

RESEARCH ARTICLE | JULY 11 2023

Deciphering the dynamics of distorted turbulent flows: Lagrangian particle tracking and chaos prediction through transformer-based deep learning models

Special Collection: [Turbulence in Plasmas and Fluids](#)

R. Hassanian   ; H. Myneni  ; Á. Helgadóttir  ; M. Riedel 



Physics of Fluids 35, 075118 (2023)

<https://doi.org/10.1063/5.0157897>



Physics of Fluids
Special Topic: Overview of Fundamental
and Applied Research in Fluid Dynamics in UK
[Submit Today](#)



Deciphering the dynamics of distorted turbulent flows: Lagrangian particle tracking and chaos prediction through transformer-based deep learning models

Cite as: Phys. Fluids **35**, 075118 (2023); doi: 10.1063/5.0157897

Submitted: 12 May 2023 · Accepted: 24 June 2023 ·

Published Online: 11 July 2023



View Online



Export Citation



CrossMark

R. Hassanian,^{1,a)}  H. Myneni,¹  Á. Helgadóttir,¹  and M. Riedel^{1,2} 

AFFILIATIONS

¹The Faculty of Industrial Engineering, Mechanical Engineering, and Computer Science, University of Iceland, 102 Reykjavík, Iceland

²Juelich Supercomputing Centre, Jülich 52428, Germany

Note: This paper is part of the special topic, Turbulence in Plasmas and Fluids.

^{a)}Author to whom correspondence should be addressed: seh@hi.is

ABSTRACT

Turbulent flow is a complex and vital phenomenon in fluid dynamics, as it is the most common type of flow in both natural and artificial systems. Traditional methods of studying turbulent flow, such as computational fluid dynamics and experiments, have limitations such as high computational costs, experiment costs, and restricted problem scales and sizes. Recently, artificial intelligence has provided a new avenue for examining turbulent flow, which can help improve our understanding of its flow features and physics in various applications. Strained turbulent flow, which occurs in the presence of gravity in situations such as combustion chambers and shear flow, is one such case. This study proposes a novel data-driven transformer model to predict the velocity field of turbulent flow, building on the success of this deep sequential learning technique in areas such as language translation and music. The present study applied this model to experimental work by Hassanian *et al.*, who studied distorted turbulent flow with a specific range of Taylor microscale Reynolds numbers $100 < Re_\lambda < 120$. The flow underwent a vertical mean strain rate of 8 s^{-1} in the presence of gravity. The Lagrangian particle tracking technique recorded every tracer particle's velocity field and displacement. Using this dataset, the transformer model was trained with different ratios of data and used to predict the velocity of the following period. The model's predictions significantly matched the experimental test data, with a mean absolute error of 0.002–0.003 and an R^2 score of 0.98. Furthermore, the model demonstrated its ability to maintain high predictive performance with less training data, showcasing its potential to predict future turbulent flow velocity with fewer computational resources. To assess the model, it has been compared to the long short-term memory and gated recurrent units model. High-performance computing machines, such as JUWELS-DevelBOOSTER at the Juelich Supercomputing Center, were used to train and run the model for inference.

© 2023 Author(s). All article content, except where otherwise noted, is licensed under a Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>). <https://doi.org/10.1063/5.0157897>

I. INTRODUCTION

In fluid dynamics and physics, turbulent flow is a complex problem.¹ Turbulent flow is a nonlinear and high-dimensional phenomenon,² and it is commonly seen in industrial and natural applications.³ In addition, the universe is composed of turbulent and unsteady components.³ Thus, there is a tremendous interest in studying turbulent flow. In this study, experimental data from Hassanian *et al.* work⁴ are applied from the Lagrangian particle tracking technique for tracer particle seeded a turbulent deformation flow with a specific mean strain rate and a particular range of Taylor microscale Reynolds numbers.

Turbulent flow with deformation can be observed in various scenarios, including leading-edge erosion in compressors and turbines,⁵ combustion in internal engines, and particle interaction in mixing chambers.⁴ It can also occur in the external flow over an airfoil and in the internal flow of pipes with variable cross sections.⁴

The prior applied solution in turbulent flow has been experimented,⁶ and it is still the most robust way.⁷ However, designing and conducting experiments is expensive in the case of most natural and industrial flow studies because of their dimensions and scales, leading to constraints and making them even more often impossible to

perform.¹ The most known numerical approach to solving turbulent flow problems is the computational fluid dynamics (CFD) method.^{8,9} In CFD, the numerical approaches can be broadly classified into three categories based on the accuracy vs computation time: Reynolds averaged Navier Stokes (RANS), large eddy simulation (LES), and direct numerical simulation (DNS).¹⁰ RANS is applied widely in the industry and provides an average solution, not an exact one.¹¹ LES solves the problem with better accuracy than RANS, but weaker than DNS, and consequently, DNS proposes the exact answer.^{2,11} LES and DNS suffer cost computation, and the computation cost proliferates depending on the problem size.⁸ In this term, to implement LES and DNS, it is essential to apply high-performance computing.⁹ Despite development in parallel computing, this issue limits these two solver applications. It must be noted in most CFD solutions, validating the results via experiment plays a crucial role.¹⁰ Accordingly, to overcome above-mentioned obstacles, discovering and using an alternative method to provide the possibility to study the turbulent flow widely is an essential matter. Deep learning (DL) has recently been used broadly, proving remarkable capability in fluid dynamics.¹² To analyze turbulent flow in the Lagrangian framework, both spatial and temporal perspectives are crucial to identify flow characteristics in the future period. Among the various deep learning techniques, sequential architectures such as long short-term memory (LSTM) and combinations of convolutional neural networks (CNN) that cover the temporal perspective have proved to be effective models for resolving or predicting turbulent flow issues. It is well addressed that the statistics of turbulent flow are applicable,^{1,2} and they are sequence features based on the Lagrangian framework. Obviously, as literature has noted, LSTM variants are an excellent way for sequential datasets.¹³ CNN compositions for sequential data require a large dataset to train, and it has several layers, which leads to a lot of computing time.¹⁴

