marine risers and offshore wind turbines.

For flexible bodies, the vortex formation frequency coincides with the frequency of vibration in a phenomenon known as lock-in (Navrose and Mittal, 2016); this occurs across a wide range of oscillating frequencies resembling a nonlinear resonance (Park et al., 2016). Given that flexible body VIV are not span-wise uniform as the flexible body undergoes a spatially traveling and/or standing wave response from the forcing exerted by the fluid (Wang et al., 2021; Triantafyllou et al., 2016; Fan, 2019), the observed motions are unsteady, nonstationary, and can transition to different responses even for seemingly unchanged experimental conditions (Williamson, 1996) making continuous forecasting notoriously challenging. In addition, the vibrations are not Markov and alterations of the flow field in the past affect the future response outcomes (i.e. the driving mechanism has memory).

Current state-of-the-art prediction technologies for VIV are semi-empirical physics based models like VIVA (Zheng et al., 2011), VIVANA (Larsen et al., 2017), and Shear7 (Vandiver, 1999), whose accuracy relies heavily on the semi-empirical coefficients used and are limited to forecasting the vibrations on average: predicting the root-mean-square (rms) of the vibrations averaged over many cycles. Continuous time–space reconstructions with no forecasting capabilities of flexible body VIV have only recently been attempted when Kharazmi et al. (2021) attempted to continuously reconstruct the vibrations using LSTM networks in modal space (LSTM-Modnet).

* Corresponding author.
*E-mail address:* ament@mit.edu (A.P. Mentzelopoulos).

https://doi.org/10.1016/j.oceaneng.2024.118639
Received 2 May 2024; Received in revised form 26 June 2024; Accepted 1 July 2024

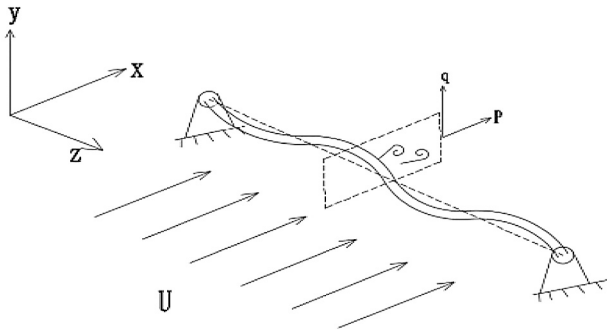


**Fig. 1.** Schematic of a flexible body undergoing vortex-induced vibrations (VIV). *Source:* Adapted from Feng et al. (2019).

Although leveraging transformers to predict time series is a very active field of research (Zhou et al., 2021; Zeng et al., 2023; Liu et al., 2022; Zhou et al., 2022), transformers have not yet been used to predict VIV of flexible bodies, which are physical non-stationary time-series, to the best of the authors' knowledge. In addition, only limited work has been performed in generating physical data using generative models (Zhong and Meidani, 2023; Takeishi and Kalousis, 2021; Shu et al., 2023) and there are no identifiable applications to VIV in the literature. Moreover, imposing physical characteristics on synthetic VIV data through a combined (physics informed) loss is currently not possible, as it would require force measurements on the body, an impossible task using data from real experiments. Applications of machine learning methods for VIV mostly include solving physical equations using physics-informed neural networks (PINNs) (Bai and Zhang, 2022; Raissi et al., 2019) and learning hydrodynamic coefficients or other relevant quantities (Ma et al., 2022) in order to predict the motions on average rather than instantaneously (Ma et al., 2021; Rudy et al., 2021; Mentzelopoulos et al., 2022, 2023).

In this work, a purely data-driven approach will be employed to assess whether synthetic VIV data generated using a variational autoencoder are physical. Physicality in this context will be measured by the ability of the synthetic data to inform models: a comparison will be made between a transformer architecture tasked with forecasting VIV trained on real data and the same transformer trained on synthetic data. In both cases the transformer will be tasked with forecasting real data collected from experiments. The rest of the paper will be organized as follows: first, a VAE architecture will be developed, trained on real data from experiments and used to generate synthetic VIV data. Second, a transformer architecture for continuous time–space VIV forecasting will be developed and trained on real data to assess its predictive capabilities. Finally, the same transformer architecture will be trained on the synthetic data (only) and tasked with forecasting the real experiments. In doing so, the ability of the VAE to generate data which carry (at least partially) meaningful information of the real data obtained via the physical process will be examined. Finally, an architecture comparison will be made between forecasting experiments using transformers, Long Short-Term Memory (LSTM) networks, and Deep Neural Networks (DNN).

### 1.2. Data in brief

#### 1.2.1. Physical experiments

All data used for this study were collected during experiments conducted by the authors at the MIT Towing Tank, a facility consisting of a 100ft × 8ft × 4ft water tank equipped with a towing carriage capable of reaching speeds exceeding 2 m/s as well as a flow visualization window. In this and the following sections the terms riser model, riser, flexible body, and flexible cylinder will be used interchangeably to refer to the flexible cylinder model used during experiments.

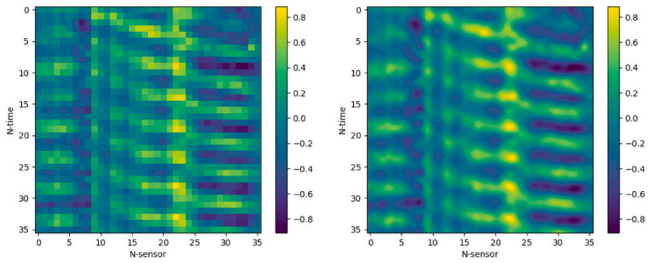


**Fig. 2.** Visualization of experimental data as images. By storing the data in 2D arrays of size N-time × N-sensor = 36 × 36, the x-axis corresponds to body location and the y-axis corresponds to time step (i.e. $\Delta t = 1/\text{fps} = 1/120$ s between rows). The displacement normalized by the body's diameter is highlighted on the plane. On the left we visualize the data stored in a 2D array. On the right, we plot the interpolated values which may be more intuitively visualized as flexible body vibrations. Sections parallel to the x-axis are "snapshots" of the flexible body vibrating in the direction perpendicular to the paper (i.e. "in and out of the page").

