

PHOTONICS Research

Highly robust spatiotemporal wavefront prediction with a mixed graph neural network in adaptive optics

JU TANG,^{1,†}  JI WU,^{1,†} JIAWEI ZHANG,¹ MENG MENG ZHANG,¹ ZHENBO REN,^{1,2,5}  JIANGLEI DI,^{1,3,6} 
LIUSEN HU,⁴ GUODONG LIU,⁴ AND JIANLIN ZHAO^{1,7} 

¹Key Laboratory of Light Field Manipulation and Information Acquisition, Ministry of Industry and Information Technology, and Shaanxi Key Laboratory of Optical Information Technology, School of Physical Science and Technology, Northwestern Polytechnical University, Xi'an 710129, China

²Research & Development Institute of Northwestern Polytechnical University in Shenzhen, Shenzhen 518063, China

³Key Laboratory of Photonic Technology for Integrated Sensing and Communication, Ministry of Education, and Guangdong Provincial Key Laboratory of Information Photonics Technology, Guangdong University of Technology, Guangzhou 510006, China

⁴Institute of Fluid Physics, China Academy of Engineering Physics, Mianyang 621900, China

⁵e-mail: zbren@nwpu.edu.cn

⁶e-mail: jiangleidi@gdut.edu.cn

⁷e-mail: jlzhao@nwpu.edu.cn

[†]These authors contributed equally to this work.

Received 13 June 2023; revised 6 August 2023; accepted 19 August 2023; posted 21 August 2023 (Doc. ID 497909); published 5 October 2023

The time-delay problem, which is introduced by the response time of hardware for correction, is a critical and non-ignorable problem of adaptive optics (AO) systems. It will result in significant wavefront correction errors while turbulence changes severely or system responses slowly. Predictive AO is proposed to alleviate the time-delay problem for more accurate and stable corrections in the real time-varying atmosphere. However, the existing prediction approaches either lack the ability to extract non-linear temporal features, or overlook the authenticity of spatial features during prediction, leading to poor robustness in generalization. Here, we propose a mixed graph neural network (MGNN) for spatiotemporal wavefront prediction. The MGNN introduces the Zernike polynomial and takes its inherent covariance matrix as physical constraints. It takes advantage of conventional convolutional layers and graph convolutional layers for temporal feature catch and spatial feature analysis, respectively. In particular, the graph constraints from the covariance matrix and the weight learning of the transformation matrix promote the establishment of a realistic internal spatial pattern from limited data. Furthermore, its prediction accuracy and robustness to varying unknown turbulences, including the generalization from simulation to experiment, are all discussed and verified. In experimental verification, the MGNN trained with simulated data can achieve an approximate effect of that trained with real turbulence. By comparing it with two conventional methods, the demonstrated performance of the proposed method is superior to the conventional AO in terms of root mean square error (RMS). With the prediction of the MGNN, the mean and standard deviation of RMS in the conventional AO are reduced by 54.2% and 58.6% at most, respectively. The stable prediction performance makes it suitable for wavefront predictive correction in astronomical observation, laser communication, and microscopic imaging. © 2023 Chinese Laser Press

<https://doi.org/10.1364/PRJ.497909>

1. INTRODUCTION

Adaptive optics (AO), which can compensate wavefront distortions, has wide applications in astronomical observation [1], laser communication [2], and biological imaging [3]. For a typical AO system, a wavefront sensor (WFS) measures the wavefront distorted by atmospheric turbulence, and a controller converts the distorted wavefront into the voltage of a deformable mirror (DM) for correction. According to the phase conjugate principle, the subsequent wavefront is compensated by the DM immediately

in a closed-loop correction [4]. However, the AO system always has a time delay between the measurement and compensation, due to the conversion and transmission of measured signals, as well as the response of the DM. Generally, the time lapse is <20 ms, which is basically equivalent to two to three captured frames of the WFS [5]. Therefore, the wavefront correction mismatch will significantly increase correction errors, especially when the turbulence changes rapidly. Facing some ever-changing turbulence scenarios, such as fast-moving orbital objects and high-speed

laser communication, it is more essential and critical to alleviate the time-delay problem for stable performance [6,7]. Hence, the predictive AO technique, which predicts future wavefronts from earlier ones, is proposed to cover the delayed time for synchronous correction.

Although the evolution of turbulence is random and unpredictable on a long-term scale, it is still continuous and predictable within a short time interval. Its statistical temporal characteristics appear to be particularly important in predictive AO. Since Jorgenson and Aitken demonstrated the predictability of atmospheric wavefront [8], many prediction methods have been proposed to improve the prediction accuracy. Current approaches can be categorized into two groups: linear and non-linear. Linear prediction (LP) algorithms, such as the linear minimum mean square error (LMMSE), were investigated first based on the statistical knowledge of atmosphere [9,10]. By establishing the linear equations between the previous and future wavefronts, the corresponding transfer matrix for distortion compensation can be calculated quickly [11]. However, the matrix is constant and hence it cannot be used to correct the dynamic atmospheric turbulence. Therefore, it needs recalculation when the atmospheric conditions change, resulting in its inapplicability for real applications. Another Fourier transform prediction in LP algorithms [12] is established based on Taylor's frozen flow hypothesis. It needs to dynamically monitor the windspeed along the propagation path, increasing the complexity of the AO system. Overall, these LP algorithms are difficult to handle the non-linear relationships of turbulence, keeping them away from optimal performance.

In contrast, non-linear prediction algorithms, which are always machine-learning-based, extract the non-linear features to improve accuracy of prediction. They are mainly divided into two types: time-series prediction and spatiotemporal prediction. The former applies to multivariate data on time series, such as the wavefront slope [13], Zernike coefficients [14], and DM voltage [15]. Noteworthy, the multi-layer perceptron (MLP) has been explored for turbulence prediction since 1992 [8]. However, the MLP overly focused on the regression fitting and relatively ignored the continuity of temporal features. More recent long short-term memory (LSTM) can partially make up for this weakness [16]. Disappointingly, the lack of spatial constraints causes the distribution of prediction not to match the true wavefront. Comparatively, spatiotemporal prediction algorithms pay more attention to spatial features, and the conventional convolutions are useful to deal with the structured discrete signals. The well-known convolutional neural network (CNN) is involved in dealing with sequential wavefront phases for better corrections. Chen *et al.* established a U-shaped CNN to predict subsequent eight frames [17]. Swanson *et al.* used the convolutional LSTM to extract the temporal and spatial features simultaneously and improved the Strehl ratio by over 55% compared with classical methods [18,19]. In our previous work, we upgraded the convolutional LSTM with the attention structure, and reduced the phase error of conventional AO by 80%

at most in one-frame correction [20]. However, the distribution of convolutional output only follows the statistical spatial patterns from limited data. For prediction applications, it may be inadequate and may not conform to the real turbulence due to the lack of physical constraints. Taking the two-dimensional wavefront phase as input also greatly increases the computational complexity of prediction. In addition, the prediction robustness to various data still needs to be verified and further enhanced, which is crucial but seldom reported previously.

To deal with the time-delay problem in AO, inspired by the widely used deep learning [21–23], we propose a highly robust spatiotemporal wavefront prediction algorithm with the mixed graph neural network (MGNN) in predictive AO. To better extract and utilize the spatiotemporal non-linear features of turbulence, the temporal feature catcher (TFC) and spatial feature analyzer (SPA) are both designed. In the feature extraction process, conventional convolution is no longer applicable to graph structure, so the graph convolution is introduced in the structure. Innovatively, we delve into the Zernike model and discover the consistency between the covariance matrix of coefficients and the graph structure. The covariance matrix is a statistical invariant in the real time-varying atmosphere and reveals the relationship between Zernike coefficients of real turbulence. Therefore, we incorporate the consistency to promote graph constraints of the covariance matrix and weight learning of the transformation matrix internally. With the optimization of graph structure, the MGNN can learn more spatial features than common networks from limited data and can generate output with high resemblance to real turbulence. The prediction accuracy of the MGNN and the robustness to varying atmospheric conditions, including the generalization from simulation to experiment, are discussed and verified. In experimental verification, the MGNN trained with simulated data can achieve an approximate effect of that trained with real turbulence. As a result, our proposed algorithm can achieve lower correction error and better robustness than conventional predictive AO algorithms, let alone conventional AO.