DL methods involve semi-supervised and unsupervised learning.¹⁴ In semi-supervised learning, the target has no label, and in unsupervised learning, the target pattern must be understood and extracted by the model.¹⁵ The main necessity in deep learning lies in the input data used to train the model. In the realm of fluid dynamics, obtaining an accurate dataset can be achieved through experiments and DNS. Zhou *et al.*¹⁶ applied a surrogate model based on CNN and higher-order dynamic mode decomposition to predict the unsteady fluid force time history for twin tandem cylinders. An unsupervised machine learning Gaussian mixture model for the detection of viscous-dominated and turbulent regions has been proposed by Otmani *et al.*¹⁷ Salehipour *et al.*¹⁸ applied a DL model to discover a generic parameterization of diapycnal mixing. Raissi *et al.*¹⁹ employed a DL model for the prediction of the lift and drag forces on the vortex structure. Kim *et al.*²⁰ presented an unsupervised learning model that can be trained with unpaired turbulence data for super-resolution reconstruction. Yousif *et al.*²¹ proposed a DL method composed of a convolutional auto-encoder and LSTM for generating turbulent inflow conditions. Lee *et al.*²² determined a data-driven deep learning model to predict unsteady flow over a circular cylinder. Hassanian *et al.*²³ applied LSTM variants to predict a deformed turbulent flow velocity field. Eivazi *et al.*²⁴ presented a physics-informed neural network application for solving RANS equations. Duru *et al.*²⁵ represents an application of DL to forecast the transonic flow around airfoils. Bukka *et al.*²⁶ defined a hybrid DL model to predict unsteady flows. Most fluid flow studies that applied DL use data extracted from CFD

computations.¹² Furthermore, most works included preprocessing steps to identify the dominant features, such as proper orthogonal decomposition or dynamic mode decomposition.¹² Recently, LSTM and GRU models are used to predict a turbulent flow with only temporal features.^{27,28} Based on above-mentioned terms, it is essential to correspond the DL model to turbulent flow with these features:

- Predicting turbulent flow with minimum training data that are possible to generate for the study case via DNS or experiment.
- The DL model training cost does not grow in relation to the size of the data.
- The DL model is able to extract the dominant feature from available data.
- The DL model performs reliably in a broad range of high Reynolds numbers.

Recently, transformer model from the attention mechanism represents the remarkable capability to simulate and forecast sequential datasets.^{29,30} This study aims to apply the transformer model²⁹ in a novel data-driven approach and assess it based on the above items. A transformer is a DL model based on encoder-decoder layers, and it processes the data through an attention mechanism.²⁹ It is used widely in language translation.³¹ The distinguishing characteristic of the Transformer is its architecture, which completely eliminates the use of recurrence and convolutions.²⁹ This intrinsic feature enables the transformer model to be highly parallelizable, resulting in significantly reduced training time and computational requirements.²⁹ Previous recurrent models encounter a sequence of hidden states as a function of previous states and input, and this sequential nature impedes parallelization with training examples. In addition, the attention mechanism allows modeling of dependencies no matter how far apart the inputs or outputs sequence are.^{29,31,32}

In this study, the proposed transformer model can be effectively utilized to analyze the strained turbulent flow cases mentioned earlier, offering valuable insights into the physical properties of turbulent flow. This enhanced understanding has wide-ranging applications in both industrial and natural settings. The dataset comprises two components: the velocity of each individual particle and the corresponding time recorded during the experiment. A subset of the dataset is employed to train the transformer model, while the remaining data are used to test the model's predictions and forecast the flow velocity for subsequent periods. Hence, this paper is organized as follows; the theory is described in Sec. II. The methodology and setup are presented in Sec. III. The results and conclusions are represented in Secs. IV and V, respectively.

II. THEORY

This study designs and proposes a transformer model in a data-driven approach using experimental turbulent flow data.⁴ In Secs. II A and B, first, the applied turbulent flow theory is described, and then, the transformer model architectures are explored.

A. Turbulent flow and Lagrangian particle tracking

The data used are from turbulent flow conducted in a laboratory, and details can be found in the original work.⁴ The flow has a Taylor microscale in the range of $100 < Re_\lambda < 120$. The flow underwent strain deformation with a specific mean rate of 8 s^{-1} in the y direction. Figure 1 displays a sketch of the generated flow. The experiment was

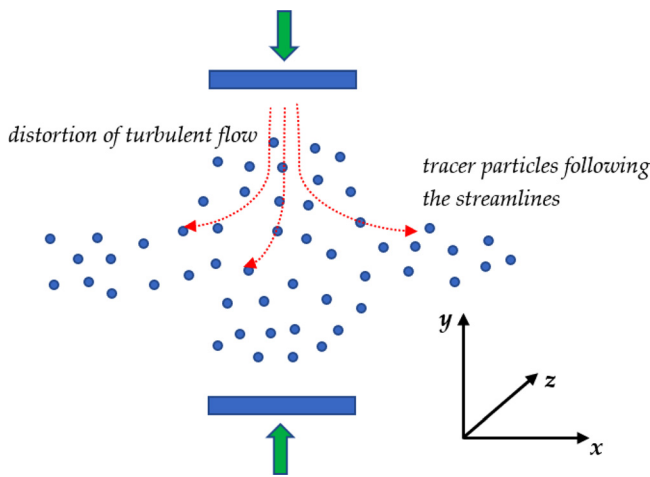


FIG. 1. Sketch of the generated turbulent flow: the particle size is for visualization in the figure, and it is not the actual scale.

