# Fiber-Optic Shape Sensing Using Neural Networks Operating on Multispecklegrams

Caroline G. L. Cao, *Member, IEEE*, Bernard Javot, *Member, IEEE*, Shreeram Bhattarai, Karin Bierig, Ivan Oreshnikov, and Valentin V. Volchkov

Abstract—**Application of machine learning techniques on fiber speckle images to infer fiber deformation allows the use of an unmodified multimode fiber to act as a shape sensor. This approach eliminates the need for complex fiber design or construction (e.g., Bragg gratings and time-of-flight). Prior work in shape determination using neural networks trained on a finite number of possible fiber shapes (formulated as a classification task), or trained on a few continuous degrees of freedom, has been limited to reconstruction of fiber shapes only one bend at a time. Furthermore, generalization to shapes that were not used in training is challenging. Our innovative approach improves generalization capabilities, using computer vision-assisted parameterization of the actual fiber shape to provide a ground truth, and multiple specklegrams per fiber shape obtained by controlling the input field. Results from experimenting with several neural network architectures, shape parameterization, number of inputs, and specklegram resolution show that fiber shapes with multiple bends can be accurately predicted. Our approach is able to generalize to new shapes that were not in the training set. This approach of end-to-end training on parameterized ground truth opens new avenues for fiber-optic sensor applications. We publish the datasets used for training and validation, as well as an out-of-distribution (OOD) test set, and encourage interested readers to access these datasets for their own model development.**

Index Terms—**Machine learning, neural networks, optical fibers, shape measurement, speckle patterns.**

multispecklegram → neural network model → free-form fiber shape
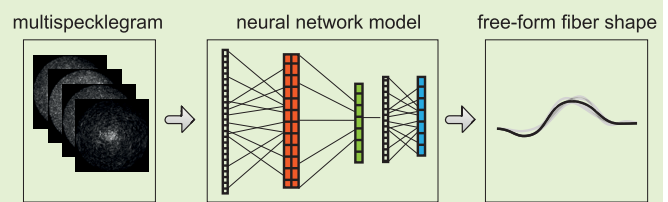
I. INTRODUCTION

SHAPE sensors based on optical fibers have been extensively developed, greatly improving upon the performance of traditional sensors (e.g., piezoelectric, electromagnetic, shape-memory alloys, and so on) [1], [2], [3]. Typically, fiber-based sensors benefit from low price, compactness, robustness, and flexibility of silica fibers. These characteristics allow fiber shape sensors (FSSs) to be deployed in critical environments, such as for tracking medical instruments in surgery [4] or for structural integrity monitoring in civil engineering [5], where no other solution is available. Over the years, the main avenues to improve the performance of FSSs were to increase the number of individual fiber cores or to modify the fiber itself (e.g., embedded quantum dots [4] and fiber Bragg grating [6], [7]) or to develop more elaborate interrogation techniques (e.g., time-of-flight [8]).

Alternatively, shape information could be imprinted on the phases of the spatial modes of the light propagating through a multimode fiber at the locations of deformation and read out in the form of an interference pattern at the end of the fiber. This principle was demonstrated using a triple-core optical fiber [9]. However, it is only recently with the advancement of machine learning techniques that it has become possible to process and analyze these complex interference patterns (known as specklegrams) (see reviews [10], [11], [12]). Specklegrams-based fiber sensors can detect external perturbations, such as deformation, temperature, or stress [13]. Deep neural networks have emerged as an ideal tool to correlate specklegrams with external perturbations without knowledge of the specific changes in physical properties.

For shape sensing, deep neural networks were trained to analyze specklegrams to extract curvature information at predefined locations [14] based on classification. Curvature sensing has been demonstrated for single [15] and multiple bends [16]. Classification-based approaches typically work

with a fixed number of discrete shapes and are not suitable for continuous deformations, as they are unable to classify an intermediate configuration. On the other hand, deflection sensing [17] and spatially resolved bend sensing [18] have been demonstrated using regression models, making it possible for continuous measurements and thereby increasing generalization capability of the models. For example, in [19], a fiber was deformed using three linear stages, generating a large dataset of shape configurations that was used to learn the relative changes of the stage position. While the absolute positions of the linear stages were obtained via integration of the relative position changes, the actual shape of the fiber remained undefined.

Here, we present a direct, 2-D shape reconstruction of a fiber using an end-to-end deep learning approach that addresses the above shortcomings. The shape of the fiber was continuously deformed, presenting multiple varying bends and captured using a camera. A flexible parameterization of the shape was developed to serve as ground truth for training neural network models. A large range of fiber shapes and their corresponding specklegrams were collected and used to investigate different neural network architectures, as well as to study the impact of different parameters, such as specklegram resolution, or the minimum number of segments used to describe the fiber shape. In particular, we introduced the concept of "multispeckle-grams" where we actively positioned the input beam at different positions on the proximal facet of the fiber end to obtain multiple specklegrams per fiber shape. Reconstruction of the shape from the specklegrams was achieved as a regression task with N-dimensional output, allowing for representation of nearly arbitrary shapes. The novelty of this approach is that it allows us to reconstruct the shape of the fiber with multiple bends at the same time, which represents an advancement over prior work that can handle the reconstruction of fiber shapes only one bend at a time. In this study, test sets that were not part of the original training set were employed to evaluate the generalization capabilities of the algorithms.

II. EXPERIMENTAL SETUP

Fig. 1 presents the optical setup for data collection for training and testing of the deep neural networks. The setup consists of a laser source (LaserQuantum, torus 532, not shown in Fig. 1), a fiber shape manipulation assembly, and cameras to capture the fiber shape and the speckle output of the fiber. As the laser source, we are using a beam at a wavelength of $\lambda = 532$ nm with a power of approximately $P = 1$ mW and collimated to a waist of $w_0 = 1$ mm.