A riser model with length $L = 0.89$ m and diameter $D = 0.005$ m with negligible stiffness was towed at a uniform flow of speed $U = 0.7$ m/s. The resulting motions were recorded using two underwater cameras facing perpendicular directions: one for in-line motions (parallel to the incoming flow) and one for cross-flow motions (perpendicular to the incoming flow). The displacement (i.e. position with units of physical distance) at 36 uniformly spaced locations on the body was tracked using a machine-vision framework. The endpoints were fixed and thus their displacement was zero throughout the experiments.

For this study, the cross-flow displacement data were used: specifically, the cross-flow displacement data normalized by the body's diameter. For more information on the experimental setup one may refer to Appendix A. The machine-vision framework for motion tracking of the flexible body vibrations from raw frames is discussed in Mentzelopoulos et al. (2024).

#### 1.2.2. Vibration data as images

Given the data sampled at 36 uniformly spaced locations along the body's span at 120 fps, the vibrations were stored as 2D arrays of shape N-time × N-sensor.

Fig. 2 illustrates how the 2D data arrays can easily be visualized and treated like single channel images. If necessary, scaling pixel values invertibly to an interval of choice, like [0,1], is achievable in just a few operations leveraging the maximum and minimum values of the data. In the images shown above, each row corresponds to a different time of the recorded vibration at the sampled locations. The time difference between consecutive rows is $\Delta t = 1/\text{fps} = 1/120$ s with time increasing *downwards*. The 36 "sensor locations" correspond to the uniformly spaced tracked positions on the body. Plotting the interpolated values of the arrays yields a more intuitive visualization of the vibrations. For convenience, all the data collected from experiments were stored in a single 4D array of size $\text{N}_{batch} \times 1 \times \text{N-time} \times \text{N-sensor} = 260 \times 1 \times 36 \times 36$, yielding hundreds of square arrays of size 36 × 36 which could be easily visualized and collected in batches for training models.

## 2. Methodology & experiments

### 2.1. Generative-AI for physical vibration data using variational autoencoders

In this section we focus on generating physical vibration data using generative-AI techniques. Specifically, a VAE is trained on the experimental data to generate synthetic data of the vibrations. We are primarily interested in understanding whether the generated data preserve physicality.

As shown in Fig. 3, the variational autoencoder consists of an encoder network which learns the mean $\mu$ and variance $\sigma$ of the posterior

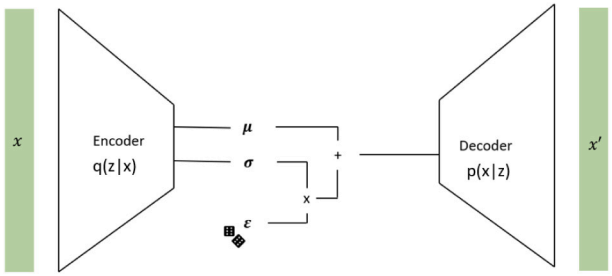


**Fig. 3.** Schematic of the variational autoencoder architecture. The network structure consists of an encoder network mapping data from the input space to the mean $\mu$ and variance $\sigma$ of the posterior distribution in the latent space, assuming a Gaussian prior, and a decoder network mapping data probabilistically from the latent space back to the input space. The variable $\epsilon$ is standard Gaussian random.

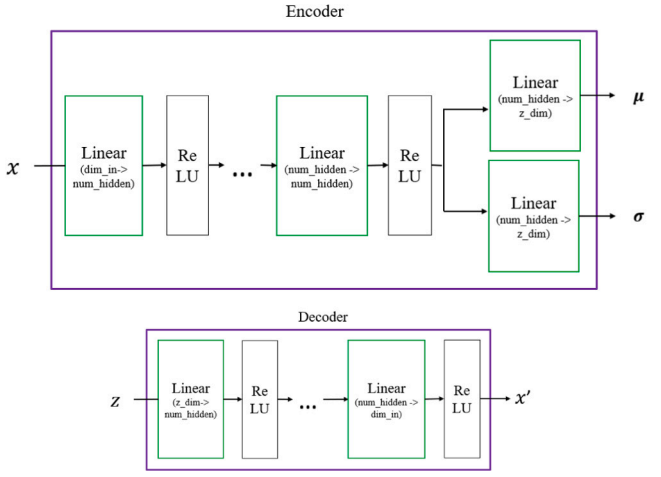


**Fig. 4.** Architecture of the encoder and decoder networks of the VAE. The encoder network uses a total of 4 MLP layers to map from the inputs to $\mu$ and $\sigma$ (2 sequential and two stacked) while the decoder uses 2 sequential MLP layers.