conducted in the presence of gravity. The flow was seeded by tracer particles with median diameters 8–10 μm and specific gravity 1.1 g/cm³ (hollow glass) to extract the flow features. The Lagrangian particle tracking technique based on the work of Ouellette *et al.*³³ was employed to record the tracer particles' movement via the high-speed camera in 2D view. The original work assessed the tracer particle based on Stokes number to ensure that the particle track is following the flow streamlines. Stokes number *St* is an identification when it is *St* ≪ 1 for a particle, and it proves the particle pathline corresponding to the flow streamlines. The Stokes number for seeded particles has been reported in the range of 0.00632–0.01807.⁴ Thus, the measured data included velocity components in the *x* and *y* directions, location in the *x* and *y* coordinates, and corresponding time *t* for every moment measured. The data were recorded with 10000 frames per second (10 kHz) in the period of 0.2 s. From the Lagrangian perspective, the velocity of a fluid particle is defined by²

$$x_i = x_i(t, x_{i,0}), \tag{1}$$

$$U_i = U_i(t, x_1(t, x_{1,0}), x_2(t, x_{2,0}), x_3(t, x_{3,0})), \tag{2}$$

where the fluid particle position and velocity in 3D coordinates are determined by notations 1 and 2, respectively, *x* is the position, *U* is the velocity, *t* is the time, and *i* specifies the vector component. In this work, data were measured in 2D; therefore, the 3rd dimension has not been addressed.

This study applies a data-driven approach based on the above-described dataset. Since turbulent flow is a complex concept and is a high-dimensional phenomenon, the primary of the DL model application is to discover the dominant feature to be able to predict the following periods of the target segments. In this work, the available data consist of velocity and location in corresponding time. The turbulent flow is affected by the external mean strain rate and has deformed. In addition, the experiment was conducted in the presence of gravity, and its effect is unknown from previous studies.³⁴ In order to specify the input data, this study proposes a novel approach: the velocity components in sequence feeding the model as input and training the model. This design is based on the concept that velocity in turbulent

flow is considerable, and it is attained and carries the most properties of turbulent flow. Moreover, the model gains training for each velocity component individually in the *x* and *y* directions. This configuration allows the model to be used with 3D components and without limits in flow dimensions.

The superiority of this proposed approach is that it employs raw measured velocity datasets without preprocessing to specify the dominant feature or make the dimensional reduction. The turbulent flow velocity could be measured in many industries and natural applications via available devices.

B. Transformer and attention mechanism architecture

The transformer is a DL network composed of encoder–decoder architecture.³⁵ The input data feed the encoder layers and generate the output via the decoder.³⁶ The mechanism of this process has several steps. The number of encoder layers must be equal to the decoder layers. In order to specify the input sequence and distance, positional embedding is added to the input vectors. The positional embedded input vectors feed the first encoding layer, and the output of the previous encoder layer provides the successive layers. Every encoder layer is broken down into two sublayers. The first encoder inputs stream through a multi-head attention sublayer. In a multi-head sublayer, all inputs' dependencies are considered to create the weight matrices. The next step is the outputs from the multi-head sublayer stream to the feed-forward sublayer. Between these sublayers, there is an Add&Norm intermediate sublayer, which adds the inputs of the multi-head sublayer to its input and normalizes them. In the feed-forward sublayer, the data are independently applied to each position. Thus in the feed-forward sublayer, data can process in parallel and independently. The outputs from the feed-forward sublayer, in the same manner, should pass the Add&Norm intermediate sublayer. Consequently, the data have been proceeded into an encoder layer and then flowed into the next encoder layer. The number of the encoder layers have no specific and magic numbers,²⁹ and they must be resolved and determined in the architecture design for every problem. When the first transformer architecture was born,²⁹ it was designed with only six encoder–decoder layers with notable achievement. This feature of the transformer architecture mentions a remarkable capability to model sequential data problems with fewer layers. Reducing the number of layers in deep learning models could lead to less computing.

When the data cross through all encoder layers, then they will be encoded data flow to decoder layers to embed the outputs. The decoder layer consists of two multi-head and feed-forward sublayers that do the same as the encoder layer and are located after a masked sublayer. The first sublayer in the decoder, as a masked multi-head sublayer, makes a masking layer for the embedded outputs in this way that it only depends on the earlier data. It masks the next sequence of data and avoids their effects. When passed the Add&Norm intermediate sublayer, the output from the masked multi-head flows through the multi-head sublayer. Training the model in the encoding layers creates three weight matrices: Query, Key, and Value. Key and Value matrices from the latter encoder feed the multi-head sublayers of decoder layers directly. However, the multi-head sublayer of every encoder layer gains the Query matrix from the earlier masked sublayer. The following steps in decoder layers are similar to encoder layers. Consequently, the output goes through linear and softmax layers from the last decoder layer. The linear layer is a fully connected

neural network that converts the vector created by the stack of a decoder to a much larger vector called a logit vector. Then, softmax turns the scores from linear vector to probability (all positive, all add up to 1.0), and the cell with the highest probability is chosen as the output for this time step. This study applied a transformer model with four encoder-decoder layers, and its architecture is displayed in Fig. 2. This study uses the sequential velocity vector as input and is processed through a determined transformer encoder-decoder to train and predict the following velocity. To measure the model error of the prediction, this study applied to mean absolute error (MAE) and the R^2 score. These two metric measurements are defined by the following equations:³⁷

$$MAE = \frac{1}{N} \sum_{i=1}^N |s_{i,a} - s_{i,p}|, \quad (3)$$

$$R^2 = 1 - \frac{\sum_{i=1}^N (s_{i,a} - s_{i,p})^2}{\sum_{i=1}^N (s_{i,a} - s_m)^2}, \quad (4)$$

where $s_{i,a}$ is the actual measured data, $s_{i,p}$ is the predicted data, s_m is the mean of the actual data, i refer to the corresponding time (or vector array number), and N is the number of the test data.

III. METHODOLOGY

In this section, the design setup of the transformer model is explored, and then, the experimental data from the turbulent flow are described.