2. PRINCIPLE

A. Covariance Matrix of Zernike Polynomial Model

According to the Zernike polynomial, the wavefront phase measured by the WFS can be expressed as

$$\varphi(t) = \sum_j^m a_j(t) Z_j = \mathbf{A} \cdot \mathbf{Z}, \quad (1)$$

where m is the total orders of polynomials to describe the wavefront, Z_j is the j th Zernike polynomial, a_j is the corresponding coefficient varying with time t , $\mathbf{A} = [a_1, a_2, \dots, a_m]$ is the coefficient vector, and $\mathbf{Z} = [Z_1, Z_2, \dots, Z_m]^T$ is the polynomial vector. According to Noll's derivation [24], the covariance value between any two coefficients a_w and a_v in \mathbf{A} of one frame can be expressed as

$$E(a_w, a_v) = \frac{\delta K_{wv}(Z_w, Z_v) \Gamma[(n_w + n_v - 5/3)/2] (D/r_0)^{5/3}}{\Gamma[(n_w - n_v + 17/3)/2] \Gamma[(n_v - n_w + 17/3)/2] \Gamma[(n_w + n_v + 23/3)/2]}, \quad (2)$$

where δ is the covariance parameter with a value of 0 or 1, K_{wv} is the frequency characteristic factor depending on the polynomial terms Z_w and Z_v , n_w and n_v are the radial orders of Z_w and Z_v , D is the optical diameter of system, r_0 is the Fried parameter of atmosphere, and $\Gamma[\cdot]$ is the gamma function. According to Eq. (2), the coefficients are statistically correlated, and depend on each other. Thus, ignoring the piston term Z_1 , the covariance matrix \mathbf{C} of Zernike coefficients can be expressed as

$$\mathbf{C} = E(\mathbf{A}^T \cdot \mathbf{A}) = \begin{bmatrix} E(a_2, a_2) & 0 & \dots & 0 \\ 0 & E(a_3, a_3) & \dots & E(a_w, a_v) \\ \vdots & \vdots & \ddots & \vdots \\ 0 & E(a_v, a_w) & \dots & E(a_m, a_m) \end{bmatrix} \quad (3)$$

Ignoring the scaling factor $(D/r_0)^{5/3}$ in Eq. (2), \mathbf{C} is a statistical invariant between these Zernike coefficients from turbulence. The elements of its off-diagonal are non-zero, and the complete representation is in Appendix A. The relevance of coefficients is non-negligible, but it does not meet our expectations because it makes the coefficients prediction more complex. Therefore, we introduce the Karhunen–Loève (K-L) functions [25], and the wavefront phase of one frame can be also expressed as

$$\varphi = \sum_i^m b_i K_i = \mathbf{B} \cdot \mathbf{K}, \quad (4)$$

$$K_i = \sum_j V_{ij} Z_j, \quad (5)$$

where b_i is the random coefficient varying independently, and K_i is the i th K-L polynomial, which can be expanded as Eq. (5). \mathbf{B} and \mathbf{K} are corresponding vectors like \mathbf{A} and \mathbf{Z} . Consequently, according to Eqs. (1), (4), and (5), we can

get the desired coefficient vector \mathbf{A} from the independent vector \mathbf{B} and the transformation matrix \mathbf{V} as

$$\mathbf{A} = \mathbf{B} \cdot \mathbf{V}. \quad (6)$$

Specifically, the covariance matrix $\mathbf{S} = E(\mathbf{B}^T \cdot \mathbf{B})$ is a diagonal matrix due to the independence of elements and has a relationship with \mathbf{C} as $\mathbf{S} = \mathbf{V}\mathbf{C}\mathbf{V}^T$.

B. Spatiotemporal Wavefront Prediction Model

In the predictive AO system, we can get a sequence of coefficient vectors $\{\mathbf{A}_d(1), \mathbf{A}_d(2), \dots, \mathbf{A}_d(t)\}$ over time easily by the WFS. These time-continuous vectors of historical wavefronts form the inputs of the prediction algorithm, and the prediction model can be expressed as

$$\mathbf{A}_p(t_0 + F) = f[\mathbf{A}_d(t_0 - G + 1), \dots, \mathbf{A}_d(t_0)], \quad (7)$$

$$\hat{\mathbf{A}}_p = \arg \min \sum |\mathbf{A}_f(t_0 + F) - \mathbf{A}_p(t_0 + F)|, \quad (8)$$

where $f(\cdot)$ represents the prediction algorithm, \mathbf{A}_p is the predicted vector, \mathbf{A}_d and \mathbf{A}_f are vectors of measured and future wavefronts, G is the number of frames for input, t_0 is the present moment, and $t_0 + F$ is the predicted moment. The time t is represented and divided by the captured frames of the WFS.

According to the prediction model, we establish the flowchart of spatiotemporal wavefront prediction in Fig. 1. In fact, we can only collect the continuous turbulence for a while and separate them into \mathbf{A}_d and \mathbf{A}_f based on different moments for optimization. These separated vectors are paired one by one and assembled to compose the training dataset in Fig. 1(a). For predictive AO, its prediction timespan G must cover the time delay of measurement and correction, and the calculation delay of the algorithm. For an AO system with a frame rate of hundreds of hertz, the actual response time is often 2–3 times of the sampling period of a camera. In other words, its time delay is two to three frames, so we set $F = 3$ in our algorithm for

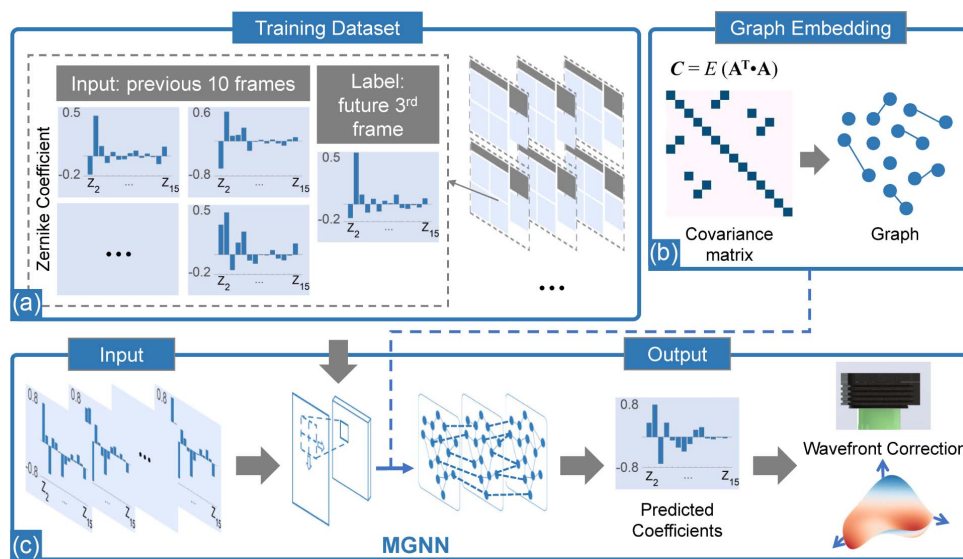


Fig. 1. Flowchart of spatiotemporal wavefront prediction. (a) Training dataset configuration. (b) Graph embedding from the covariance matrix of coefficients. (c) In the testing process, Zernike coefficients of previous 10 frames are inputs, and the predicted coefficients are obtained by the trained MGNN.

prediction and compensation. The number of previous frames is set as $G = 10$ for balance, because more frames represent larger data volume and longer processing time. According to Eq. (3), the covariance matrix \mathbf{C} , which is highly similar to the adjacency matrix of a graph, represents the connections between coefficients. So, we can regard coefficients as nodes and describe their connections with edges in a graph. In Fig. 1(b), the graph embedding is performed from the covariance matrix \mathbf{C} , applying physical constraints to the predicted coefficients in the MGNN. The graph embedding process is expressed as

$$\mathbf{M} = \left(1 + \sum_j \mathbf{C}_{ij} \right)^{-1} (\mathbf{C} + \mathbf{I}), \quad (9)$$

where \mathbf{I} is a unity matrix, and \mathbf{M} is the adjacency matrix of the initial graph. After full optimization of the MGNN, we take vectors of previous 10 frames as inputs for testing, and its outputs are the predicted coefficients of future third frame, which can be used for wavefront correction. In short, the MGNN consists of two parts: conventional convolutional layers and graph convolutional layers [26]. The former wholly catches temporal features, while the latter comprehensively analyzes spatial features. Therefore, the proposed MGNN fully utilizes spatiotemporal characteristics of limited data for wavefront prediction.