**Table 1**
Parameter values for the VAE.

| Dimension | Value |
|---|---|
| dim_in | 36 × 36 |
| num_hidden | 32 |
| z_dim | 5 |

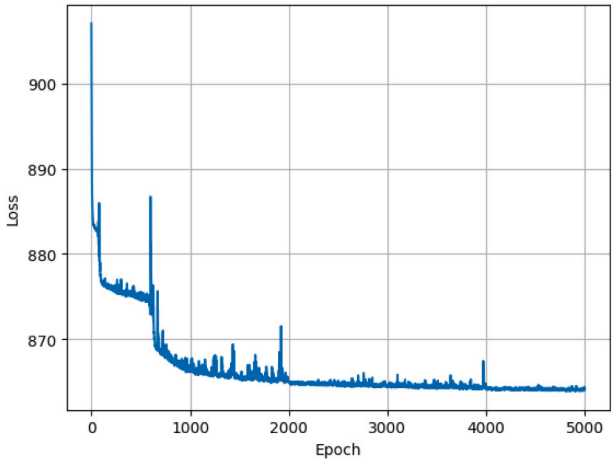


**Fig. 5.** Variational autoencoder loss.

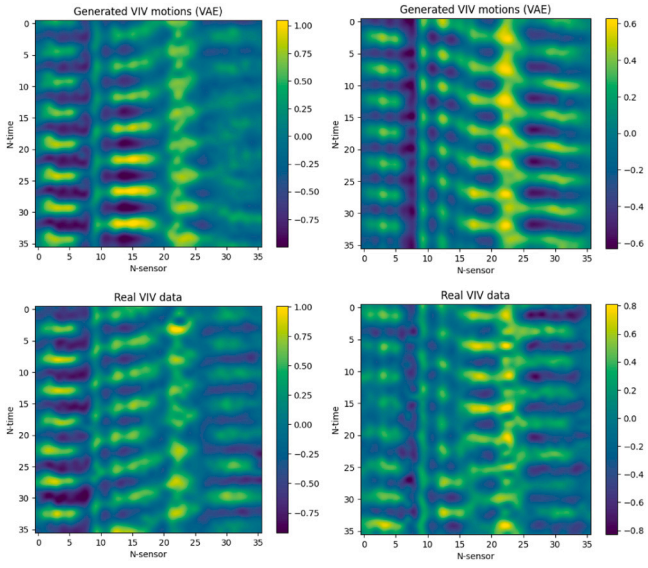


**Fig. 6.** Synthetic vortex-induced vibration data generated using the variational autoencoder. Two random samples of generated data are shown in the top row along with two random samples of real data from experiments in the bottom row.

(latent) distribution of the data, $q(z \mid x)$, assuming a Gaussian prior, and the decoder which learns the posterior of the input data given their latent representations $p(x \mid z)$.

Although asymmetric, both the encoder and decoder architectures consist of MLP layers. Both architectures are shown in Fig. 4. The encoder network uses a total of 4 MLP layers to map from the inputs to $\mu$ and $\sigma$ while the decoder uses two sequential MLP layers. The hidden units (num_hidden) of all linear layers was the same. The dimensions of the architecture parameters are summarized in Table 1.

Training was done by maximizing the evidence lower bound (ELBO) on the experimental data and the outputs of the variational autoencoder. This is equivalent to minimizing the following loss (negative of ELBO).

$$\text{Loss} = -\mathbb{E}_{q(z|x)}\Big[\log p(x \mid z) - D_{KL}(q(z \mid x) \,||\, q(z))\Big] \tag{1}$$

where $D_{KL}$ refers to the Kullback–Leibler divergence and $q(z)$ is the prior latent distribution, assumed standard multivariate Gaussian. The VAE was trained using Adam optimizer with a learning rate $lr = 0.01$ for a total of 5000 epochs. A step scheduler was set to decay the step by $\gamma = 1/2$ every 2000 iterations. The training loss as a function of epoch is shown in Fig. 5.

Having trained the VAE, samples from the standard normal distribution were drawn and decoded in order to generate synthetic VIV data. Two random samples are included in Fig. 6, along with two random samples of real data observed during experiments.

Albeit the generated data are qualitatively similar to the real data obtained from experiments as shown in Fig. 6, their promise begs the question of whether they preserve the intrinsic properties of the real experimental data and whether they carry meaningful information of the underlying physical process (i.e. whether physicality is preserved). In order to address this question, we will examine whether a model trained on synthetic data can be used to predict real experiments. Should the data preserve physicality, the generative models could potentially be used both to study and to simulate VIV.

On the choice of the generative model, we note the VAE was chosen due to its ability to learn and generate smooth and continuous data which is important for physical vibrations as well as its inherent regularization and robustness. The latent space of the VAE is a smooth Gaussian allowing for effective interpolations among latent

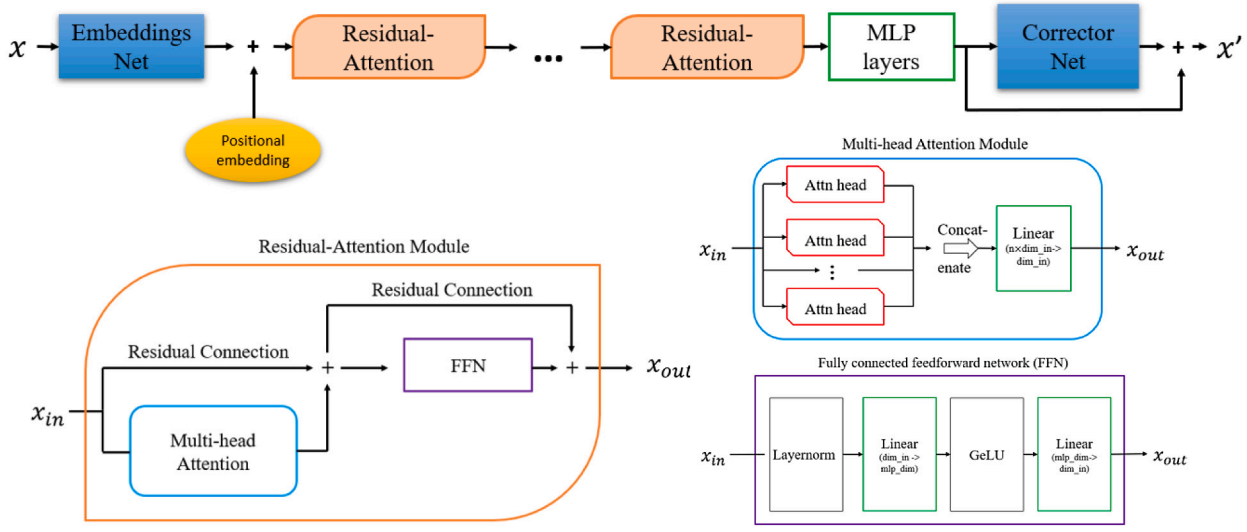


**Fig. 7.** VIVformer: a transformer architecture used for predicting VIV. The VIVformer consists of sequential Residual-Attention modules followed by a final linear layer. The VIVformer performs multi-head attention with residual connections while also leverages a fully connected feed-forward networks as its core data processing schemes. The architecture also leverages and embeddings network which transforms the input into embeddings and a corrector network operates on the outputs of the residual attention modules.