A. Transformer model tuning

The model is coded in Python and uses the TensorFlow platform.^{38,39} The transformer model is set up with 4 encoder and decoder layers, which shows the attention mechanism ability and distinction of the transformer model from other sequential architecture. This study has tuned the transformer model with the optimum layers number to present the best result. Indeed, it is possible to build a transformer model with more layers. However, the subject of this work applies an optimum model with a minimum layer number. It must be noted that the primary transformer model, when invented, only had six encoder-decoder.²⁹ Adam is specified as an optimizer.¹⁴ The dataset was normalized by MinMaxScaler transformation.⁴⁰ The MinMaxScaler is a type of scaler that scales the minimum and maximum values to be 0 and 1, respectively.⁴⁰ Since the modeling was implemented on the DevelBooster module⁴¹ from Jewels parallel computing machine, we have applied a distributed strategy application programming interface from the TensorFlow platform abstraction to distribute the training across multiple custom training loops.⁴² The strategy has been set up with four GPUs (Graphics processing units) on one node.

B. Experimental turbulent flow velocity dataset

Each particle has a vector of velocity and displacement during the strain motion. This study proposes a transformer data-driven model for the sequence dataset relying on the velocity. The dataset of this study composed of 2862,119 tracking points for every vector is as follows:

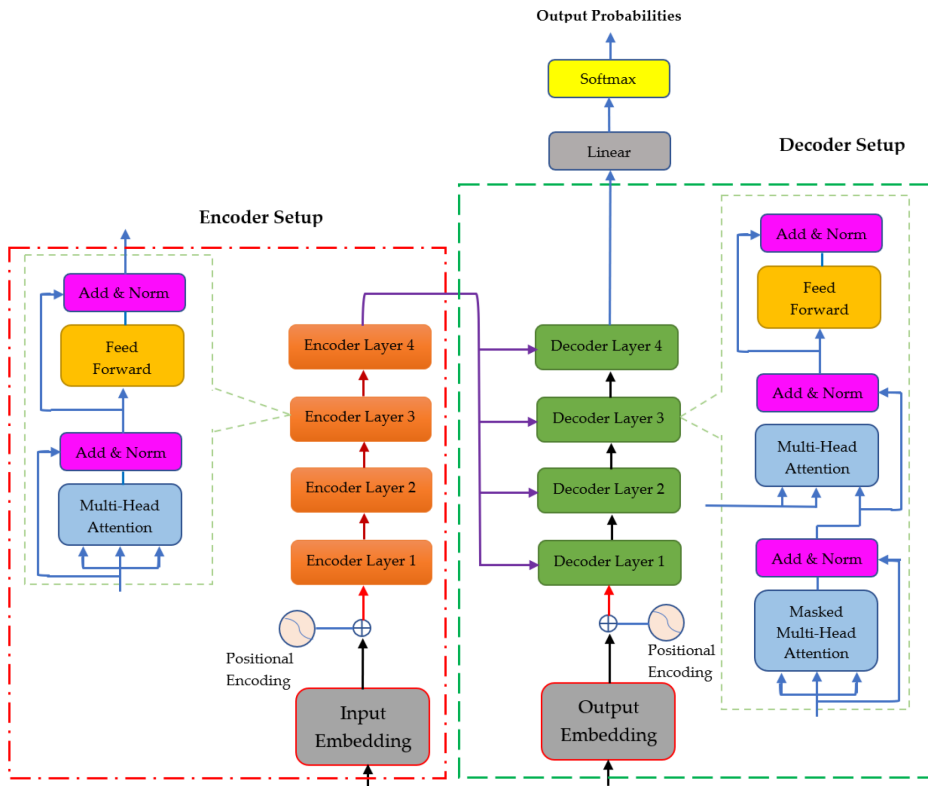


FIG. 2. Transformer model architecture with four encoder-decoder layers.

- Velocity component in the y direction
- Velocity component in the x direction
- Time vector specifies the time t for every tracking point

These tracking points consist of all particles' velocity vectors. Moreover, it is expected to observe several tracking lines, as it is presented for the velocity in the results in Sec. IV; every tracking line specifies a particle. Although the dataset is sequential, it is split into training data and test data for the first model, 80% and 20%, respectively, and for the second model, 60% and 40%, proportionally. Therefore, we assessed the velocity prediction of the following period with the test data for the transformer model.

IV. RESULTS AND DISCUSSION

In this section, the achieved results from the proposed transformer model to forecast the velocity field in a turbulent flow are displayed and discussed. First, the visualization of the measured velocity field will be presented, obtained by the Lagrangian particle tracking.

A. Actual measured velocity

The experimental data from the original work⁴ have been applied to the Lagrangian particle tracking techniques. The recorded data involve the velocity vectors in two directions of the x and y . Figures 3 and 4 are presentations of the velocity in the period of the experiment. Accordingly, the turbulent flow underwent a deformation in the y direction, and it is obvious that the velocity fluctuates much more in the y direction than in other directions. It has been noted in the literature that deformation leads to extra fluctuations in a turbulent flow.^{1,3} In addition, the velocity in the y orientation gains a slope in the mean velocity, which is caused by the mean strain rate value.^{2,4} These two datasets of the velocity fed the transformer model in this study with different training portions, and the observations are reported in the next subsection. For flow with 3D measurement, the third velocity direction will be available and can be performed the same as these two velocity components.

B. Transformer model velocity prediction

The transformer model is trained for each velocity component individually, for the direction of the y once and again the direction of the x . The model has been assessed with two training ratios: start with 80% and then 60% training, and the rest of the data portion has tested

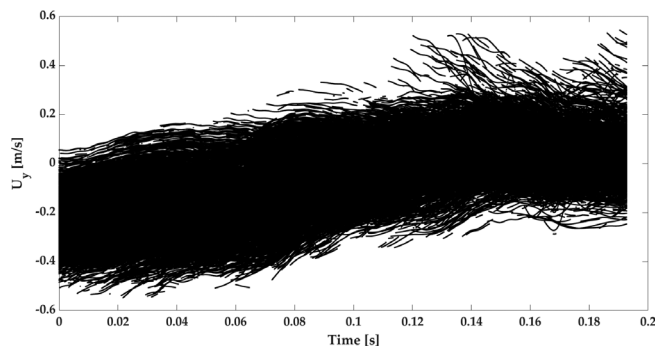


FIG. 3. The measured velocity component in the y direction U_y via Lagrangian particle tracking technique during the deformation.