The input position was controlled using a motorized galvo mirror system (Thorlabs, GVS002), which focused the laser beam via a microscope objective (Olympus Plan N 20×) onto the input facet of the fiber. The fiber selected for the experiment was a step-index MMF with a 200-μm-diameter core (Thorlabs, FG200UEA) and a length of 550 mm.

A custom-built motorized assembly was used to systematically introduce fiber bending simultaneously in multiple locations and with curvatures in the range of 0–17 m^{-1}. This assembly consisted of a flat rectangular surface of 425 × 210 mm to support the deformed region of the fiber, and two

sliders, each stacked with four cylindrical permanent magnets (Neodymium N45 zinc-plated rod magnet D 4 × 5 mm) that were flushed against the rectangular surface, separated by 100 mm, and with a displacement range of 80 mm orthogonal to the main direction of the fiber. The fiber was then passed through two metal rings (Alfa Aesar Stainless-Steel Type 304 tubing, 0.82-mm (0.032 in) OD and 0.51-mm (0.02 in) ID) before lying on an actual sheet of paper (124 g/m^2). As the sliders were moved underneath the paper surface, the ring followed the position of the stack of magnets. As illustrated in the zoomed-in view of Fig. 1, the rings were free to rotate and reorient according to the local direction of the fiber. As a result, the fiber curvature was freely adapting to the two position constraints while remaining flat on the surface. Each slider was attached to a rack and driven by a Maxon motor with embedded encoder, gearbox, and pinion. The distal end of the fiber was imaged using a microscope objective and a 100-mm camera lens (Navitar, NMV-100M23) on a CMOS camera (FLIR, CM3-U3-31S4M-CS), giving rise to a resolution of 0.31 μm/pixel. This imaging setup was mounted on a linear translation stage aligned with the main direction of the fiber. The linear translation stage also controlled the amount of fiber slack, allowing the fiber to assume different shapes. The stage position $d_{\text{Stage}} = 0$ mm corresponds to no fiber slack, such that the fiber shape is a straight line. In this study, the position of the linear translation stage was varied between 2.5 and 15 mm by a motorized linear actuator (Thorlabs, Z825B). Finally, a second camera was used to image the proximal end of the fiber in order to check the position of the focused laser beam. The ground-truth shape of the fiber was recorded using a third, external, camera (FLIR, CM3-U3-31S4M-CS, with E3Z4518CS-MPIR objective with a resolution of 189 μm/pixel) mounted above the plane of the fiber on the flat rectangular surface.

A. Data Acquisition and Preprocessing

Using the above described setup, a dataset was generated for training and shape reconstruction [20]. The linear translation stage was employed to set the initial fiber slack, while the sliders were manipulated in the increments of 2.7 mm to create fiber bends in various configurations. At each slider position, the shape of the fiber (i.e., ground truth) was first captured by the externally mounted camera. Then, the input laser focus was positioned on nine different locations on the fiber input facet using the galvo mirror, as shown in Fig. 1(b). At each of the nine positions, a specklegram was recorded. The resulting nested *for* loops are detailed in Algorithm 1. The raw speckle images at the distal end of the fiber were cropped to the dimensions of 656 × 656 pixels. Subsequently, a circular mask was applied to set all pixels outside of the core to 0.

B. Ground-Truth Generation

As shown in Fig. 2, an image of the fiber shape was first captured by a camera mounted above the plane of the fiber. The fiber was colored in black in order to increase the contrast of the image, making the postprocessing and ground-truth extraction more robust. The cropped image of the fiber was processed using different computer vision methods, such as
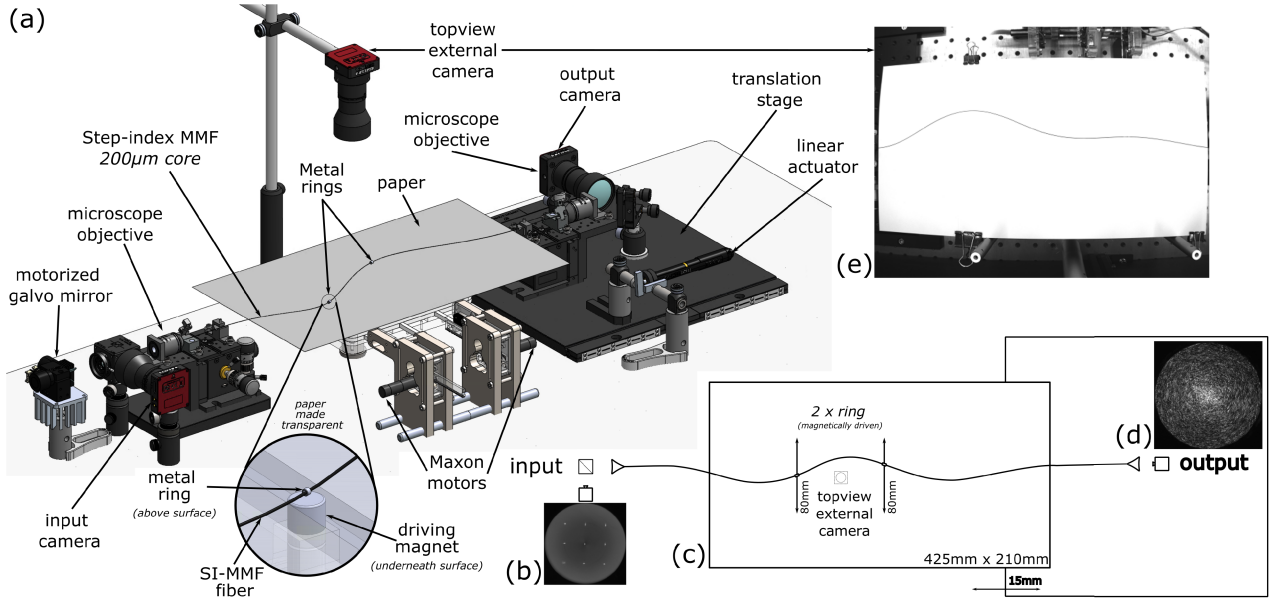
Fig. 1. (a) Illustration of the experimental setup. (b) Laser beam (source not shown) is controlled and focused by the motorized galvo mirror and microscope objective, and injected into the fiber at nine different locations. (c) Fiber shape is formed by moving two sliders on the manipulation platform. The speckle image at the distal end of the fiber is captured by the output camera. (d) Distal end and the camera are placed on a translation stage allowing to vary the slack of the fiber. (e) 2-D shape of the fiber as imaged by the camera mounted above the manipulation platform is shown.