representations and the KL divergence term in the objective function (Eq. (1)) acts a s a regularizer ensuring that the latent representation is meaningful. Moreover, the probabilistic nature of the encoder and decoder makes the model fit for handling noisy physical data without propagating noise in the generated samples. On the other hand, VAEs can be tricky to train and may suffer from poor sample quality should the hyperparameters chosen not be well enough tuned.

### 2.2. VIVformer - A transformer architecture for VIV

In this section we develop the VIVformer, a transformer architecture to model and forecast the VIV of flexible bodies. The model's architecture is shown in Fig. 7: the VIVformer consists of 1. an embeddings network, 2. sequential Residual-Attention modules followed by linear layers, 3. a corrector network with a residual connection. The input to the VIVformer is a batch of vibration data with shape N-batch × $N_{time-in}$ × N-observer. The data are then passed through the embeddings net, $N_{attn-layers}$ residual attention modules, scaled to the desired $N_{time-out}$/N-sensor, and corrected, yielding an N-batch × $N_{time-out}$ × N-sensor output.

The residual-attention modules are the drivers of the data processing and are similar to the encoder modules of the original transformer proposed by Vaswani et al. (2017). These modules accept an input on which they perform two sequential tasks: 1. multi-head attention with a residual connection, and 2. pass the output of the multi-head attention module through a fully connected feedforward network (FFN) with a residual connection. The process can be visualized in the bottom left of Fig. 7.

The multi-head attention layer is comprised of $N_{heads}$ number of attention heads which calculate the self-attention of the input. The superposition of the input and output from the Multi-head attention module is then passed through the FFN. The FFN performs layer normalization, passes the output through a linear layer which scales the input to mlp-dim, then through a Gaussian Error Linear Unit (GeLU) activation and scales the output back to the original dimension by passing through a second linear layer. Both these processes are illustrated on the bottom right of Fig. 7.

The embeddings network is a deep network consisting of linear layers with GELU activations and transforms the input data to a high dimensional space on which the residual-attention modules operate. The corrector network accepts the output of the residual-attention modules (output dimension) and then applies a correction. This network consists of linear layers with GELU activations and employs a residual connection, so the network can learn to operate mildly or heavily on the inputs. The input and output dimension to the corrector network are the same.

**Table 2**
Architecture parameters for the VIVformer.

| Dimension | Value |
|---|---|
| N-observer (spatial points in) | 3 |
| N-sensor (spatial points out) | 36 − 3 = 33 |
| $N_{time-in}$ (time steps in) | 6 |
| $N_{time-out}$ (time steps out) | 1 |
| emb-dim (dimension of input embeddings) | 128 |
| Embeddings net layers | 5 |
| Embeddings net hidden dimension | 512 |
| $N_{attn-residual}$ (Residual-attention layers) | 8 |
| $N_{attn-heads}$ (No. attention heads) | 8 |
| attn_dim (hidden attention dimension) | 128 |
| mlp_dim (FFN hidden dimension) | 256 |
| Corrector network layers | 5 |
| Corrector network hidden dimension | 512 |

Since we are interested in making predictions of physical vibration data, the VIVformer's parameters were trained to minimize the Mean Square Error (MSE) between forecasted and observed vibrations.

#### 2.2.1. VIVformer trained on data from real experiments

In this section, the experimental data obtained during experiments were used to train the VIVformer. Specifically, the VIVformer was tasked to predict $N_{time-out}$ = 1 future time step of data at N-sensor = 36 − 3 = 33 locations using $N_{time-in}$ = 6 time steps of input data at N-observer = 3 locations. The architecture parameters are shown in Table 2.

The model was trained on the MSE loss between predictions and experimental observations (targets) and the parameters were updated using the AdamW algorithm. The initial learning rate was set to $lr$ = 0.0001 and a cosine annealing step scheduler was set to adjust the learning rate during training.

The training data were split into 80% for training and 20% for validation. The training data were shuffled randomly and split in mini-batches of size 128 while the validation data were not in order preserve the continuity of the vibrations when validating (important mainly for visualization purposes). The VIVformer was trained for a total of 60 epochs.