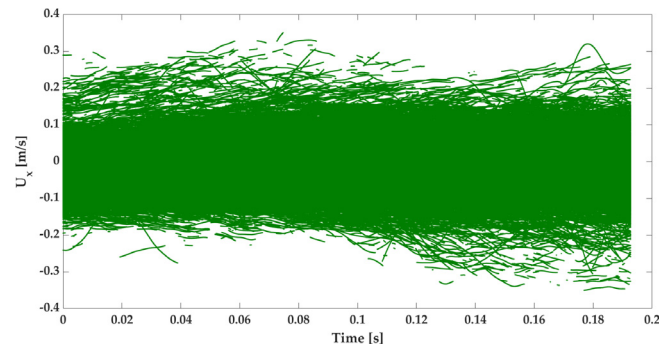


FIG. 4. The measured velocity component in the x direction U_x via Lagrangian particle tracking technique during the deformation.

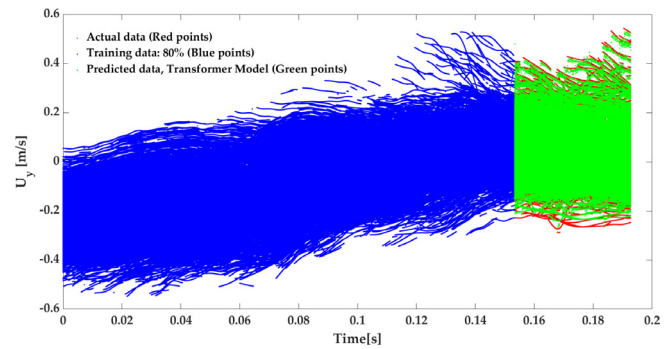


FIG. 5. The prediction of the velocity in the y direction U_y by the proposed transformer model with 80% training data and 20% test data.

the prediction and measured the metric error. Figures 5 and 6 display the transformer model velocity prediction in the y and the x , respectively, which has 80% training data, and 20% are used to test the velocity forecasting. The mean absolute error (MAE) and R^2 score 0.002 and 0.98, respectively. In order to evaluate the transformer model capability with less training, the model again trained with 60% of the data, and 40% are employed for the test. In Figs. 7 and 8, the outcome of the second transformer model training in the y and the x orientation

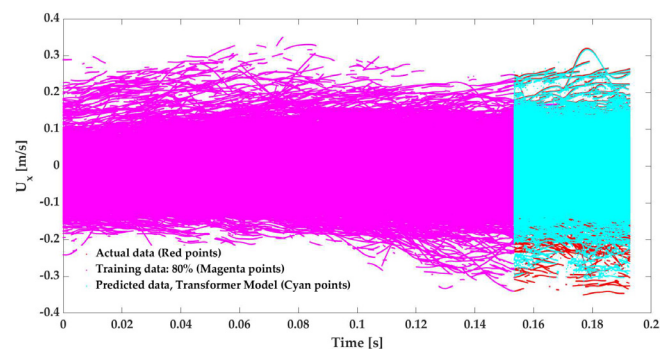


FIG. 6. The prediction of the velocity in the x direction U_x by the proposed transformer model with 80% training data and 20% test data.

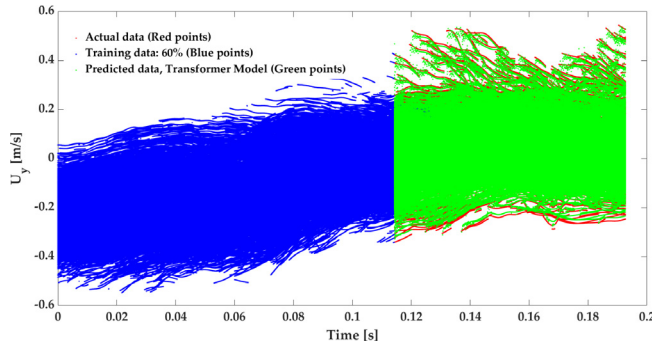


FIG. 7. The prediction of the velocity in the y direction U_y by the proposed transformer model with 60% training data and 40% test data.

is illustrated, respectively. With less training data, the MAE is 0.003, and the R^2 score is 0.98.

It must be noted the applied data from a turbulent flow with identified Taylor microscale Reynolds number range underwent deformation with a specific rate, and the experiment was conducted in the presence of gravity. In this proposed transformer model, training data did not have any direct information regarding the turbulent intensity, strain rate, and gravity effects. This is the superiority of the transformer model that can extract the dominant feature and its dependencies. Moreover, in this model, the study only applied the velocity vector as an inherent flow feature, which carried the flow properties, and can transfer these crucial features to the model and predict the next period of the turbulent flow. The predicted velocity field by the transformer model remarkably matches the actual available data based on the MAE and the R^2 score.

C. Sequential dataset and transformer model

To evaluate the transformer model as an attention mechanism in comparison to well-established sequential deep learning models, its performance is compared to LSTM and GRU models utilized in previous studies with similar datasets, focusing on physical properties and size.²³ Table I presents the results, showcasing that the transformer model achieves comparable mean absolute error (MAE), R^2 scores, and training time in predicting the velocity of strained turbulent flow

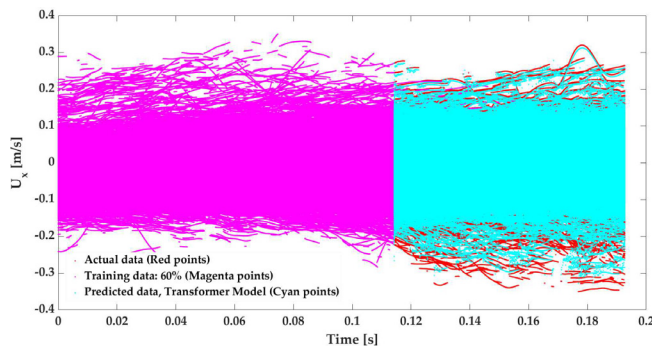


FIG. 8. The prediction of the velocity in the x direction U_x by the proposed transformer model with 60% training data and 40% test data.