C. Mixed Graph Neural Network

The architecture of the MGNN is presented in Fig. 2. Input tensors of size (B, C, W, H) are successively processed by the start convolutional layer, TFC, SFA, and end convolutional layer in Fig. 2(a). The start and end convolutional layers consist of several conventional convolutional operations with 1×1 kernel. $B, C, W,$ and H are the batch size, channel, width, and height of the tensors ($B = 32, C = 1, W = 14,$

and $H = 10$). The width W depends on the orders of polynomials to describe one-frame wavefront, and the height H depends on the number of previous frames G before. In Fig. 2(b), the TFC applies two branches of inception units to extract high-level temporal features. Inspired by GoogleNet [27], the inception unit combines four convolutional operations with different receptive fields for discovering different temporal patterns. One branch is followed by a “tanh” activation function, while the other takes “sigmoid” to control information flow like the gate function in the LSTM [18]. The inception unit is repeated five times in succession while Fig. 2(b) only presents one. These features fuse together after calculating Hadamard’s product. With different receptive fields, the TFC can extract the characteristics as a whole from the coefficient tensors of previous frames, and predict the future coefficients based on temporal information. However, the predicted features of the TFC may not be real or effective, due to the assumption of coefficient independence in the TFC. It is necessary to modulate the predicted features reasonably, so we propose the SFA in Fig. 2(c). The SFA also applies two branches of graph units to optimize spatial features. With the adjacent matrix \mathbf{M} in Eq. (9), one branch of the graph unit layer can be expressed as

$$\begin{cases} \mathbf{F}^{(i)} = \beta \mathbf{F}_{\text{in}} + (1 - \beta) \mathbf{M} \cdot \mathbf{F}^{(i-1)}, & i = 1, 2, \dots, k \\ \mathbf{F}^{(0)} = \mathbf{F}_{\text{in}}, & i = 0 \end{cases}, \quad (10)$$

where k is the depth of the layer, \mathbf{F}_{in} is the input feature of this layer, β is a hyperparameter to control ratio, and “ \cdot ” is matrix multiplication. All features $\mathbf{F}^{(i)}$ are concatenated together, and then a linear layer is created to get the output feature of this graph convolutional layer given by

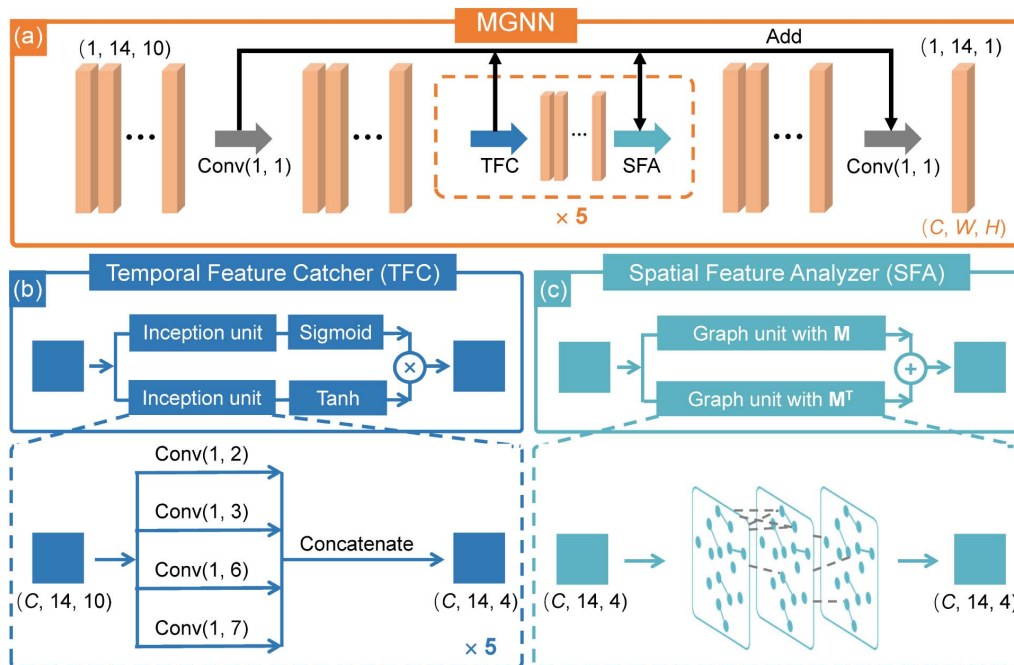


Fig. 2. Network architecture of the MGNN. (a) Overall framework. (b) Details of temporal feature catcher. (c) Details of spatial feature analyzer. (C, W, H) are channel, width, and height, respectively.

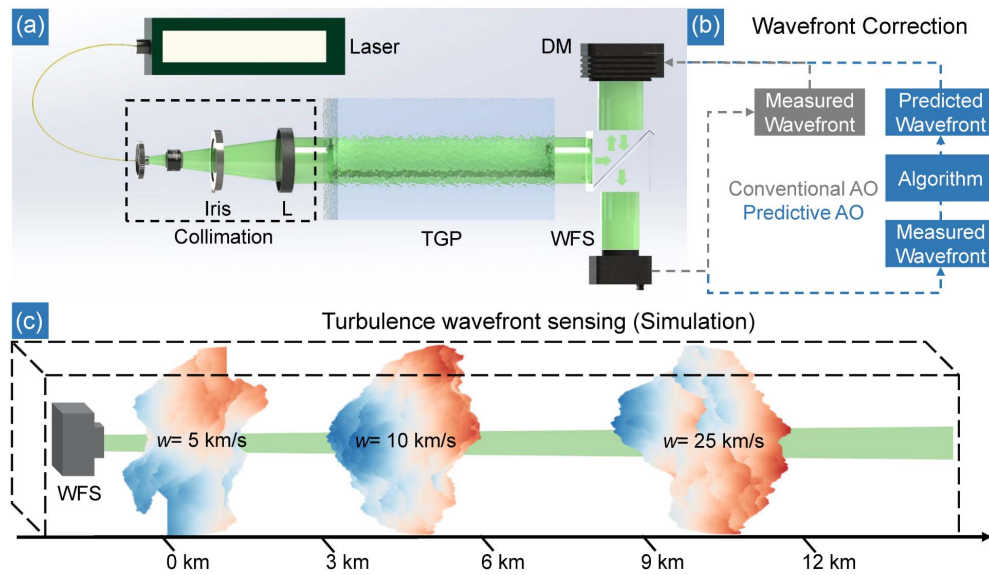


Fig. 3. (a) Optical setup model of the AO system. TGP, turbulence generating pool; WFS, wavefront sensor; DM, deformable mirror. (b) Data flow of conventional AO (gray lines) and predictive AO (blue lines). (c) Simulated atmospheric turbulence measuring process.

$$\mathbf{F}_{\text{out}} = \sum_i^k \mathbf{F}^{(i)} \mathbf{W}^{(i)}, \quad (11)$$

where the weight $\mathbf{W}^{(i)}$ is trainable for filtering and transforming. The other branch is similar, except for replacing \mathbf{M} with its transpose \mathbf{M}^T . The output features of two branches are added for the next TFC. The spatial wavefront is represented by Zernike coefficients, so the optimization of spatial features can be achieved by adjusting these coefficients. The modules including the TFC and SFA are also implemented five times in succession. All features from the start layer, TFCs and SFAs are adjusted by 1×1 convolutional operation and added together for long residual connections.

Specifically, the architecture design of the MGNN is unique and innovative, leaving aside the details of data flow. From previous related works, a satisfactory prediction algorithm requires non-linear fitting ability and spatiotemporal feature extraction ability for accurate prediction, and sufficient parameter capacity and information modeling ability for robust generalization. However, existing algorithms, including state-of-the-art deep learning-based ones, focus more on accurate prediction with similar data, leading to poor generalization performance in various environments. Limited data for training cannot cover all variations of turbulence, so it is implausible to predict unfamiliar turbulence only with data-driven learning. Indeed, the prediction accuracy within limited data can be improved by special structures with enough training, but the generalization capability needs the introduction of physical rules for further enhancement, which was never considered before. Although lots of networks utilize spatiotemporal information in prediction, they essentially assume that all elements are independent, which is different from the distribution of real atmosphere. In the MGNN, Zernike polynomials are introduced to reduce data dimensions and export covariance matrix. The concept of graph is introduced due to its high consistency with the covariance

matrix. The derived adjacency matrix can emphasize the relevant coefficients in iterative optimization, while others are not affected because this matrix is highly sparse. Further, the weight $\mathbf{W}^{(i)}$ also plays an important role in transforming these coefficients, just like the transformation matrix \mathbf{V} in Eq. (6). They add physical constraints to spatial features, making output closer to the real turbulence statistically. For temporal features, one catcher has fewer parameters because of the dimension reduction of wavefront with the Zernike model. As a result, our proposed MGNN is applicable to the accurate prediction of various atmospheric environments, because it builds a general mapping relationship within limited data.