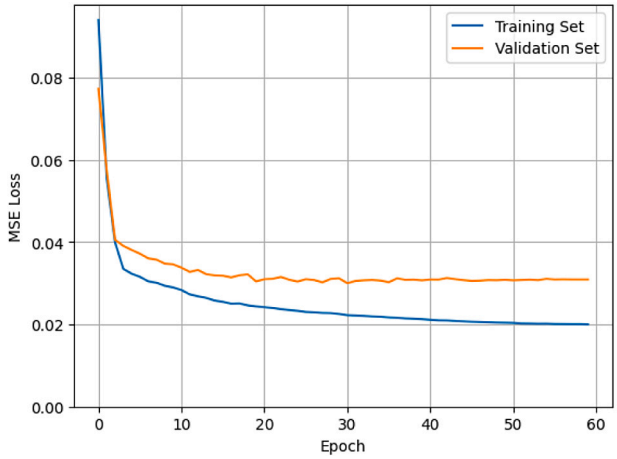


**Fig. 8.** VIVformer training and validation loss trained on experimental VIV data.

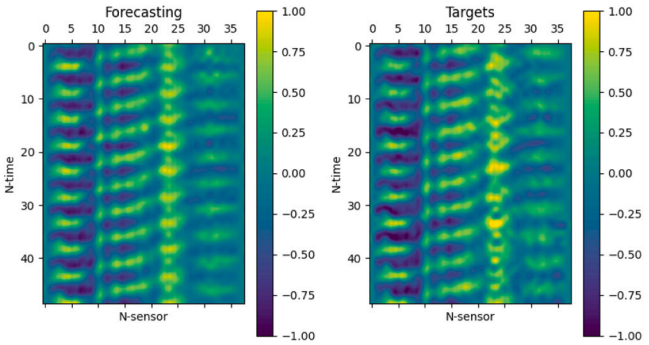


**Fig. 9.** Continuous time–space forecasting of VIV on unseen data from real experiments (left) and target data from observations (right).

Fig. 8 illustrates the training results of the VIVformer trained on real data. As is evident in the figure, the model is able to gradually decrease the MSE loss between targets and predictions. The loss on both the training set and the validation set seems to be decreasing and converging.

The continuous time–space forecasting of the VIVformer architecture as well as the target data from a random sample of vibration data from the validation set are shown in Fig. 9. As is evident in the figure, the model's predictions on unseen experimental data are in reasonable agreement with the target predictions. The absolute difference between forecasting and observation is included in Appendix B: the mean absolute difference between forecasting and observation is less than 0.09 body diameters. Another way to visualize the accuracy of the model would be to estimate the root mean square (RMS) of the predictions, which also gives a sense of the predictive capabilities on average and allows for direct comparisons with semi-empirical models. Fig. 10 illustrates the RMS vibration of the experiments as well as the prediction. Evidently, the model can predict the vibrations reasonably accurately on average.

Overall, with respect to training on real data, the transformer is reasonably accurate in terms of forecasting future motions at 36 locations on the body using information provided by 3 observers. The model trains well on the MSE loss and seems to be converging.

### 2.2.2. *VIVformer trained on synthetic data generated using the VAE*

So far we have established that the VIVformer architecture can forecast the physical VIV of flexible bodies to reasonable accuracy given sparse observers on the body. This section will mainly focus on addressing the question of whether synthetic VIV data generated

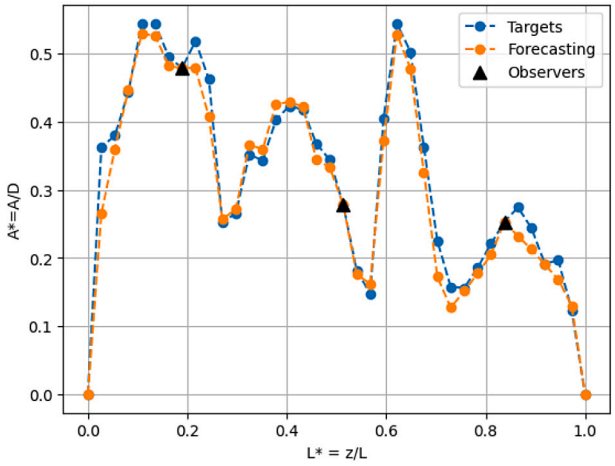


**Fig. 10.** Root mean square (RMS) of the predicted VIV as well as the experimentally observed. RMS displacement is shown on the y-axis while span (body position) is shown on the x-axis. Reasonably accurate agreement is evident between model estimation and experimental observations.

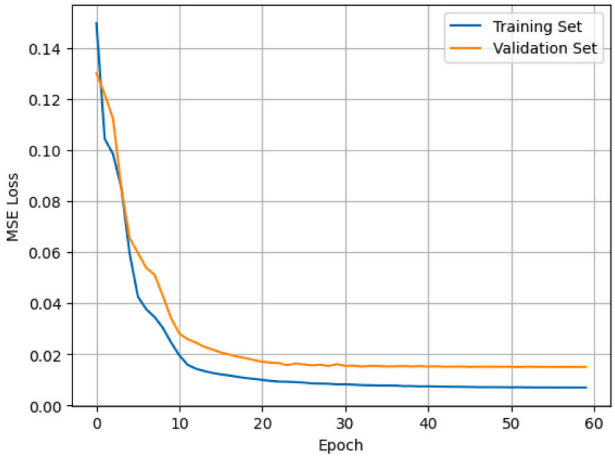


**Fig. 11.** VIVformer training and validation loss trained on synthetic VIV data.

using the VAE are physical (or at least partially physical): that is, whether the physical properties of the vibrations are preserved during the generative process. In order to address this question, we will train the VIVformer on synthetic data only and then use the trained model to predict real data. Achieving a similar performance in terms of predictive capabilities using the same VIVformer architecture trained on real data and synthetic data (separately) would imply that the synthetic data carry physical information of the underlying process.

A total of 160 arrays of shape N-time × N-sensor = 36 × 36 were generated using the VAE (this can be thought of as generating 160 images similar to the ones shown in section "Generative-AI for physical vibration data using variational autoencoders"). The synthetic data were then used to train the VIVformer architecture. Training parameters were the same as those used for training on the real data (see Table 2); the only difference was the training data which were in this case synthetic only. The same split of 80% for training and 20% for validation was used on the synthetic data. The training results are shown in Fig. 11. The MSE loss on both the training and validation sets seems to be decreasing and converging. Given the training results, we can be confident that the VIVformer has learned to predict the synthetic data reasonably accurately. Notably, the training loss is a bit smaller on the synthetic data using the same architecture and training epochs which hints that the synthetic data are a bit easier to approximate. This

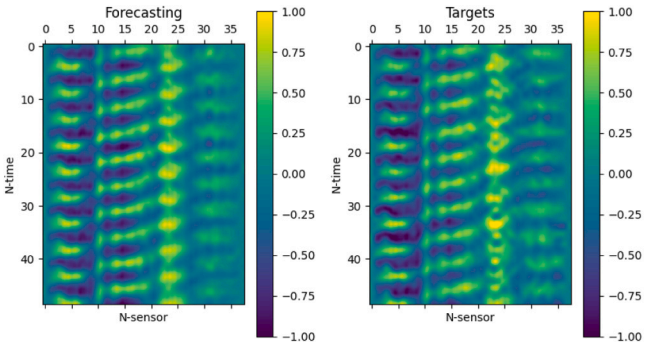


**Fig. 12.** Forecasting of the VIV on the same subset of the validation set (left) and target data from real experiments (right). The data are the same as those used in Section 2.2.1.