TABLE I. To assess the capability of the transformer model as a mechanism for attention, a comparison is made between its performance and that of long short-term memory (LSTM)²³ and gated recurrent units (GRU)²³ from previous studies with similar datasets.

Training ratio	Performance	LSTM	GRU	Transformer
80%	MAE	0.002	0.002	0.002
	R^2 score	0.98	0.98	0.98
	Training time (s)	295	318	301
60%	MAE	0.002	0.002	0.003
	R^2 score	0.98	0.98	0.98
	Training time (s)	214	229	219

when compared to LSTM and GRU models. The LSTM and GRU models in Table I utilized identical training and test ratios as the transformer model employed in this study. LSTM and GRU models have been widely employed as sequential models in numerous studies, making them suitable benchmarks for comparison.

While the use of transformer models in the field of fluid dynamics has been relatively limited compared to the widespread adoption of LSTM and GRU models, Table I demonstrates the competence of the transformer model for such applications. However, further enhancements can be made by leveraging a larger and more diverse dataset specific to fluid dynamics. Moreover, the original work introducing the transformer model²⁹ highlights its parallelizability and significantly reduced training time compared to traditional models. This inherent characteristic warrants evaluation with a larger-scale dataset to fully exploit its potential benefits in fluid dynamics research.

D. Transformer and sequential models in turbulent flow and physics applications

The primary challenge of the present century lies in enhancing deep learning models, including transformers, to tackle turbulent flow and accurately predict its features. This advancement holds the potential to provide a comprehensive understanding of turbulent flow applications, thereby offering valuable insights and advancements in various domains.

One crucial application is the utilization of wind energy, where the inherent relationship between wind speed and turbulence plays a significant role. Accurate long-term and short-term forecasting of turbulent wind patterns can greatly enhance the reliability and stability of power grids, contributing to the pursuit of sustainable and efficient energy systems.

Another critical area is the study of turbulent flow over airplane wings. Understanding the intricate features and making precise predictions in this domain is essential for addressing the challenge of reducing drag force. Such advancements are instrumental in realizing goals related to green energy and minimizing fuel consumption on a broader scale.

Particle-laden turbulent flow represents an open frontier in fluid dynamics. By harnessing the capabilities of deep learning models like transformers, it becomes feasible to forecast the trajectories of particles in subsequent periods. Additionally, these models can shed light on crucial physical concepts, such as the impact of gravity, which may have been inadequately explored using traditional numerical methods.

In the field of combustion, understanding reactive flows holds great significance for controlling, predicting, and optimizing the conversion processes. Recent attention has been directed toward alternative fuels such as hydrogen, where experimental studies can be complex and costly. Leveraging the capabilities of Transformers and other sequential deep-learning models can lead to remarkable advancements in this area, revolutionizing the exploration and utilization of alternative fuels.

In summary, the integration of deep learning models, particularly transformers, into the study of turbulent flow has the potential to drive substantial progress in understanding complex flow dynamics, unlocking valuable insights, and enabling advancements in a wide range of applications.

E. Limits and enhancing Transformer and sequential models in turbulent flow

The study at hand presents a deep learning-based model utilizing the transformer architecture to forecast the velocity of deformed turbulent flow under specific conditions, encompassing the effects of gravity, defined turbulent intensity, and determined strain rate. Experimental data obtained through the LPT technique were employed as the dataset. However, compared to other deep learning methods such as LSTM, GRU, and CNN, the application of the transformer model in predicting turbulent flow is still an active area of research. It is crucial to understand the capabilities and limitations of this model in the context of turbulent flow, particularly when dealing with high Reynolds numbers characterized by heightened fluctuations. Additionally, the flow characteristics differ depending on whether it is compressible or incompressible. Previous studies applying deep learning techniques to turbulent flow have often focused on specific data or narrow ranges of Reynolds numbers. Consequently, further investigation is necessary to identify the limitations and failure points of these models in prediction tasks. It should be noted that each deep learning model applied to turbulent flow requires specific tuning, and there is no universally applicable setup. Recent developments in deep learning, such as hyperparameter tuning, have underscored its significance in optimizing model performance. Incorporating this technique can aid in identifying the most suitable model design for a given task. Expanding on the proposed model, this study suggests exploring the potential of enhancing deep learning models, including variants of LSTM, Transformer, and CNN, across a wider range of turbulent flow scenarios. Future research endeavors should strive to uncover the working range and performance boundaries of these models in turbulent flow applications.

V. CONCLUSION

This study proposed a novel transformer model from DL approaches in the context of data-driven concepts to predict the velocity field of a turbulent flow with deformation and in the presence of gravity from an experiment. Transformer model architecture is based on encoder-decoder layers and processes the data via an attention mechanism. The transformer model is state-of-the-art for sequential models and is mostly applied in language translation, and it made remarkable development in this area. In the realm of turbulent flow, the application of the transformer model is relatively nascent compared to established methods such as LSTM variants and CNN compositions. However, recent studies in fluid dynamics have

demonstrated significant advancements and notable precedents in utilizing the transformer model.^{31,43} Long short-term memory and convolutional neural networks are employed in many data-driven and compute-driven fluid dynamics areas.

The application of deep learning models in the field of fluid dynamics, specifically in turbulent flow, has emerged as a crucial area of study. In various fields of fluid dynamics, a combination of analytical, numerical, and artificial intelligence methods has gained traction. Several examples include analyzing flow on airplane wings, reactive flow phenomena, wind speed prediction in wind turbines, studying multiphase flow dynamics, investigating boundary layers, and exploring particle-laden turbulent flow. This integration highlights the increasing recognition of artificial intelligence as a powerful tool alongside traditional methods, empowering researchers to tackle complex fluid dynamics challenges and could open up new avenues for understanding and predicting complex flow phenomena. These models have the potential to uncover a wealth of physical concepts and facilitate accurate flow feature predictions in various domains, contributing to advancements in energy, transportation, and environmental research.