Algorithm 1 Training and Validation Set Acquisition

---

for *stage_position* **in** (2.5, 5, 7.5, 8.5, 10, 12.5, 15) **do**
    move linear_stage **to** *stage_position*;
    load *range_1,range_2* =
    slider_limits(*stage_position*);
    for *slider_1_position* **in** *range_1* **do**
        move slider_1 **to** *slider_1_position*;
        for *slider_2_position* **in** *range_2* **do**
            move slider_2 **to** *slider_2_position*;
            capture *GT = ground_truth_image*;
            for *focus_position* **in** (1..9) **do**
                move laser **to** *FP = focus_position*;
                capture *SG = specklegram*;
            end
            store *datapoint = (GT, SG, FP)*;
        end
    end
end

Canny edge detection, eroding, and dilating, to finally extract the contour of the fiber shape using methods provided by OpenCV library [21]. We discretized the detected curve in **N + 1** discretization points separated by segments of length **R** and calculated the angles α_i between two consecutive segments using `atan2` function. The angles constituted the ground-truth shape vector of size **N**, which was then used for neural network training.

A single training sample consisted of one specklegram and its corresponding focus coordinates as input and a vector describing the shape as output. Thus, up to nine training samples, each with a different specklegram for the same ground-truth vector, constituting a multispecklegrams, were used independently of each other during training.

1) Shape Reconstruction: To reconstruct the shape of the fiber, the coordinates (x_i and y_i) of the endpoints of the segments were calculated as follows:

$$(x_{i+1}, y_{i+1}) = (x_i + \mathbf{R} * \cos\alpha, y_i + \mathbf{R} * \sin\alpha_i) \qquad (1)$$

with **R** being the length of the line segment and α_i being the angle between consecutive line segments. The quality of the reconstruction is measured as the mean Euclidean distance between the ground-truth coordinates and the reconstructed coordinates.

III. Model Architectures and Training

Multiple models for shape reconstruction based on deep neural networks were devised and used for training. The task of shape reconstruction from recorded specklegrams was formulated as a regression, where the output was represented as an *N*-dimensional real-valued vector.

A. Deep Neural Networks

The models employed in this study are summarized in Table I. The networks consisted of a series of layers, such as fully connected (dense) layers, convolutional layers (Conv2D), physically motivated complex layers, and phase layers. Furthermore, an activation function was applied to the output of each layer. Dropout layer was added to make the neural networks more robust against noise. Batch normalization (BN) was employed to stabilize the training by centering and rescaling the batches of training data.

Fig. 3 illustrates the neural network architecture, denoted as the *complex* network. The motivation of this architecture is the complex transmission matrix that describes the physical wave propagation in a fiber (see [22], [23]). The input consists of a specklegram and the input position (*x* and *y*) of the laser focus. In this model, the 2-D specklegram is flattened and fed into
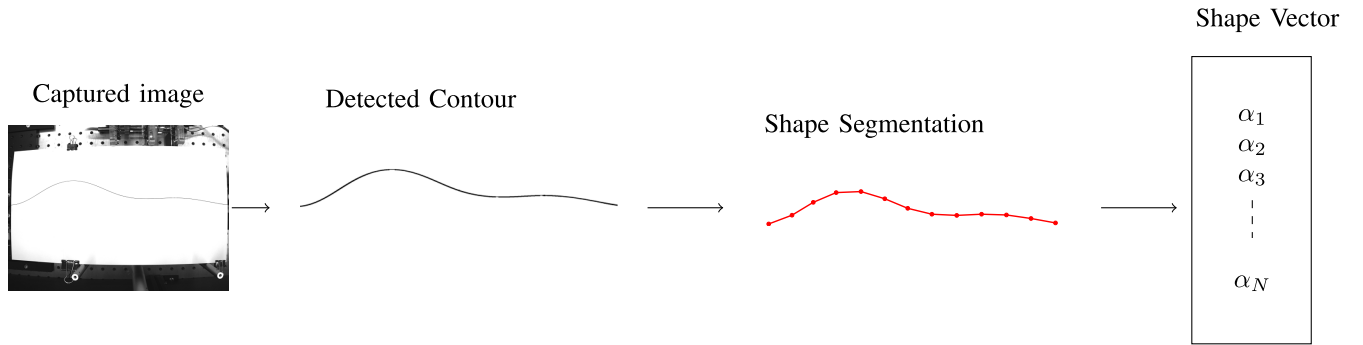
Fig. 2.   Steps for generating the shape vector. The contour is extracted from the captured image of the fiber using OpenCV contour function, and then, the curve is segmented into 12 segments of equal length. The shape vector is calculated from the angle between two consecutive segments.
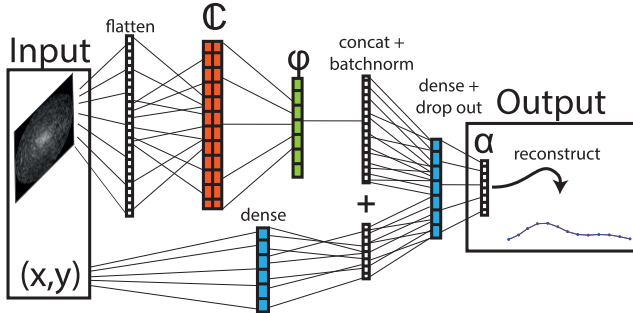