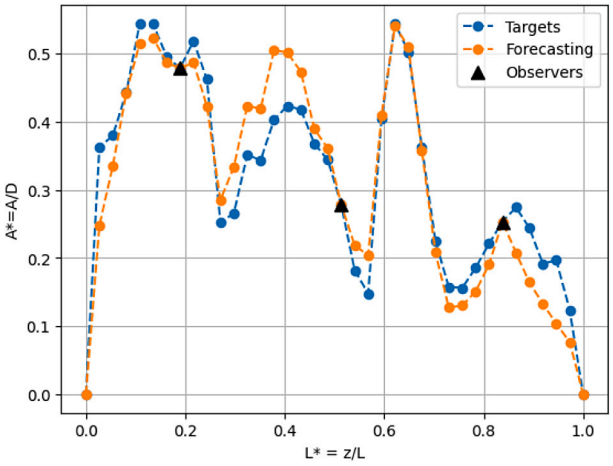


**Fig. 13.** Root mean square (RMS) of the predicted VIV plotted on top of the RMS of the experimentally observed VIV. RMS displacement is shown on the y-axis while span (body position) is shown on the x-axis. Reasonably accurate agreement is evident between forecasting (after training on synthetic data only) and experimental observation.

can be attributed to random noise present in the experimental data not carried through to the synthetic data.

The more important question is however, whether the VIVformer trained on the synthetic data can predict the real experiments. Fig. 12 illustrates the time–space forecasting of the real experiments using the VIVformer trained on synthetic data only. We underscore that the VIVformer has NOT seen a single real datum during training. The performance of the architecture is similar to that of training in the real data and the mean discrepancy between forecasting and targets is less than 0.11 diameters, which is only 0.02 diameters greater than the result obtained after training on the real data.

Albeit the VIVformer has not seen any real data during training, it can certainly make sensible predictions on the real data. The RMS of the forecasted vibrations and observed motions from experiments are shown in Fig. 13. As is evident in the figure, the VIVformer can make reasonably accurate predictions of the RMS of the vibrations. Both the trends and amplitudes are reasonably estimated although the performance is slightly worse compared to that of the architecture trained on real data, specifically for $L^* \in [0.2, 0.5]$.

Since the VIVformer has never trained on real data but can reasonably accurately predict them, we conclude that at least part of the physicality of the real data is preserved during the generative process of the VAE. In a sense, the VAE can be though of not just as a generator which makes realistic-looking data but as a framework which learns the real data's intrinsic properties: as such, it generates data which at least partially preserve physicality.

**Table 3**
Benchmarking architecture parameters for the VIVformer.

| Dimension | Value |
|---|---|
| N-observer (spatial points in) | 2 to 12 |
| N-sensor (spatial points out) | 36-(N-observer) |
| $N_{time-in}$ (time steps in) | 10 |
| $N_{time-out}$ (time steps out) | 1 |
| emb-dim (dim input embeddings) | 128 |
| Embeddings net layers | 1 |
| Embeddings net hidden dimension | N/A (single layer) |
| $N_{attn-residual}$ (Residual-attention) | 4 |
| $N_{attn-heads}$ (No. attention heads) | 2 |
| attn_dim (hidden attention dimension) | 128 |
| mlp_dim (FFN hidden dimension) | 64 |
| Corrector network layers | 2 |
| Corrector network hidden dimension | 256 |

### 2.3. Why VIVformer: comparison with LSTM and DNN

In this section we conduct experiments to investigate how the VIVformer performs compared to other benchmark deep-learning architectures: namely, LSTM networks and Deep neural nets. Specifically, for the given experimental data (real), each architecture is tasked with forecasting the observed motions with a variable number of observers; both the testing and validation losses are recorded. In order to keep the comparison fair, training parameters of different architectures are selected to be approximately equal (to a few hundred thousand) with the VIVformer having the *least* amount of trainable parameters (please see Figure 20).

The DNN architecture used is of 9 MLP layers with 256 neurons in each layer and ReLU activations. The LSTM architecture used is as follows: 1 linear layer to transform the input data into embeddings of dimension 256, 1 LSTM layer with hidden state of dimension 256, 2 linear layers to transform the output of the LSTM to the desired (N-sensor × N-time) dimension, and a corrector network with a residual connection (hidden dimension 256, 2 layers). The architecture may be visualized in Figure 19 and is virtually the same as that used for the VIVformer with the Residual-Attention modules replaced by an LSTM module. For this section, the VIVformer parameters are reduced to those shown in Table 3 in order to make the models even in terms of number of parameters.

The number of observers tested are 2, 3, 4, 6, 9, and 12, uniformly spaced and the number of input time-steps is set to 10 for all architectures. For each number of observers, each of the architectures is trained on 80% of the available data and validated on 20% of the data. The split was the same for all architectures and number of observers. The results of each individual training are included in Appendix B.