This work used only the velocity components of 2D turbulent flow measurement in a sequence way. It did not feed the other effects, such as turbulence intensity, strain rate, and gravity effect, to the model. This design relies on the concept that velocity carries and transfers the most important flow features. Moreover, the velocity measurement of the turbulent flow by devices is available in many industrial and natural applications; therefore, applying the suggested method to predict the turbulent velocity is convenient. Moreover, the model is independent of the velocity components and trains and predicts each velocity component individually. The transformer model was trained with two portions of training data, 80%, and 60%, respectively, and the rest of the data tested the velocity prediction. The error measurement illustrates the MAE and the R^2 score in 0.002–0.003 and 0.98, respectively, which is a considerable prediction and almost matches the actual data. It is extraordinary that with less training data, the transformer model is able to keep the error and prediction quantity constant and predict a period similar to a larger training ratio. It proves that the capability of the Transformer model is excellent. For future studies, it is suggested to investigate the transformer model with extensive data to evaluate its computation cost. Moreover, other turbulent flow features can be taken into model consideration and prediction. Based on the Transformer architecture and its capability, it could hopefully be useful for a deeper physical understanding of the turbulent phenomenon. The suggested method in this study could be employed broadly.

In light of the proposed model, this study advocates for an extensive exploration of deep learning models, such as various LSTM variants, transformer architectures, and CNNs, to unlock their full potential in a broader spectrum of turbulent flow scenarios. It is imperative for future research efforts to go beyond the current boundaries and unravel the working range and performance limits of these models in turbulent flow applications. By pushing the boundaries of deep learning techniques, we can gain deeper insights into their applicability and efficacy, paving the way for advancements in turbulent flow prediction and analysis. This comprehensive investigation will contribute to a more thorough understanding of the capabilities and limitations of deep learning models, allowing for their optimal utilization in real-world turbulent flow scenarios.

ACKNOWLEDGMENTS

This work was performed in the Center of Excellence (CoE) Research on AI and Simulation-Based Engineering at Exascale (RAISE) and the EuroCC 2 projects receiving funding from EU's Horizon 2020 Research and Innovation Framework Programme and European Digital Innovation Hub Iceland (EDIH-IS) under Grant Agreement Nos. 951733, 101101903, and 101083762, respectively. We thank Mr. Lahcen Bouhlali from Reykjavik University for his experimental work in data preparation.

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

Reza Hassanian: Conceptualization (equal); Methodology (equal); Resources (equal); Software (equal); Visualization (equal); Writing – original draft (equal). **Hemanadhan Myneni:** Software (equal); Writing – review & editing (equal). **Asdis Helgadottir:** Methodology (equal); Writing – review & editing (equal). **Morris Riedel:** Methodology (equal); Supervision (equal); Writing – review & editing (equal).

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author upon reasonable request.