Fig. 3.   Neural network-based shape prediction architecture. First, speckle is projected into the complex valued dense layer, and only the phase feature is taken out. Another dense layer is used to map the phase feature with ground-truth shape vector. Finally, the predicted shape vector is used to reconstruct the fiber shape.

a fully connected complex-valued layer, mimicking the transmission matrix [24]. Subsequently, the output of the complex layer is transformed into the polar representation, and only the argument (phase) is retained. In the next layer, the (phase) features are concatenated with the input position information and further processed by a fully connected (dense) layer. The output of this network is the N-sized angle representation of the shape.

The *dense* network is identical to the *complex* network, except that the complex and phase layers are replaced by a single real-valued dense layer.

Our recursive neural network (RNN) consists of the *complex* network with a recurrent neural network [25] added to the output of the architecture. The recurrent layers are designed to reproduce the shape reconstruction in position space as given by (1), with additional trainable parameters. Thus, the output of the RNN corresponds directly to the coordinates of the discretization points of the fiber shape.

The convolutional neural network (CNN) uses convolutional layers on the full image. Convolutional layers perform convolution operations on the input with a certain number of trainable kernels and are commonly used for computer vision tasks, such as pattern recognition [26]. The outputs of a stack of convolutional layers constitute the extracted features and are further processed in the same way as in the case of the *complex* network.

B. Training

1) Datasets: The experimental data were divided into several sets, as presented in Table II.

In total, 16 964 different shapes with nine specklegrams per shape were recorded. The main dataset was recorded by moving the output stage in 2.5-mm steps from 2.5 to 15 mm, with 0 mm corresponding to a straight fiber. As described in Algorithm 1, at each output stage position, sliders 1 and 2 were used to modify the fiber shape. This large dataset was divided into the training set and a first test set by randomly selecting 447 shape samples. Thus, this test set obeyed the same distribution as the training set and is denoted as the in-distribution (ID) set. An additional dataset was obtained by placing the output stage at 8.5 mm and again using both sliders to generate different fiber shapes. This dataset was denoted as the validation set. The fiber shapes of the validation set were generated using the same slider conditions as for the training set, but the validation set obeyed a slightly different distribution due to a stage position that was not present in the training set. Finally, to generate a test set that would be distinct from all the datasets obtained thus far, the translation stage was placed at the position of 11 mm, and constraints to the fiber shape were modified as follows: the metallic ring that was following slider 2 was detached and placed above an additional fixed magnet mounted between the two sliders, in the center of the top surface, directly below the top view external camera [see Fig. 1 (a) and (c)]. As a result, the shapes were generated by moving slider 1 only. This way, fiber shapes of this dataset differed substantially from the other datasets and can be considered to be out-of-distribution (OOD) relative to the training dataset. Such a dataset (OOD set) presents the largest challenge for a machine learning model.

In summary, the training set was used to train the neural network models; the validation set was used to optimize the parameters of the models (e.g., numbers and size of layers) and the hyperparameters of the training, and ID set and OOD set were used to evaluate the performance of the models.

2) Training: ADAM [27] algorithm was used for optimizing the network training. We trained the networks for up to 1000 epochs using a batch size of 224 and mean square error (MSE) as a loss function. The MSE loss was calculated from the angle representation of the shapes for the *complex, dense*, and CNN models, while the RNN model used the MSE of the x and y coordinates of the discretization points as the loss. The training was performed on a GPU and depending on the model took from 2 to 20 h.

TABLE I
OVERVIEW OF DIFFERENT MODELS FOR FIBER SHAPE RECONSTRUCTION WITH CORRESPONDING NEURAL NETWORK LAYERS.
NUMBERS IN PARENTHESES INDICATE THE NUMBER OF NEURONS

Network Architecture	Complex	Dense	RNN	CNN
Inputs	speckle(flat), position (2)	speckle(flat), position (2)	speckle(flat), position (2)	speckle(img), position (2)
Feature extraction	complex(30), dense(8)	dense(30), dense(8)	complex(30), dense(8)	3*Conv2D + max-pooling + batch-norm, dense(8)
Activation	phase, sigmoid	lin, sigmoid	phase, sigmoid	relu, sigmoid
Concatenation	concat+BN	concat +BN	concat + BN	flatten + concat+BN
Decoder	dense(500)+ drop out	dense(500)+ drop out	dense(500) + drop out	dense(500)+ drop out
Output	dense(N)	dense(N)	dense(N) + RNN	dense(N)

TABLE II
DATASETS USED FOR TRAINING AND TESTING OF THE MODELS. ACCESS TO PUBLICLY
AVAILABLE DATASETS THROUGH [20]