Fig. 14 illustrates the training (left) and validation (right) results of the different architectures as a function of number of observers. As is evident in the figure (right), the validation loss is lowest for the VIVformer regardless of number of observers followed by the LSTM and the DNN. As the number of observers increases, the gap between the VIVformer and the LSTM seems to converge while the gap between the VIVformer and the DNN seems to diverge. In addition, the exact opposite trend is evident for the training loss, which means that the VIVformer architecture has the smallest discrepancy between training and validation losses. We underscore, that in every case, the number of trainable parameters was *smallest* for the VIVformer, followed by the DNN, and the LSTM. Given the comparison results we conclude that the VIVformer architecture compares favorably against benchmark architectures of DNN and LSTM on the VIV data at hand.

## 3. Conclusions

In this work, a data driven approach is employed to study physical system vibrations. Two main topics are explored: 1. Variational autoencoders for generating synthetic data similar to those obtained via

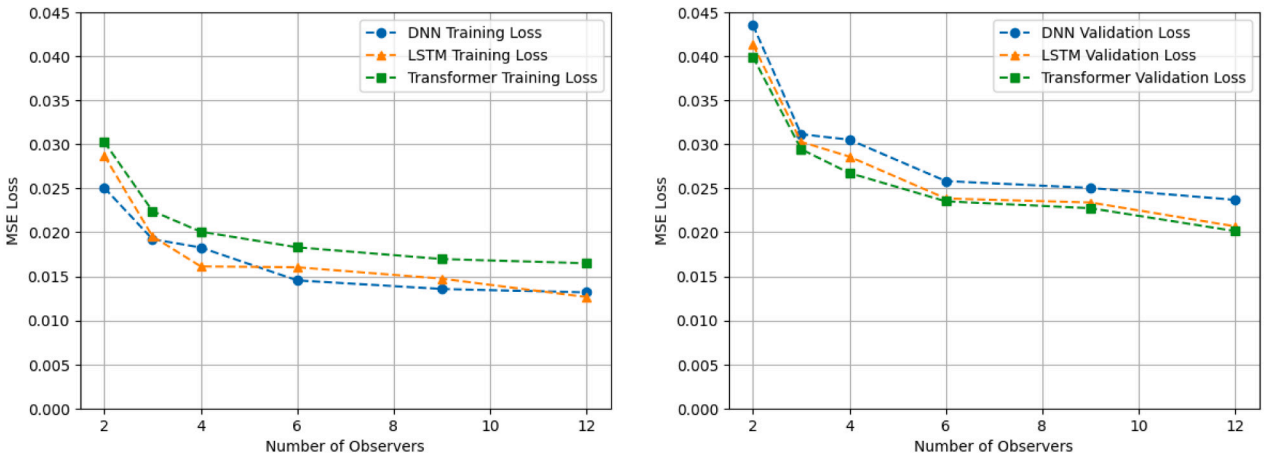


**Fig. 14.** Left: Training loss as a function of number of observers for various deep-learning architectures (DNN, LSTM, VIVformer). Right: Validation loss as a function of observers for various the different architectures (DNN, LSTM, VIVformer).

physical processes and 2. employing a transformer to forecast physical nonstationary vibrations.

A variational autoencoder is trained on physical vortex-induced vibration data in order to generate synthetic data of the vibrations. The VAE is certainly able to generate data which resemble the physical data qualitatively. Moreover, the generative process is confirmed to preserve physicality at least partially: a transformer trained on synthetic data only is capable of predicting the observed motions from physical experiments to reasonable accuracy and with a similar performance to that of the same architecture trained on real data. In that sense, the VAE can be viewed as an object which learns the intrinsic characteristics of the data and can thus be used as a data generator in order to simulate the underlying physical process or to provide data for dataset augmentations relatively cheap (VIV experiments cost from thousands to millions of USD depending on scale). In addition, we anticipate that with sufficient development and conditioning in the future, the VAE could be used to simulate specific VIV conditions, in a fashion similar to CFD codes: allowing one to predict fluid behavior in various conditions, decreasing the need for extensive model testing.

A transformer architecture for forecasting unsteady and nonstationary vortex-induced vibrations, the VIVformer, is developed. The VIVformer architecture combines multi-head attention modules and fully connected network modules with residual connections in order to forecast the vibrations' time-series in both time and space. The architecture is compared against benchmark deep-learning architectures of DNN and LSTM networks and is shown to compare favorably on the data. In addition, the architecture is shown to forecast flexible body VIV in time–space reasonably accuracy both instantaneously and on average.

## CRediT authorship contribution statement

**Andreas P. Mentzelopoulos:** Writing – review & editing, Writing – original draft, Validation, Software, Methodology, Formal analysis, Conceptualization. **Dixia Fan:** Writing – review & editing, Supervision, Software, Methodology, Conceptualization. **Themistoklis P. Sapsis:** Writing – review & editing, Supervision, Methodology, Conceptualization. **Michael S. Triantafyllou:** Writing – review & editing, Supervision, Resources, Methodology, Conceptualization.

## 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.

## Data availability

All software and data will be made publicly available through the Open VIV Repository hosted by MIT.

## Acknowledgments

The authors would like to acknowledge support from the DigiMaR Consortium, MathWorks, and the Onassis Foundation.

## Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.oceaneng.2024.118639.

## References

Bai, X.-D., Zhang, W., 2022. Machine learning for vortex induced vibration in turbulent flow. Comput. & Fluids 235, 105266.

Bernitsas, M.M., Ofuegbe, J., Chen, J.-U., Sun, H., 2019. Eigen-solution for flow induced oscillations (viv and galloping) revealed at the fluid–structure interface. In: ASME 2019 38th International Conference on Ocean, Offshore and Arctic Engineering. American Society of Mechanical Engineers Digital Collection.

Fan, D., 2019. Mapping the Hydrodynamic Properties of Flexible and Rigid Bodies Undergoing Vortex-Induced Vibrations (Ph.D. thesis). Massachusetts Institute of Technology.