REFERENCES

- ¹H. T. John and L. Lumley, *A First Course in Turbulence* (MIT Press, Massachusetts, 1972).
- ²S. B. Pope, *Turbulent Flows* (Cambridge University Press, London, 2000).
- ³P. A. Davidson, *Turbulence: An Introduction for Scientists and Engineers* (Oxford University Press, London, 2004).
- ⁴R. Hassanian, A. Helgadottir, L. Bouhlali, and M. Riedel, “An experiment generates a specified mean strained rate turbulent flow: Dynamics of particles,” *Phys. Fluids* **35**, 015124 (2023).
- ⁵R. Hassanian and M. Riedel, “Leading-edge erosion and floating particles: Stagnation point simulation in particle-laden turbulent flow via lagrangian particle tracking,” *Machines* **11**, 566 (2023).
- ⁶G. I. Taylor, “The interaction between experiment and theory in fluid mechanics,” *Annu. Rev. Fluid Mech.* **6**, 1–17 (1974).
- ⁷S. Goldstein, “Fluid mechanics in the first half of this century,” *Annu. Rev. Fluid Mech.* **1**, 1–29 (1969).
- ⁸S. Patankar, *Numerical Heat Transfer and Fluid Flow* (CRC Press, New York, 1980).
- ⁹J. Anderson, *Computational Fluid Dynamics* (McGraw-Hill Education, New York, 1995).
- ¹⁰H. Versteeg and W. Malalasekera, *Introduction to Computational Fluid Dynamics, an: The Finite Volume Method* (Pearson, London, 2007).
- ¹¹T. Kajishima and K. Taira, *Computational Fluid Dynamics: Incompressible Turbulent Flows* (Springer Cham, New York, 2017).
- ¹²R. Hassanian, M. Riedel, and L. Bouhlali, “The capability of recurrent neural networks to predict turbulence flow via spatiotemporal features,” in *2022 IEEE 10th Jubilee International Conference on Computational Cybernetics and Cyber-Medical Systems (ICCC)* (IEEE, 2022), pp. 000335–000338.
- ¹³S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Comput.* **9**, 1735–1780 (1997).
- ¹⁴A. Géron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems* (O'Reilly Media, Sebastopol, 2019).
- ¹⁵I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning* (The MIT Press, Massachusetts, 2016).
- ¹⁶T. Liu, L. Zhou, H. Tang, and H. Zhang, “Mode interpretation and force prediction surrogate model of flow past twin cylinders via machine learning integrated with high-order dynamic mode decomposition,” *Phys. Fluids* **35**(2), 023611 (2023).
- ¹⁷K.-E. Otmani, G. Ntoukas, O. A. Mariño, and E. Ferrer, “Towards a robust detection of viscous and turbulent flow regions using unsupervised machine learning,” *Phys. Fluids* **35**(2), 027112 (2023).
- ¹⁸H. Salehipour and W. R. Peltier, “Deep learning of mixing by two ‘atoms’ of stratified turbulence,” *J. Fluid Mech.* **861**, R4 (2019).
- ¹⁹M. Raissi, Z. Wang, M. S. Triantafyllou, and G. E. Karniadakis, “Deep learning of vortex-induced vibrations,” *J. Fluid Mech.* **861**, 119–137 (2019).
- ²⁰H. Kim, J. Kim, S. Won, and C. Lee, “Unsupervised deep learning for super-resolution reconstruction of turbulence,” *J. Fluid Mech.* **910**, A29 (2021).
- ²¹M. Z. Yousif, L. Yu, and H. Lim, “Physics-guided deep learning for generating turbulent inflow conditions,” *J. Fluid Mech.* **936**, A21 (2022).
- ²²S. Lee and D. You, “Data-driven prediction of unsteady flow over a circular cylinder using deep learning,” *J. Fluid Mech.* **879**, 217–254 (2019).
- ²³R. Hassanian, A. Helgadottir, and M. Riedel, “Deep learning forecasts a strained turbulent flow velocity field in temporal lagrangian framework: Comparison of LSTM and GRU,” *Fluids* **7**, 344 (2022).
- ²⁴H. Eivazi, M. Tahani, P. Schlatter, and R. Vinuesa, “Physics-informed neural networks for solving Reynolds-averaged Navier-Stokes equations,” *Phys. Fluids* **34**, 075117 (2022).
- ²⁵C. Duru, H. Alemdar, and O. U. Baran, “A deep learning approach for the transonic flow field predictions around airfoils,” *Comput. Fluids* **236**, 105312 (2022).
- ²⁶S. R. Bukka, R. Gupta, A. R. Magee, and R. K. Jaiman, “Assessment of unsteady flow predictions using hybrid deep learning based reduced-order models,” *Phys. Fluids* **33**, 013601 (2021).
- ²⁷R. Hassanian, A. Helgadottir, and M. Riedel, “Parallel computing accelerates sequential deep networks model in turbulent flow forecasting,” in the *International Conference for High Performance Computing, Networking, Storage, and Analysis, SC22*, Dallas TX, USA, 13–18 November 2022 (IEEE, Piscataway, NJ, 2022); available at https://sc22.supercomputing.org/proceedings/tech_poster/tech_poster_pages/rpost142.html.
- ²⁸R. Hassanian, M. Riedel, A. Helgadottir, P. Costa, L. Bouhlali *et al.*, “Lagrangian particle tracking data of a straining turbulent flow assessed using machine learning and parallel computing,” in *33rd International Conference on Parallel Computational Fluid Dynamics*, Alba, Italy, 25–27 May 2022 (ParCFD, 2022); available at <https://hdl.handle.net/20.500.11815/3515>.
- ²⁹A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. U. Kaiser, and I. Polosukhin, “Attention is all you need,” in *Proceedings of the 31st International Conference on Neural Information Processing Systems* (Curran Associates Inc, 2017), pp. 11; available at <https://dl.acm.org/doi/10.5555/3295222.3295349>.
- ³⁰Q. Xu, Z. Zhuang, Y. Pan, and B. Wen, “Super-resolution reconstruction of turbulent flows with a transformer-based deep learning framework,” *Phys. Fluids* **35**, 055130 (2023).
- ³¹N. Geneva and N. Zabarar, “Transformers for modeling physical systems,” *Neural Networks* **146**, 272–289 (2022).
- ³²A. Hemmasian and A. Barati Farimani, “Reduced-order modeling of fluid flows with transformers,” *Phys. Fluids* **35**, 057126 (2023).
- ³³N. T. Ouellette, H. Xu, and E. Bodenschatz, “A quantitative study of three-dimensional lagrangian particle tracking algorithms,” *Exp. Fluids* **40**, 301–313 (2006).
- ³⁴L. Brandt and F. Coletti, “Particle-laden turbulence: Progress and perspectives,” *Annu. Rev. Fluid Mech.* **54**, 159–189 (2022).
- ³⁵R. F. Miotto and W. R. Wolf, “Flow imaging as an alternative to non-intrusive measurements and surrogate models through vision transformers and convolutional neural networks,” *Phys. Fluids* **35**, 045143 (2023).
- ³⁶P. Wu, F. Qiu, W. Feng, F. Fang, and C. Pain, “A non-intrusive reduced order model with transformer neural network and its application,” *Phys. Fluids* **34**, 115130 (2022).
- ³⁷C. Gu and H. Li, “Review on deep learning research and applications in wind and wave energy,” *Energies* **15**, 1510 (2022).
- ³⁸M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin *et al.*, “Tensorflow: Large-scale machine learning on heterogeneous distributed systems,” [arXiv:1603.04467](https://arxiv.org/abs/1603.04467) (2016).
- ³⁹M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard *et al.*, “Tensorflow: A system for large-scale machine learning” in *12th USENIX Symposium on Operating Systems Design*

and Implementation (OSDI 16) (OSDI, 2016), pp. 265–283; available at <https://www.usenix.org/conference/osdi16/technical-sessions/presentation/abadi>.

⁴⁰O. Kramer and O. Kramer, “Scikit-learn,” in *Machine Learning for Evolution Strategies* (Springer, 2016), pp. 45–53.

⁴¹M. Riedel, R. Sedona, C. Barakat, P. Einarsson, R. Hassanian, “G. Cavallaro, M. Book, H. Neukirchen, and A. Lintermann, Practice and experience in using parallel and scalable machine learning with heterogenous modular

supercomputing architectures,” *Proceedings of the 2021 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)*, Portland OR, USA, 17–21 June 2021 (IPDPSW, 2021).

⁴²TensorFlow, *Tensorflow Core Tutorials* (TensorFlow, 2022).

⁴³M. Z. Yousif, M. Zhang, L. Yu, R. Vinuesa, and H. Lim, “A transformer-based synthetic-inflow generator for spatially developing turbulent boundary layers,” *J. Fluid Mech.* **957**, A6 (2023).