D. Data Preparation

Figure 3(a) shows the optical setup model of the AO system, and Fig. 3(b) represents differences between conventional and predictive AO systems in a closed-loop wavefront correction. In an AO system, the collimated laser goes through the atmospheric turbulence, so its wavefront is disturbed. The distorted wavefront is measured and demodulated by the WFS, and the corresponding control voltages of the DM, Zernike coefficients, and phase distribution of wavefront are obtained. The conventional AO directly sends the control voltages to the DM for correction with an inevitable time delay. The predictive AO utilizes these above measured signals for prediction with algorithms, and the predicted signals, representing the future wavefront, are sent to the DM for correction. The subsequent data acquisition and analysis both follow these two figures, including the simulation and experiment.

To get the training dataset in Fig. 1(a), the object-oriented MATLAB adaptive optics model is developed to effectively simulate the AO system under different conditions [28]. In simulation, a simple wavefront measurement system in Fig. 3(c) is created, which consists of a point source at infinity, an atmospheric medium, and a WFS. The light from a point source at infinity goes through the atmospheric medium and is measured

by a WFS. The WFS has a 16×16 microlens array, a resolution of 128×128 pixels, and a measurement frame rate of 600 Hz. Three large random phase screens implementing the von Kármán statistics are generated within the atmosphere medium. These phase screens are distributed at transmission altitudes of $[0, 4, 10]$ km with Taylor's hypothesis. The Fried parameter r_0 of the phase screens follows the normal distributions as $N_r(15, 2)$ cm, and the windspeed w is $[5 \pm 2.5, 10 \pm 5, 20 \pm 10]$ m/s for the training dataset. The wind directions are sampled randomly from $(0, 2\pi)$ rad. These phase screens shift randomly with the given windspeed. In each frame measurement, partial phase screens are integrated along the light propagation path and ultimately measured by the WFS. At last, we get the Zernike coefficients of 8000 frames wavefront for training. According to the prediction model, the first 10 frames are used as inputs, and the 13th frame is used as the label. The output of the MGNN is the Zernike coefficient of one-frame wavefront phase, so its orders can be adjusted according to actual needs. In the distortion of turbulence, low-order aberrations dominate and can be effectively represented by the Zernike model. Therefore, coefficients of 2–15 orders Zernike polynomial are used to represent the wavefront here for verification, and more orders are also feasible with similar performance. Coefficients of all test sets are normalized by the same rule as in the training set in advance and are restored after prediction.

To verify the prediction accuracy of the algorithms, we generate additional coefficients of 2000 frames under the same condition as test set 1. To study the robustness to varying atmospheric environments, we also generate another four test sets with 2000 frames. For example, the windspeed is changed as $[10 \pm 2.5, 15 \pm 5, 25 \pm 10]$ m/s in test set 2, and the Fried parameter is changed as $N_r(5, 2)$ cm, $N_r(10, 2)$ cm, and $N_r(20, 2)$ cm, in test sets 3, 4, and 5, respectively.

Furthermore, the experimental data is also considered to verify the prediction performance of algorithms trained by simulation. The experimental optical setup is basically consistent with Fig. 3(a). A 1.5 m long turbulence generating pool (TGP) is heated at the bottom and cooled at the top and generates Rayleigh–Bénard convection randomly. The experimental turbulence is quite different from the simulated ones, which is based on Taylor's hypothesis and von Kármán statistical model, so it can demonstrate the robustness of the algorithms effectively. The 532 nm laser is collimated, goes through the TGP, and is then captured by a WFS. The WFS has a 12×12 microlens array, a resolution of 128×128 pixels, and a measurement frame rate of 420 Hz. Finally, we get the coefficients of 800 frames wavefront with the Fried parameter 15 ± 0.2 cm and random windspeed as test set 6.

For comparison, we train the MGNN, LMMSE, and LSTM with the same train set, and test them using the above six test sets. All algorithms are implemented by PyTorch 1.7 and Python 3.7.1, and the computer has an Intel Core i7-10700K CPU, 32 GB of RAM, and an NVIDIA GeForce GTX 1080Ti GPU. The L1 loss function for the MGNN and LSTM, and L2 loss function for the LMMSE are used with the Adam optimizer, and their dropout rates are 0.2. For the MGNN, the learning rate is 1×10^{-4} , and the epoch

is set as 100. For the LMMSE and LSTM, the learning rate and epoch are 1×10^{-3} and 200, respectively. The inputs of all algorithms are the Zernike coefficients of the previous 10 consecutive frames, and the output is the coefficients of the future third frame. The training of the MGNN takes ~ 14 min for 100 epochs, and the forward reasoning in testing takes ~ 1.721 s for 2000 frames. On average, the time consumption of one single frame is approximately 0.862 ms, which is less than the sampling time of one frame by the WFS. In contrast, the trained LMMSE and LSTM take 0.385 ms and 0.183 ms for one-frame prediction, respectively. The parameter quantity and floating-point operations per second (FLOPS) of the LMMSE, LSTM, and MGNN are 9153 and 3.08 kFLOPS, 36221 and 1.48 kFLOPS, 337825 and 38.5 MFLOPS, respectively. It is obvious that the MGNN is more complex and needs more computing resources, due to its repetitive TFC and SFA structures. The increase in parameter quantity also corresponds to the strong robustness of the MGNN, which is the key point we will focus on later.

3. ROBUSTNESS VERIFICATION AND DISCUSSION

After adequate optimization, our proposed MGNN was compared with the LMMSE and LSTM on the above six test sets. In addition, the conventional AO was also performed with time delay in wavefront correction for comparison. The prediction accuracy is quantified by the root mean square error (RMS) of wavefronts, which can be expressed as

$$\text{RMS} = \sqrt{\frac{1}{MN} \sum_i^M \sum_j^N [P(\mathbf{A}_p) - P(\mathbf{A}_f)]^2}, \quad (12)$$

where M and N are the length and width of the wavefront phase, $P(\cdot)$ is the function to convert coefficients into wavefront phase, and \mathbf{A}_p and \mathbf{A}_f are the predicted and future coefficient vectors as explained before.

A. Robustness to Turbulence under the Same Conditions

Test set 1 generated under the same condition for turbulence simulation is taken for verification of prediction accuracy. Figure 4 shows the predicted Zernike coefficients and corresponding wavefronts of future third frames in test set 1 as two samples. From these samples in Figs. 4(a) and 4(b), the future wavefronts to be predicted have a large phase range, while the absolute errors of the predicted ones by the three algorithms are much smaller. In comparison, our MGNN has the lowest error in Zernike coefficients and reconstructed wavefronts visually, compared with the LMMSE and LSTM. To better evaluate the overall performance of the algorithms, we calculate the RMS values in Eq. (12) of all 1997 data pairs and count their histograms and RMS curves of consecutive frames in Fig. 5. In Fig. 5(a1), the normal curves are calculated with the mean μ and standard deviation (SD) σ for intuitive comparison, and S is a magnification factor. On the normal curves, being the closer to the left axis means the smaller overall error, and the narrower peak width means the better stability. The mean μ and standard deviation σ are also listed in Table 1.