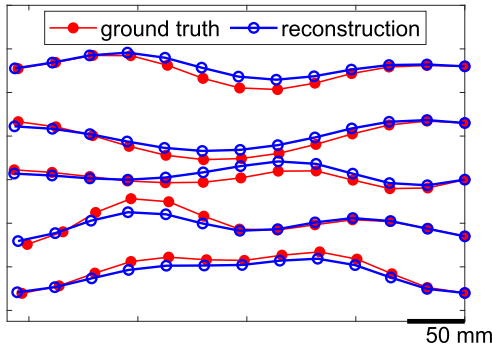| Datasets	Training	Validation	ID set	OOD set
Number of shapes	15657	427	447	433
Stage position (mm)	2.5, 5, 7.5, 10, 12.5, 15	8.5	2.5, 5, 7.5, 10, 12.5, 15	11
Sliders used	1 and 2	1 and 2	1 and 2	1

Fig. 4. Range of representative samples from ID set, corresponding to the distribution of the training set.
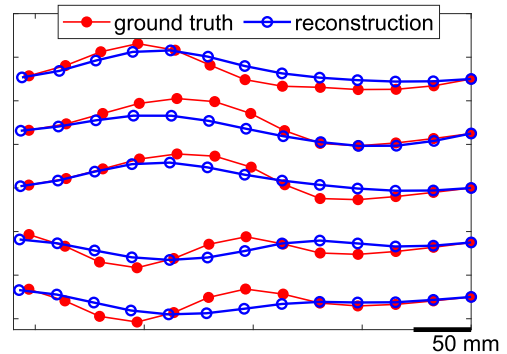

Fig. 5. Range of representative samples from OOD set, containing shapes that are considered OOD of the training set.

IV. RESULTS AND ANALYSIS

In this study, we conducted extensive hyperparameter optimization, varying the neural network architectures, the number of input specklegrams and their resolution, as well as the number of segments used for the reconstruction. Fiber-shape reconstruction capabilities of different trained models are presented below. Deviation of the reconstructed shape from the ground truth (Figs. 4 and 5) was measured by the average Euclidean distance of the discretization points, which we denote as mean position error. By averaging across the discretization points, this figure of merit applies to all models considered in this study. Figs. 4 and 5 illustrate the typical samples of fiber shape reconstruction from the two datasets obtained with the best-performing model. Box plots in Figs. 6–8 show results comparing performance on ID set (left-hand side of the box plots) and OOD set (right-hand side of the box plots). In the presented figures, one parameter was varied, while the others were kept at their optimal value.

A. Neural Network Model

The choice of the neural network model has substantial impact on the quality of the reconstruction, as shown in Fig. 6. The complex network outperformed all other models, especially when applied on test set with the training set distribution (ID set). More importantly, we observed that the

neural networks are also able to generalize to OOD data (see [28]). The larger deviation from the ground truth as can be seen in Fig. 5 compared with Fig. 4 witnesses the difficulty of the generalization. When considering only the mean position error for OOD set, the architectural differences of the complex, dense, and RNN models had little impact. Performance of the CNN was notably inferior compared with the other models. We believe that the translation invariance of CNNs causes them to overlook crucial position-dependent information within the specklegrams. Indeed, while the speckle grains look very similar to each other, the information regarding the shape is encoded in the arrangement of the speckle patterns.

B. Specklegram Resolution

As information about the shape of the fiber is encoded in the pattern of the speckle grains, it follows that the resolution of the specklegram plays a crucial role in the training of the model. During the data acquisition process (see Section II-A), the specklegram was recorded with a resolution of 656 × 656 pixels. We trained our models with downsampled specklegrams obtained using bilinear interpolation and presented the results for the complex model in Fig. 7. When evaluated on the ID set, the model's performance improves with increasing speckle resolution. Since the number of trainable parameters
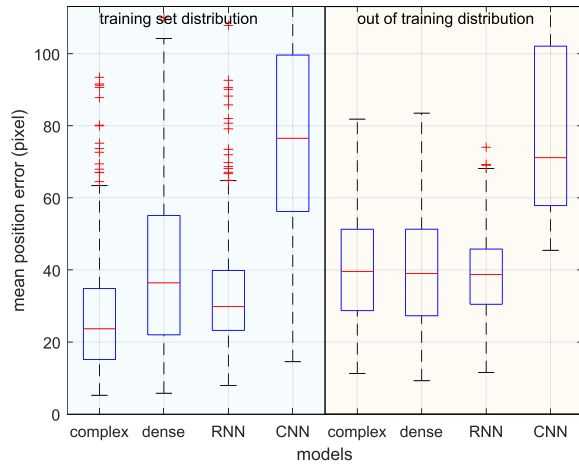
Fig. 6. Comparison of different models, trained with 56 × 56 sized specklegrams and ground-truth discretization with 12 segments. (*L*) Training set distribution = ID set. (*R*) Out of training distribution = OOD set.
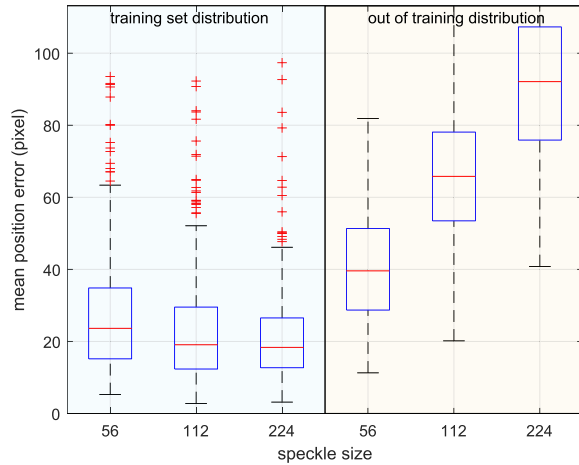

Fig. 7. Reconstruction error as a function of speckle resolution. (*L*) Training set distribution = ID set. (*R*) Out of training distribution = OOD set.