Feng, Y., Li, S., Chen, D., Xiao, Q., 2019. Predictions for combined in-line and cross-flow viv responses with a novel model for estimation of tension. Ocean Eng. 191, 106531.

Kharazmi, E., Wang, Z., Fan, D., Rudy, S., Sapsis, T., Triantafyllou, M.S., Karniadakis, G.E., 2021. From data to assessment models, demonstrated through a digital twin of marine risers. In: Offshore Technology Conference. OnePetro.

Larsen, C., Lie, H., Passano, E., Yttervik, R., Wu, J., Baarholm, G., 2017. Vivana—Theory Manual, Version 4.10. 1. Sintef Ocean. Trondheim, Norway.

Liu, Y., Wu, H., Wang, J., Long, M., 2022. Non-stationary transformers: Exploring the stationarity in time series forecasting. Adv. Neural Inf. Process. Syst. 35, 9881–9893.

Ma, L., Resvanis, T.L., Vandiver, J.K., 2021. Enhancing Machine Learning Models with Prior Physical Knowledge to Aid in Viv Response Prediction. Vol. 85185, V008T08A021.

Ma, L., Resvanis, T.L., Vandiver, J.K., 2022. Understanding the higher harmonics of vortex-induced vibration response using a trend-constrained, machine learning approach. Mar. Struct. (ISSN: 0951-8339) 83, 103195. http://dx.doi.org/10.1016/j.marstruc.2022.103195, URL https://www.sciencedirect.com/science/article/pii/S0951833922000363.

Mentzelopoulos, A.P., del Águila Ferrandis, J., Rudy, S., Sapsis, T., Triantafyllou, M.S., Fan, D., 2022. Data-driven prediction and study of vortex induced vibrations by leveraging hydrodynamic coefficient databases learned from sparse sensors. Ocean Eng. (ISSN: 0029-8018) 266, 112833. http://dx.doi.org/10.1016/j.oceaneng.2022.112833, URL https://www.sciencedirect.com/science/article/pii/S0029801822021163.

Mentzelopoulos, A.P., Fan, D., Resvanis, T., Sapsis, T., Triantafyllou, M.S., 2023. Physics-based unsupervised learning of vortex-induced vibrations from riser field experimental stain data. In: The 33rd International Ocean and Polar Engineering Conference. OnePetro.

Mentzelopoulos, A.P., Prele, E., Fan, D., del Aguila Ferrandis, J., Sapsis, T., Triantafyllou, M.S., 2024. Reconstructing flexible body vortex-induced vibrations using machine-vision and predicting the motions using semi-empirical models informed with transfer learned hydrodynamic coefficients. J. Fluids Struct. (ISSN: 0889-9746) 129, 104154. http://dx.doi.org/10.1016/j.jfluidstructs.2024.104154, URL https://www.sciencedirect.com/science/article/pii/S0889974624000896.

Navrose, Mittal, S., 2016. Lock-in in vortex-induced vibration. J. Fluid Mech. 794, 565–594. http://dx.doi.org/10.1017/jfm.2016.157.

Park, H., Kumar, R.A., Bernitsas, M.M., 2016. Suppression of vortex-induced vibrations of rigid circular cylinder on springs by localized surface roughness at 3 ×104 ≤ re ≤ 1.2 ×105. Ocean Eng. 111, 218–233.

Raissi, M., Wang, Z., Triantafyllou, M.S., Karniadakis, G.E., 2019. Deep learning of vortex-induced vibrations. J. Fluid Mech. 861, 119–137.

Rudy, S., Fan, D., Ferrandis, J. d. A., Sapsis, T., Triantafyllou, M.S., 2021. Learning optimal parametric hydrodynamic database for vortex-induced crossflow vibration prediction. arXiv preprint arXiv:2104.05887.

Shu, D., Li, Z., Farimani, A.B., 2023. A physics-informed diffusion model for high-fidelity flow field reconstruction. J. Comput. Phys. 478, 111972.

Takeishi, N., Kalousis, A., 2021. Physics-integrated variational autoencoders for robust and interpretable generative modeling. Adv. Neural Inf. Process. Syst. 34, 14809–14821.

Triantafyllou, M.S., Bourguet, R., Dahl, J., Modarres-Sadeghi, Y., 2016. Vortex-induced vibrations. In: Springer Handbook of Ocean Engineering. Springer, pp. 819–850.

Vandiver, J., 1999. Shear7 Program User Manual. Massachusetts Institute of Technology, Cambridge, MA.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I., 2017. Attention is all you need. Adv. Neural Inf. Process. Syst. 30.

Wang, Z., Fan, D., Triantafyllou, M.S., 2021. Illuminating the complex role of the added mass during vortex induced vibration. Phys. Fluids 33 (8), 085120.

Williamson, C.H.K., 1996. Vortex dynamics in the cylinder wake. Annu. Rev. Fluid Mech. 28 (1), 477–539.

Zeng, A., Chen, M., Zhang, L., Xu, Q., 2023. Are transformers effective for time series forecasting? In: Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 37, pp. 11121–11128.

Zheng, H., Price, R., Modarres-Sadeghi, Y., Triantafyllou, G.S., Triantafyllou, M.S., 2011. Vortex-induced vibration analysis (viva) based on hydrodynamic databases. In: International Conference on Offshore Mechanics and Arctic Engineering. Vol. 44397, pp. 657–663.

Zhong, W., Meidani, H., 2023. Pi-vae: Physics-informed variational auto-encoder for stochastic differential equations. Comput. Methods Appl. Mech. Engrg. 403, 115664.

Zhou, T., Ma, Z., Wen, Q., Wang, X., Sun, L., Jin, R., 2022. Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting. In: International Conference on Machine Learning. PMLR, pp. 27268–27286.

Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., Zhang, W., 2021. Informer: Beyond efficient transformer for long sequence time-series forecasting. In: Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 35, pp. 11106–11115.