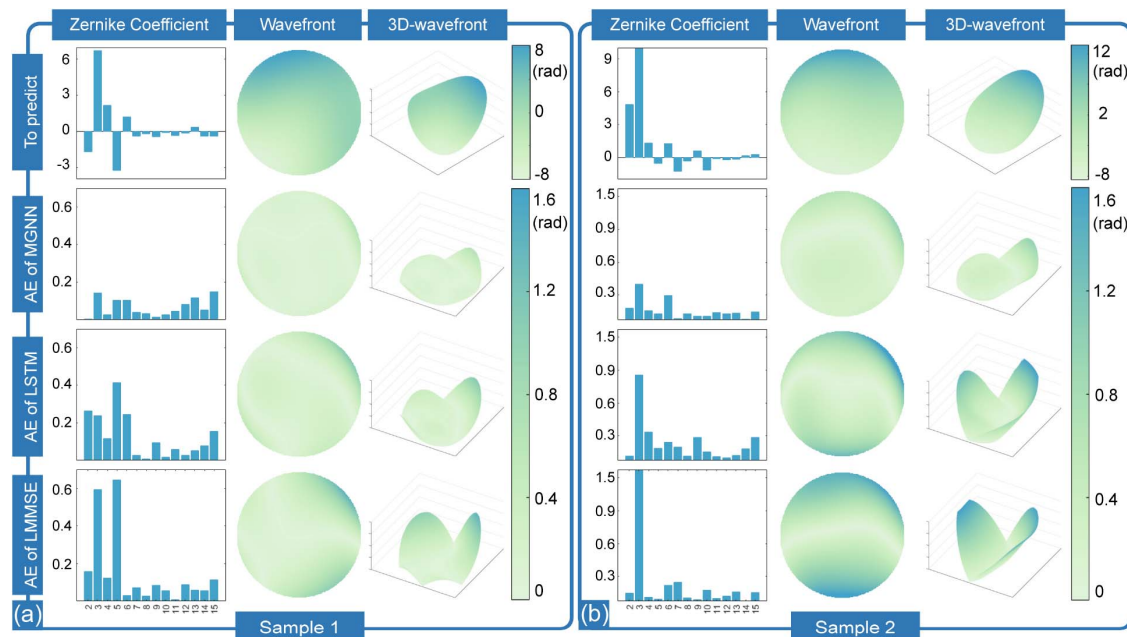


Fig. 4. Comparison of one-frame prediction accuracy by three algorithms. (a), (b) Predicted Zernike coefficients and corresponding wavefronts of future third frame in test set 1 as two samples. MGNN, our proposed method; LSTM, one non-linear algorithm; LMMSE, one linear algorithm; AE, absolute error.

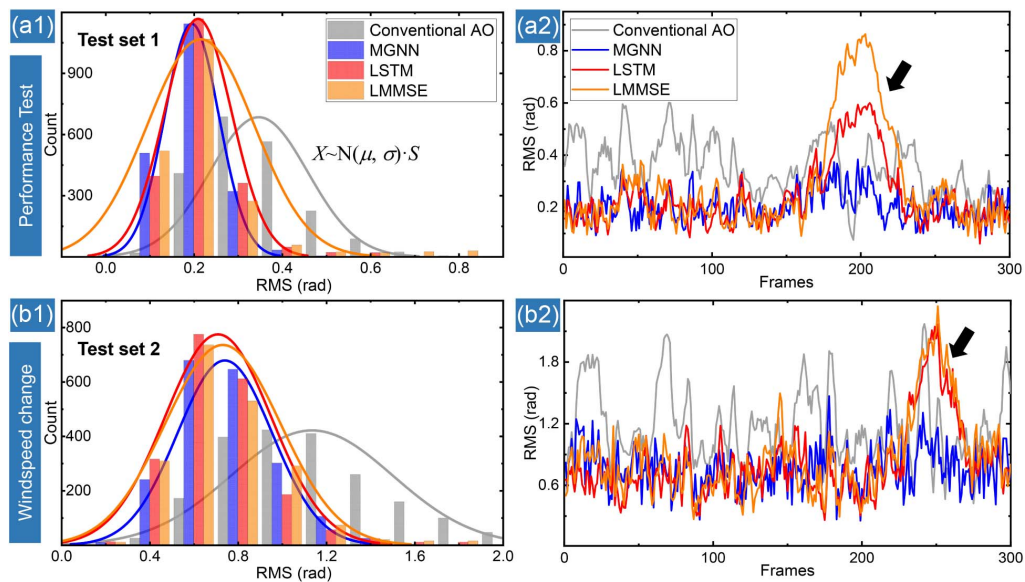


Fig. 5. Comparison of overall prediction performance in different test sets. (a1) Histograms and normal curves of RMS values in test set 1. $N(\mu, \sigma)$ is the normal distribution with mean μ and standard deviation σ , and S is a magnification factor. (a2) RMS curves of consecutive frames with or without prediction in (a1). (b1) Histograms and normal curves of RMS values in test set 2. (b2) RMS curves of consecutive frames with or without prediction in (b1). MGNN: blue; LSTM: red; LMMSE: orange; conventional AO: gray. Test set 1: the same condition as training; test set 2: windspeed changes. Black arrow: emphasis in comparison.

From Fig. 5(a1) and Table 1, our MGNN has the lowest values compared with the other two algorithms, which indicates that the MGNN achieves the most accurate prediction and stable prediction here. From the RMS curves of consecutive frames in Fig. 5(a2), the blue curve (MGNN) is the most stable one when a prediction variation occurs. For comparison, the red and orange curves (LSTM and LMMSE) are higher than

the gray curve (conventional AO). It indicates that the incorrect predictions result in worse correction performance, which is not expected in predictive AO. The prediction variation in Fig. 5(a2) occurs due to the significant change of phase. This evolution mode may not exist in the training set, so all algorithms have correction performance degradation at the arrow. Thanks to the introduced physical constraints, our

Table 1. Statistical RMS Values of All Test Sets in Simulation and Experiment^a

Test Sets	Parameters	RMS (rad)	Conventional AO	Predictive AO			
				MGNN	LSTM	LMMSE	
Sim.	Set 1	$w = [5, 10, 20]$ m/s	Mean (μ)	0.346	0.194	0.209	0.215
		$r_0 = N_r(15, 2)$ cm	SD (σ)	0.111	0.061	0.076	0.121
	Set 2	$w = [10, 15, 25]$ m/s	Mean (μ)	1.134	0.739	0.709	0.730
			SD (σ)	0.366	0.215	0.241	0.262
	Set 3	$r_0 = N_r(5, 2)$ cm	Mean (μ)	0.283	0.136	0.339	0.281
			SD (σ)	0.078	0.047	0.142	0.200
	Set 4	$r_0 = N_r(10, 2)$ cm	Mean (μ)	0.370	0.197	0.335	0.272
			SD (σ)	0.104	0.050	0.103	0.094
	Set 5	$r_0 = N_r(20, 2)$ cm	Mean (μ)	0.166	0.076	0.149	0.162
			SD (σ)	0.058	0.024	0.115	0.062
Exp.	Set 6	$r_0 = 15 \pm 0.2$ cm	Mean (μ)	0.033	0.025	0.092	0.064
			SD (σ)	0.012	0.007	0.029	0.025

^a $N_r(v_1, v_2)$: random number of normal distributions with mean v_1 and standard deviation v_2 . Sim: simulation; Exp: experiment.

MGNN has a significant advantage of stability in predicting turbulence with temporal variation.

B. Robustness to Turbulence with Changing Windspeed

In the natural atmosphere, the windspeed changes easily without a specific range, so we must consider the robustness of algorithms to the unknown turbulence with changing windspeed. Test set 2 is generated through three phase screens with a higher windspeed, which represents faster changes in turbulence. On the basis of calculation way above, their histograms and RMS curves of consecutive frames are shown in Figs. 5(b1) and 5(b2). Compared with Fig. 5(a1), the RMS values of conventional AO significantly increased, due to the rapidly changing turbulence. Within the time delay of three frames, the turbulence undergoes faster and greater changes, resulting in

greater correction errors. In comparison, the predictive AO is effective in reducing the correction errors, and our MGNN is still the most stable one from Figs. 5(b1) and 5(b2). All prediction algorithms have similar mean values of RMS, which are higher than those in Fig. 5(a1). The changes of windspeed introduce faster and more complex changes in turbulence, resulting in a significant impact on prediction. The new inherent changing model does not match the temporal pattern obtained by training, making prediction more difficult for the algorithms. Overall, the effect of predictive AO is still held, and the stability advantage of the MGNN exists here.