Fig. 8. Reconstruction error as a function of the number of specklegrams used per fiber shape. (*L*) Training set distribution = ID set. (*R*) Out of training distribution = OOD set.

of the complex layer scales with the total number of pixels, improvement in performance could be attributed to overfitting. It is surmised that due to the data acquisition process, samples in the ID set likely have very similar shapes to certain samples in the training set, potentially benefiting from the overfitting of the training process. This hypothesis is supported by the large decrease in generalization capability, as evidenced by the large mean position error as a function of speckle resolution, when evaluated on OOD set. Indeed, the 16-fold increase in the number of trainable parameters of the first layer (for speckle size 56 versus 224) allows the network to memorize the corresponding input–output combinations instead of learning an approximation of the physical model. Another challenge posed by the increasing layer size is the escalating computational cost during training, impacting memory demands, the number of required training epochs, and the duration of each individual epoch.

C. Number of Inputs

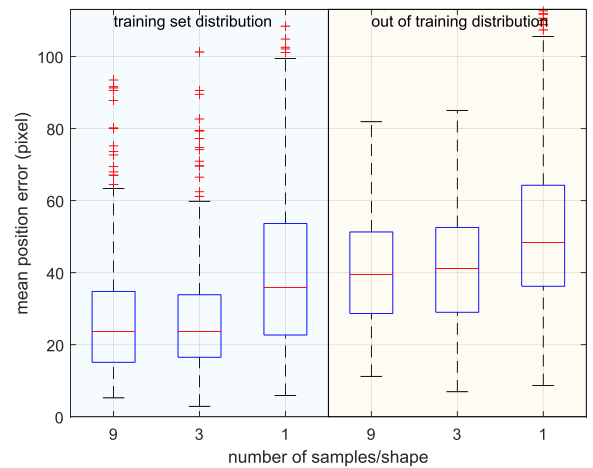With our unique setup, it is possible to inject arbitrary light fields into the fiber, connecting our work to the research of imaging through multimode fibers (see [24]). In preliminary experiments, we have tested various input patterns and found that injecting a single tightly focused beam [see input in Fig. 1 (c)] yields the best results for learning of the shape reconstruction. Here, we analyze how multispecklegrams, i.e., increasing the number of specklegrams for the same fiber shape, could lead to a better reconstruction. Since the network architecture was agnostic to the number of specklegrams per shape, each training sample was used independently during training. However, in testing, the output vectors that were regressed from individual specklegrams were averaged before applying the reconstruction (1). This procedure significantly improved the reconstruction capability of our models. Fig. 8 shows that the reconstruction error increased when the number of specklegrams per fiber shape sample was decreased from 9 to 1. This behavior is observed for both test sets (Fig. 8) and is also found for other models (data not shown). In the case of a single specklegram per fiber shape, the information about the location of the input loses its relevance. Injecting the light at different locations of the fiber facet breaks the rotational symmetry of the fiber, thereby exciting different modes of the fiber. The slight improvement observed for 9 versus 3 specklegrams suggests that going beyond nine would not significantly improve results.

D. Number of Segments

A novelty in our approach is that the shape of the fiber can be parametrized via an arbitrary number of discretization points, i.e., segments of equal length connecting the points. In this section, we study how the number of segments used to approximate the shape of the fiber affects the fidelity of the approximation, and the capabilities of the trained model to reconstruct the shape of the fiber and to generalize to yet unseen types of shapes (i.e., OOD set). Depending on the number of bends, more or fewer segments are needed to faithfully represent the shape of the fiber. In our setup, the number of bends was not defined a priori and varied depending on the shape. Therefore, a minimal number of segments needed to represent the shapes of our datasets were
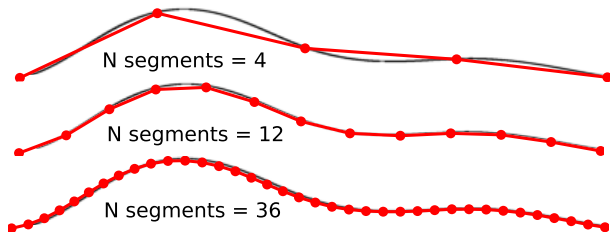
Fig. 9. Examples of fiber shape segmentation.

first established by calculating the area enclosed between the original and discretized fiber shapes as a function of the number of segments. In principle, the enclosed area should tend to zero with the increasing number of segments, as shown in Fig. 9. In practice, however, determining the segment length generates finite-size errors that lead to a residual enclosed area. Fig. 10 shows the calculated areas for the ground-truth shapes contained in the two test sets. The left panel displays the enclosed area over a wider range of curvature (and number of curves), encompassing the entire spectrum of the training set. This breadth is reflected in the substantial variance observed for segmentation with only four segments. The OOD set results shown in the right panel were acquired at a single linear stage position $d_{\text{Stage}} = 11\,\text{mm}$. Therefore, the fiber shapes had similar curviness and yielded values for enclosed areas that are more concentrated. For both test sets, a minimum in the enclosed area is reached at 18 segments, followed by a slow increase for a larger number of segments. This increase stems from the aforementioned finite-size error during the determination of the segment lengths. We have trained and evaluated the complex model with different numbers of segments (Fig. 11). As described above, the mean position error plotted in the figures is an average value over the number of discretization points and, thus, is meaningful even when comparing models with different numbers of discretization points. It appears that the performance of our complex model only weakly depends on the number of segments. As for the other neural network models discussed, we have conducted training with different numbers of segments and have observed qualitatively the same behavior (data not shown). Nevertheless, a trend of larger errors for fewer segments suggests that the reconstruction of shapes within our datasets benefits from a higher number of segments. In other words, the neural network models with an insufficient number of discretization points not only fail to represent the ground truth faithfully but also struggle to learn the inversion of the model underlying the formation of the specklegram. Based on results presented in Figs. 10 and 11, and considerations on computational costs, we have selected the model with 12 segments as the overall optimally high-performing neural network model. Increasing the number of segments to 18 would only marginally improve the reconstruction, while significantly increasing the training time.