C. Robustness to Turbulence with Changing Fried Parameters

The Fried parameter represents the coherence length of atmospheric turbulence, which is one of the key parameters to

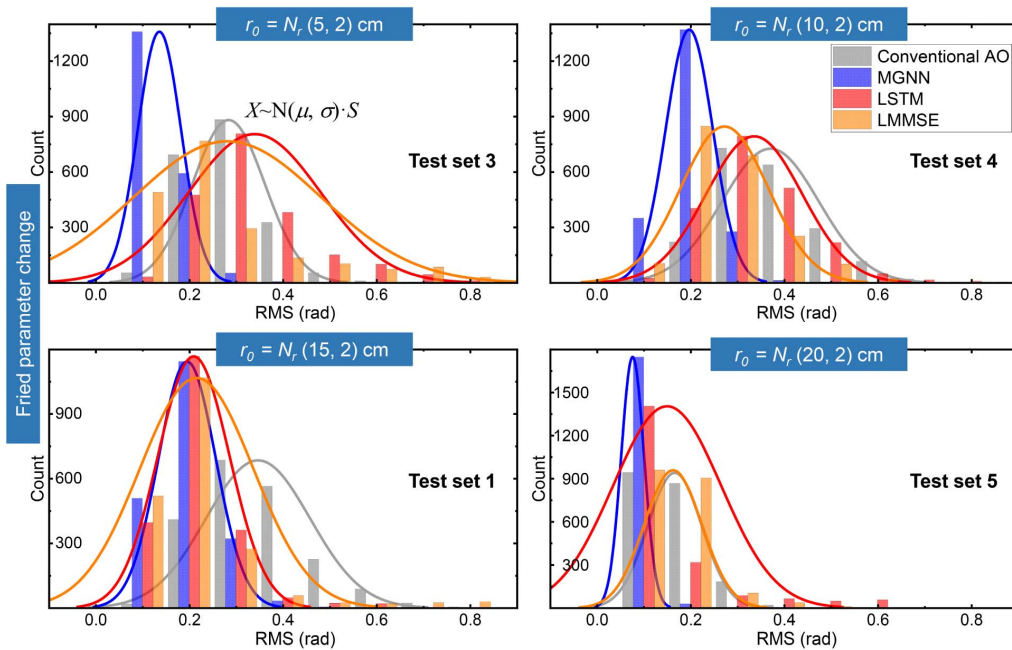


Fig. 6. Comparison of overall prediction performance using test sets with changing Fried parameters. Histograms and normal curves of RMS values are counted in test sets 3, 4, 1, 5, respectively. MGNN: blue; LSTM: red; LMMSE: orange; conventional AO: gray. Test sets 3, 4, and 5: Fried parameters change.

quantify the atmosphere. To verify the robustness to unknown turbulence with changing Fried parameters, we take the test sets 1, 3, 4, and 5 for comparison. Here the Fried parameter is generated by the random number of normal distributions $N_r(v_1, v_2)$ with mean v_1 and standard deviation v_2 , while smaller v_1 represents stronger turbulence. Figure 6 shows the histograms and corresponding normal curves calculated from RMS values. It is obvious that the blue normal curves are left-most and have the narrowest peak width in test sets 3, 4, and 5, which means the MGNN has the lowest prediction errors and the most stable performance. In contrast, the performance of the LSTM and LMMSE is unsatisfactory, because the RMS curves of them are further right and wider than that of conventional AO. As shown in Table 1, the MGNN has almost all the lowest values in mean and SD values from the RMS, compared with the other two universal algorithms. With the help of the MGNN, the mean and SD values of the RMS for conventional AO are reduced by 54.2% and 58.6% at most, respectively. The robustness of our MGNN is highlighted here thanks to the specially designed network, which is helpful to learn more general spatiotemporal patterns from limited simulation data.

D. Robustness to Turbulence in Experiment

After verifying the robustness on simulated data, the MGNN is also expected to have stable performance on experimental data. Under the constriction of supervised learning, the hardware and time cost are required to obtain enough data from the experiment for deployment. In addition, the turbulence is uncontrollable in the experiment, resulting in the uncertain richness in the training set. Therefore, the capability of generalization from simulated data to experimental data is highly important for reducing costs and difficulty. In our experiment, the actual parameters are different from those in simulation, which is persuasive for the robustness of the MGNN. Figures 7(a1) and 7(a2) show the histograms, the corresponding normal curves, and the RMS curves of consecutive frames like above. Similarly, the performance of the LSTM and LMMSE is unacceptable, due to their negative optimization to the correction results of conventional AO. From Fig. 7(a2), when turbulence changes slowly and the correction error of conventional AO is relatively low, the prediction algorithm should maintain this state in predictive AO. While turbulence undergoes a sudden change, the prediction algorithm needs to reduce the correction

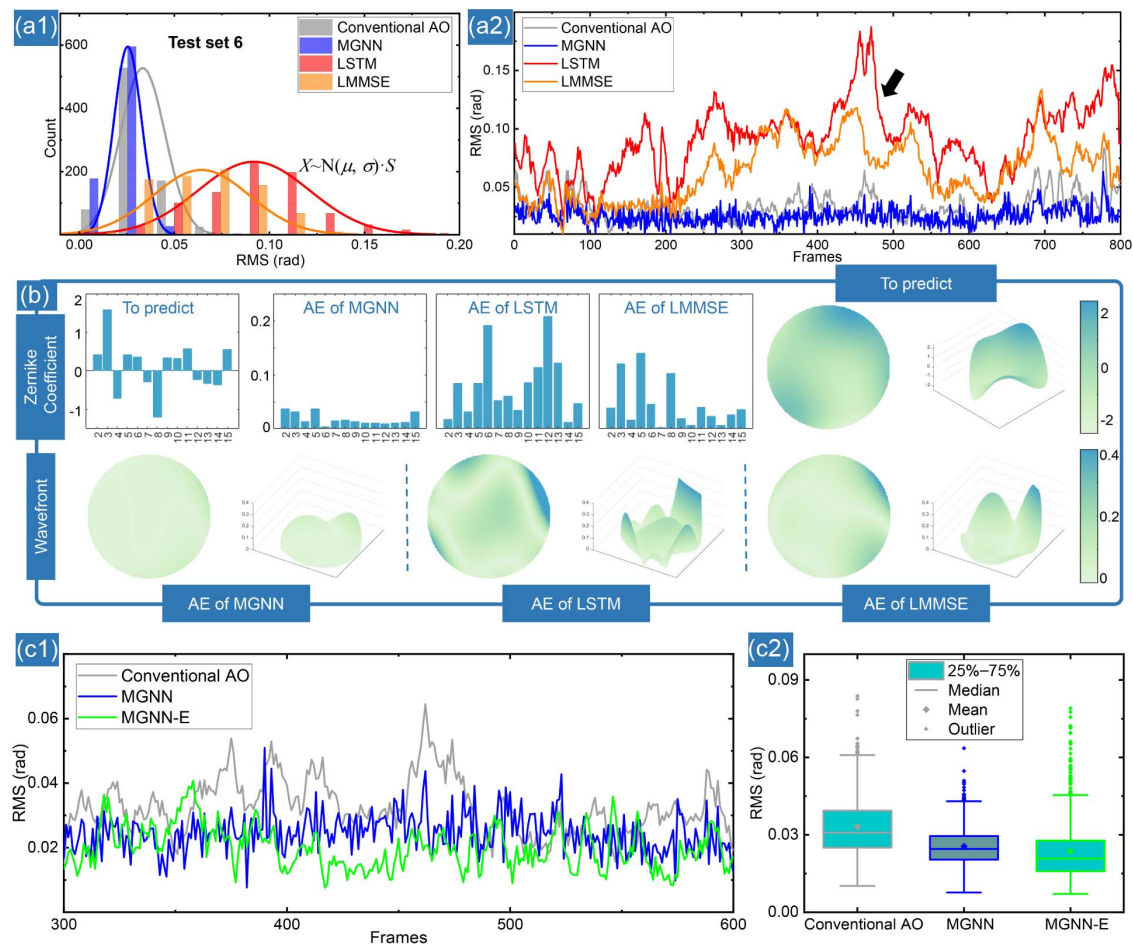


Fig. 7. Robustness to the experimental data. (a1) Histograms and normal curves of RMS values in test set 6. (a2) RMS curves of consecutive frames with or without prediction in (a1). MGNN: blue; LSTM: red; LMMSE: orange; conventional AO: gray. Black arrow: emphasis in comparison. (b) Predicted Zernike coefficients and corresponding wavefronts of future third frame in test set 6. AE, absolute error. (c1) RMS curves of consecutive frames with prediction by the MGNN and MGNN-E. MGNN-E (green): MGNN trained by the experimental data. (c2) Box plots of RMS in (c1).

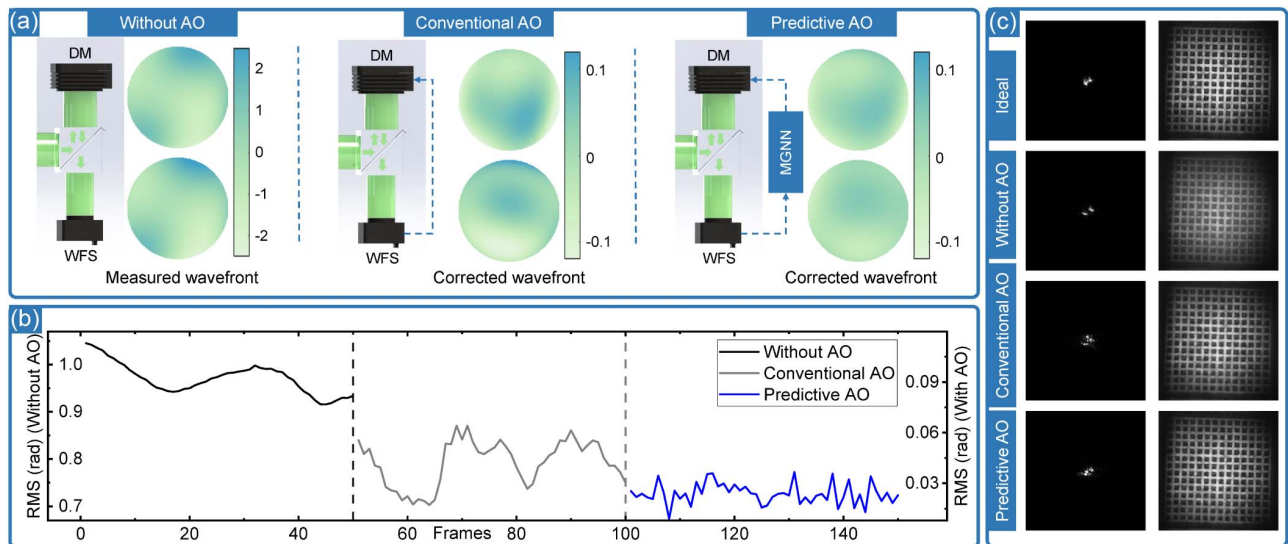


Fig. 8. Closed-loop correction wavefront error in experiment. (a) Three stages of the AO system and their corresponding corrected wavefronts. (b) RMS curves of three stages in closed-loop correction. Without AO: black line; conventional AO: gray line; predictive AO with MGNN: blue line. (c) Focal spot and grid imaging of three stages.

error for the stable state. Obviously, our MGNN achieves this goal while the other two algorithms do not. In fact, the LMMSE and LSTM are also trained with the simulated data and have fewer parameter quantities than the MGNN. They recognize the turbulence changes in the training set but are unable to establish a general mode of turbulence evolution. Therefore, they still make predictions based on familiar but incorrect relationships, resulting in huge errors in the experimental data. As shown in Fig. 7(b), the Zernike coefficients and the absolute error of the MGNN are the best. From Table 1, the mean and SD values of the RMS in predictive AO with the MGNN are 75.8% and 58.3% of those in conventional AO for frozen turbulence. To further demonstrate its robustness, we collected 8000 frames as experimental data from the TGP, and retrained the proposed network MGNN-E. In Fig. 7(c1), the RMS curves of the MGNN and MGNN-E from the 300th to 600th frames in test set 6 are shown for comparison, and the complete statistical results are displayed in the box plots in Fig. 7(c2). From these two figures, the correction performance with the MGNN is similar to that with the MGNN-E. The MGNN establishes the internal laws of turbulence evolution from the simulated data, so it has sufficient robustness for the generalization from simulation to experiment.

Furthermore, we take the closed-loop wavefront correction numerically by the experimental data, as shown in Fig. 8. The correction process is divided into three stages: without AO, with conventional AO, and with predictive AO. The corresponding wavefronts measured by the WFS are shown in Fig. 8(a), and the RMS curves of consecutive frames for three stages are shown in Fig. 8(b). Intuitively, our MGNN still has good robustness to the experimental data, although it is optimized by simulated data and has never seen these real turbulence data before. Both simulated and experimental data follow the statistical invariant covariance matrix. Therefore, the physical constraints with the Zernike model enable the MGNN to reveal and establish the generic spatiotemporal evolution patterns of turbulence

during optimization. To verify the effect of prediction on wavefront correction, we load the wavefront phases of three stages onto a spatial light modulator (SLM). The grid is imaged, and the focal spots are obtained with the phase modulation of the SLM, as shown in Fig. 8(c). To visualize the imaging changes, the phases are appropriately amplified in the same way. With predictive AO, the focal spot is the most concentrated and the grid is the clearest. They are similar to these of ideal cases without wavefront disturbance. Overall, the MGNN has good robustness to the experimental data. With the help of the MGNN, the wavefront correction error can be further reduced, which improves the light focusing and imaging effect.

4. CONCLUSIONS

In this paper, we have demonstrated a mixed graph neural network for spatiotemporal wavefront prediction. The temporal feature catcher and spatial feature analyzer are both specially designed to deal with temporal and spatial features, respectively. The Zernike model plays an important role in reducing data dimensions and exporting covariance matrix. Our MGNN takes the invariant covariance matrix of Zernike coefficients as physical constraints and introduces the graph neural network for spatial feature analysis. Compared with the LMMSE and LSTM, our MGNN has the best stability and predictive correction in different test sets, including simulated turbulences under the same condition, with the changing windspeed, with the changing Fried parameters, and the real turbulence in experiment. It can perform well with stable and reasonable prediction in practical AO systems and achieve similar performance to that trained in real turbulence, even if it is only trained with simulated data. Our proposed MGNN innovatively introduces the statistical rules of physical turbulence into network design and exhibits astonishing generalization robustness. Its stable prediction ability is applicable to wavefront correction of AO in astronomical observation, laser communication, and microscopic imaging.

APPENDIX A

According to Noll's representation, the covariance value between any two coefficients a_w and a_v in \mathbf{A} of one frame can be expressed as

$$E(a_w, a_v) = \frac{\delta K_{wv}(Z_w, Z_v) \Gamma[(n_w + n_v - 5/3)/2] (D/r_0)^{5/3}}{\Gamma[(n_w - n_v + 17/3)/2] \Gamma[(n_v - n_w + 17/3)/2] \Gamma[(n_w + n_v + 23/3)/2]}, \quad (\text{A1})$$

where n_w and n_v are the radial orders of Z_w and Z_v , D is the optical diameter of the system, r_0 is the Fried parameter of atmosphere, and $\Gamma[\cdot]$ is the gamma function. K_{wv} is the frequency characteristic factor depending on the polynomial terms Z_w and Z_v , and δ is the covariance parameter with a value of 0 or 1. They can be expressed as

$$K_{wv}(Z_w, Z_v) = 2.2698(-1)^{n_w+n_v-2m_w} \sqrt{(n_w+1)(n_v+1)}, \quad (\text{A2})$$

$$\delta = \begin{cases} 1, & (m_w \neq 0) \cap (m_w = m_v) \cap \text{Par}(w, v), \\ 0, & \text{others} \end{cases}, \quad (\text{A3})$$

where m_w and m_v are the angular orders of Z_w and Z_v , and $\text{Par}(w, v)$ is the function to judge whether the parity of w and v is the same. We take Eqs. (A2) and (A3) into Eq. (A1), and the covariance value $E(a_w, a_v)$ is equal to the following values besides zero:

$$E(a_w, a_v) = \begin{cases} \frac{2.2698(n_w+1)\Gamma[n_w-5/6](D/r_0)^{5/3}}{\Gamma[17/6]^2\Gamma[n_w+23/6]}, & w = v \\ \frac{-2.2698\sqrt{(n_w+1)(n_v+1)}\Gamma[(n_w+n_v-5/3)/2](D/r_0)^{5/3}}{\Gamma[(n_w-n_v+17/3)/2]\Gamma[(n_v-n_w+17/3)/2]\Gamma[(n_w+n_v+23/3)/2]}, & w \neq v \end{cases}. \quad (\text{A4})$$

According to Eq. (A4), ignoring the piston term Z_1 , the covariance matrix \mathbf{C} of Zernike coefficients can be expressed as

$$\mathbf{C} = E(\mathbf{A}^T \cdot \mathbf{A}) = \begin{bmatrix} E(a_2, a_2) & 0 & \dots & 0 \\ 0 & E(a_3, a_3) & \dots & E(a_w, a_v) \\ \vdots & \vdots & \ddots & \vdots \\ 0 & E(a_v, a_w) & \dots & E(a_m, a_m) \end{bmatrix}. \quad (\text{A5})$$