V. DISCUSSION AND CONCLUSION

We have generated a large dataset and extensively investigated the reconstruction of fiber shapes using trained neural networks. We have established that multiple input
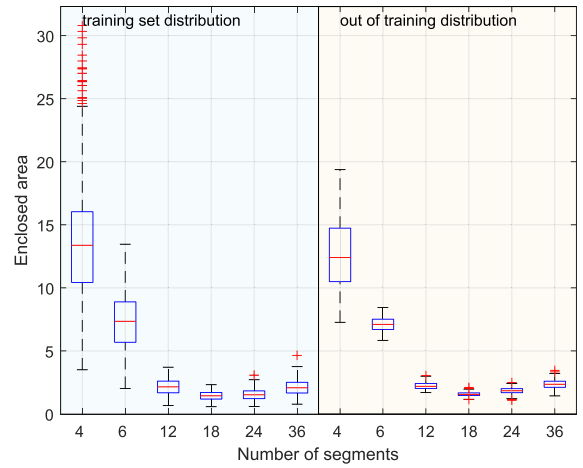

Fig. 10. Fidelity of shape segmentation as a function of the number of segments, measured by the enclosed area between the original and the segmented curve. (L) Training set distribution = ID set. (R) Out of training distribution = OOD set.

Fig. 11. Reconstruction error as a function of the number of segments for the complex model. (L) Training set distribution = ID set. (R) Out of training distribution = OOD set.

specklegrams and a minimum number of discretization segments are required for a faithful reconstruction of the fiber shape. Furthermore, we have observed strong dependence on the resolution of the specklegram, highlighting the risk of overfitting and showcasing the usefulness of OOD test sets. Our results can, thus, serve as a baseline for further developments of machine learning models for our dataset [20]. In quantitative terms, we achieve an average position accuracy of 5.2 mm for ID test set and 7.7 mm for OOD test set, which is an improvement compared with the state-of-the-art results of 13.9 mm reported in [29].

While, on average, the neural network model delivers faithful reconstruction, as shown in Fig. 4, the box plots in Fig. 6 also contain outliers with large reconstruction errors, which correspond to cases of failed reconstruction. We have examined the failed samples closely and have found that local twists of the fiber that occur during data acquisition caused increased reconstruction errors. Indeed, in certain configurations, an arc section of the fiber can undergo a flip during the positioning of a slider, creating a local twist. This means that the image of the shape of the fiber on a 2-D plane does not contain the

full information about the fiber geometry, as it fails to detect possible local twists along the fiber. Fibers that have similar 2-D shapes might have different twist configurations, leading to entirely different speckle patterns. This ambivalence of the training data generally reduces the reconstruction capabilities of the trained model. Another source of ambivalence in the training data is changes in environmental factors, such as temperature, air pressure, and humidity. For instance, temperature affects the refractive index of fiber and, thus, influences the speckle formation process. The impact of temperature on the specklegram has been shown by [30] and [31]. Our results were obtained in a laboratory with the temperature maintained at a constant value; however, we do not exclude that a part of the reconstruction error could be caused by variations in the temperature in the range of $1\,°C$.

In future experiments, environmental parameters could be recorded along with the training data and used in training to eliminate the ambivalence of the training data. Furthermore, the algorithms presented in this work can be equally used to train models that can predict these environmental factors. These factors may play an important role in such applications as in vivo shape sensing of medical devices, such as guidewires and catheters.

Another intriguing research direction would be to attach a multimode fiber to deformable objects of interest [32] (such as an airplane wing, as shown in [1]) and to learn their deformation from the specklegrams generated by the fibers. In such a case, the knowledge of the actual shape of the fiber itself would not be required, as long as it is fully defined by the shape of the object it is attached to (airplane wing). Hence, it should be possible to directly learn the (parametrized) shape of any object in question from the specklegrams generated by the fiber. Here, the main tasks consist of recording a sufficiently general training dataset of parameterized deformations, for example, using algorithms and tools of CV (stereo cameras, scanning, and active illumination). In the future, many more new applications, thus, become feasible.

Acknowledgment

References

[1] M. Amanzadeh, S. M. Aminossadati, M. S. Kizil, and A. D. Rakić, "Recent developments in fibre optic shape sensing," *Measurement*, vol. 128, pp. 119–137, Nov. 2018.

[2] I. Floris, J. M. Adam, P. A. Calderón, and S. Sales, "Fiber optic shape sensors: A comprehensive review," *Opt. Lasers Eng.*, vol. 139, Apr. 2021, Art. no. 106508.

[3] B. Wang, L. Wang, N. Guo, Z. Zhao, C. Yu, and C. Lu, "Deep neural networks assisted BOTDA for simultaneous temperature and strain measurement with enhanced accuracy," *Opt. Exp.*, vol. 27, no. 3, p. 2530, Feb. 2019.

[4] J. Eisenstein, R. Gavalis, P. Y. Wong, and C. G. L. Cao, "Optical design of endoscopic shape-tracker using quantum dots embedded in fiber bundles," *Proc. SPIE*, vol. 7429, pp. 287–296, Aug. 2009.