Specifically, by ignoring the scaling factor $(D/r_0)^{5/3}$, the covariance matrix $\tilde{\mathbf{C}}$ is a constant matrix as follows:

$$\tilde{\mathbf{C}} = \begin{bmatrix} 0.4557 & 0 & 0 & 0 & 0 & 0 & -0.0144 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.4557 & 0 & 0 & 0 & -0.0144 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.0236 & 0 & 0 & 0 & 0 & 0 & 0 & -0.0039 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.0236 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -0.0039 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.0236 & 0 & 0 & 0 & 0 & 0 & -0.0039 & 0 & 0 & 0 \\ 0 & -0.0144 & 0 & 0 & 0 & 0.0063 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.0144 & 0 & 0 & 0 & 0 & 0 & 0.0063 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.0063 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.0063 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -0.0039 & 0 & 0 & 0 & 0 & 0 & 0 & 0.0025 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -0.0039 & 0 & 0 & 0 & 0 & 0 & 0.0025 & 0 & 0 & 0 \\ 0 & 0 & 0 & -0.0039 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.0025 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.0025 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.0025 \end{bmatrix}. \quad (\text{A6})$$

Funding. National Natural Science Foundation of China (61905197, 62075183, 62275218); China Postdoctoral Science Foundation (2022M712586); Guangdong Introducing Innovative and Entrepreneurial Teams of "The Pearl River Talent Recruitment Program" (2021ZT09X04); Basic and Applied Basic Research Foundation of Guangdong Province (2023A1515011335); Fundamental Research Funds for the Central Universities (D5000230117).

Disclosures. The authors declare no conflicts of interest.

Data Availability. Data underlying the results presented in this paper are not publicly available at this time but may be obtained from the authors upon reasonable request.

REFERENCES

1. D. Kim, H. Choi, T. Brendel, H. Quach, M. Esparza, H. Kang, Y. Feng, J. N. Ashcraft, X. Ke, T. Wang, and E. S. Douglas, "Advances in optical engineering for future telescopes," *Opto-Electron. Adv.* **4**, 210040 (2021).
2. H. Chang, X. Yin, H. Yao, J. Wang, R. Gao, X. Xin, and M. Guizani, "Adaptive optics compensation for orbital angular momentum optical

- wireless communications," *IEEE Trans. Wireless Commun.* **21**, 11151–11163 (2022).
3. C. Rodríguez, A. Chen, J. A. Rivera, M. A. Mohr, Y. Liang, R. G. Natan, W. Sun, D. E. Milkie, T. G. Bifano, X. Chen, and N. Ji, "An adaptive optics module for deep tissue multiphoton imaging in vivo," *Nat. Methods* **18**, 1259–1264 (2021).
 4. Y. Guo, L. Zhong, L. Min, J. Wang, Y. Wu, K. Chen, K. Wei, and C. Rao, "Adaptive optics based on machine learning: a review," *Opto-Electron. Adv.* **5**, 200082 (2022).
 5. M. A. M. van Kooten, N. Doelman, and M. Kenworthy, "Robustness of prediction for extreme adaptive optics systems under various observing conditions: an analysis using VLT/sphere adaptive optics data," *Astron. Astrophys.* **636**, A81 (2020).
 6. M. Salama, J. Ou, C. Baranec, M. C. Liu, B. P. Bowler, P. Barnes, M. Bonnet, M. Chun, D. A. Duev, and S. Goebel, "Large adaptive optics survey for substellar objects around young, nearby, low-mass stars with robo-AO," *Astron. J.* **162**, 102 (2021).
 7. I. Toselli and S. Gladysz, "Improving system performance by using adaptive optics and aperture averaging for laser communications in oceanic turbulence," *Opt. Express* **28**, 17347–17361 (2020).
 8. M. B. Jorgenson and G. J. M. Aitken, "Prediction of atmospherically induced wave-front degradations," *Opt. Lett.* **17**, 466–468 (1992).
 9. M. Van Kooten, N. Doelman, and M. Kenworthy, "Impact of time-variant turbulence behavior on prediction for adaptive optics systems," *J. Opt. Soc. Am. A* **36**, 731–740 (2019).
 10. K. Jackson, C. Correia, O. Lardière, D. Andersen, and C. Bradley, "Linear prediction of atmospheric wave-fronts for tomographic adaptive optics systems: modelling and robustness assessment," *Opt. Lett.* **40**, 143–146 (2015).
 11. C. Liu, L. Hu, Z. Cao, Q. Mu, and L. Xuan, "Modal prediction of atmospheric turbulence wavefront for open-loop liquid-crystal adaptive optics system with recursive least-squares algorithm," *Opt. Commun.* **285**, 238–244 (2012).
 12. L. A. Poyneer, B. A. Macintosh, and J. P. Véran, "Fourier transform wavefront control with adaptive prediction of the atmosphere," *J. Opt. Soc. Am. A* **24**, 2645–2660 (2007).
 13. X. Liu, T. Morris, C. Saunter, F. J. de Cos Juez, C. González-Gutiérrez, and L. Bardou, "Wavefront prediction using artificial neural networks for open-loop adaptive optics," *Mon. Not. R. Astron. Soc.* **496**, 456–464 (2020).
 14. Y. Chen, "LSTM recurrent neural network prediction algorithm based on Zernike modal coefficients," *Optik* **203**, 163796 (2020).
 15. Y. Chen, "Voltages prediction algorithm based on LSTM recurrent neural network," *Optik* **220**, 164869 (2020).
 16. S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.* **9**, 1735–1780 (1997).
 17. J. G. Chen, V. Shah, and L. Liu, "Performance of a U-Net-based neural network for predictive adaptive optics," *Opt. Lett.* **46**, 2513–2516 (2021).
 18. X. Shi, Z. Chen, H. Wang, D. Y. Yeung, W. K. Wong, and W. C. Woo, "Convolutional LSTM network: a machine learning approach for precipitation nowcasting," in *Proceedings of the 28th International Conference on Neural Information Processing Systems (NIPS)* (2015), pp. 802–810.
 19. R. Swanson, M. Lamb, C. M. Correia, S. Sivanandam, and K. Kutulakos, "Closed loop predictive control of adaptive optics systems with convolutional neural networks," *Mon. Not. R. Astron. Soc.* **503**, 2944–2954 (2021).
 20. J. Wu, J. Tang, M. Zhang, J. Di, L. Hu, X. Wu, G. Liu, and J. Zhao, "PredictionNet: a long short-term memory-based attention network for atmospheric turbulence prediction in adaptive optics," *Appl. Opt.* **61**, 3687–3694 (2022).
 21. K. Wang, M. Zhang, J. Tang, L. Wang, L. Hu, X. Wu, W. Li, J. Di, G. Liu, and J. Zhao, "Deep learning wavefront sensing and aberration correction in atmospheric turbulence," *Photonix* **2**, 8 (2021).
 22. J. Tang, J. Zhang, S. Zhang, S. Mao, Z. Ren, J. Di, and J. Zhao, "Phase aberration compensation via a self-supervised sparse constraint network in digital holographic microscopy," *Opt. Laser Eng.* **168**, 107671 (2023).
 23. C. Lee, G. Song, H. Kim, J. C. Ye, and M. Jang, "Deep learning based on parameterized physical forward model for adaptive holographic imaging with unpaired data," *Nat. Mach. Intell.* **5**, 35–45 (2023).
 24. R. J. Noll, "Zernike polynomials and atmospheric turbulence," *J. Opt. Soc. Am.* **66**, 207–211 (1976).
 25. G. Dai, "Modal wave-front reconstruction with Zernike polynomials and Karhunen–Loève functions," *J. Opt. Soc. Am. A* **13**, 1218–1225 (1996).
 26. Z. Wu, S. Pan, G. Long, J. Jiang, X. Chang, and C. Zhang, "Connecting the dots: multivariate time series forecasting with graph neural networks," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (2020), pp. 753–763.
 27. C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, E. Dumitru, V. Vincent, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2015), pp. 1–9.
 28. R. Conan and C. Correia, "Object-oriented Matlab adaptive optics toolbox," *Proc. SPIE* **9148**, 91486C (2014).