[5] D. Kinet, P. Mégret, K. Goossen, L. Qiu, D. Heider, and C. Caucheteur, "Fiber Bragg grating sensors toward structural health monitoring in composite materials: Challenges and solutions," *Sensors*, vol. 14, no. 4, pp. 7394–7419, Apr. 2014.

[6] M. A. Davis, A. D. Kersey, J. Sirkis, and E. J. Friebele, "Shape and vibration mode sensing using a fiber optic Bragg grating array," *Smart Mater. Struct.*, vol. 5, no. 6, pp. 759–765, Dec. 1996.

[7] G. M. H. Flockhart, W. N. MacPherson, J. S. Barton, J. D. C. Jones, L. Zhang, and I. Bennion, "Two-axis bend measurement with Bragg gratings in multicore optical fiber," *Opt. Lett.*, vol. 28, no. 6, pp. 387–389, Mar. 2003.

[8] R. G. Duncan et al., "High-accuracy fiber-optic shape sensing," *Proc. SPIE*, vol. 6530, pp. 487–497, Apr. 2007.

[9] P. M. Blanchard et al., "Two-dimensional bend sensing with a single, multi-core optical fibre," *Smart Mater. Struct.*, vol. 9, no. 2, pp. 132–140, Apr. 2000.

[10] K. Wang et al., "Multimode optical fiber sensors: From conventional to machine learning-assisted," *Meas. Sci. Technol.*, vol. 35, no. 2, Nov. 2023, Art. no. 022002.

[11] A. Newaz, M. O. Faruque, R. A. Mahmud, R. H. Sagor, and M. Z. M. Khan, "Machine-learning-enabled multimode fiber specklegram sensors: A review," *IEEE Sensors J.*, vol. 23, no. 18, pp. 20937–20950, Sep. 2023.

[12] A. G. Leal-Junior, A. Frizera, C. Marques, and M. J. Pontes, "Optical fiber specklegram sensors for mechanical measurements: A review," *IEEE Sensors J.*, vol. 20, no. 2, pp. 569–576, Jan. 2020.

[13] H. S. Efendioglu, "A review of fiber-optic modal modulated sensors: Specklegram and modal power distribution sensing," *IEEE Sensors J.*, vol. 17, no. 7, pp. 2055–2064, Apr. 2017.

[14] X. Wang, Y. Wang, K. Zhang, K. Althoefer, and L. Su, "Learning to sense three-dimensional shape deformation of a single multimode fiber," *Sci. Rep.*, vol. 12, no. 1, p. 12684, Jul. 2022.

[15] G. Li, Y. Liu, Q. Qin, X. Zou, M. Wang, and W. Ren, "Feature extraction enabled deep learning from specklegram for optical fiber curvature sensing," *IEEE Sensors J.*, vol. 22, no. 16, pp. 15974–15984, Aug. 2022.

[16] S. Lu, Z. Tan, G. Li, and J. Yang, "A sensitized plastic fiber sensor for multi-point bending measurement based on deep learning," *IEEE Photon. J.*, vol. 13, no. 5, pp. 1–7, Oct. 2021.

[17] S. Razmyar and M. T. Mostafavi, "Deep learning for estimating deflection direction of a multimode fiber from specklegram," *J. Lightw. Technol.*, vol. 39, no. 6, pp. 1850–1857, Mar. 15, 2021.

[18] H. Gao and H. Hu, "Spatially-resolved bending recognition based on a learning-empowered fiber specklegram sensor," *Opt. Exp.*, vol. 31, no. 5, p. 7671, Feb. 2023.

[19] B. Hadad, D. Marima, N. Magal, A. Eyal, and A. Bahabad, "Toward multimode-fiber shape sensing," *Opt. Lett.*, vol. 48, no. 5, p. 1160, Mar. 2023.

[20] V. Volchkov, B. Javot, S. Bhattarai, K. Bierig, C. G. L. Cao, and I. Oreshnikov, "Specklegrams and groundtruth for fiber shape reconstruction," Max Planck Soc., Munich, Germany, 2024, doi: 10.17617/3.N4FHS6.

[21] G. Bradski, "The OpenCV library," *Dr. Dobb's J. Softw. Tools*, 2000. [Online]. Available: https://github.com/opencv/opencv

[22] O. Moran, P. Caramazza, D. Faccio, and R. Murray-Smith, "Deep, complex, invertible networks for inversion of transmission effects in multimode optical fibres," in *Advances in Neural Information Processing Systems*, vol. 31, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., Red Hook, NY, USA: Curran Associates, 2018.

[23] S. Liu, Y. Sun, W. Liu, F. Xiao, and H. Song, "Information distribution on regions of speckle patterns for imaging of multimode fiber," *Heliyon*, vol. 9, no. 2, Feb. 2023, Art. no. e13357.

[24] M. Plöschner, T. Tyc, and T. Čižmár, "Seeing through chaos in multimode fibres," *Nature Photon.*, vol. 9, no. 8, pp. 529–535, Aug. 2015.

[25] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," *Proc. Nat. Acad. Sci. USA*, vol. 79, no. 8, pp. 2554–2558, 1982.

[26] A. Dhillon and G. K. Verma, "Convolutional neural network: A review of models, methodologies and applications to object detection," *Prog. Artif. Intell.*, vol. 9, no. 2, pp. 85–112, Jun. 2020.

[27] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. 3rd Int. Conf. Learn. Represent.*, San Diego, CA, USA, Jan. 2017, pp. 1–15.

[28] J. Liu et al., "Towards out-of-distribution generalization: A survey," 2021, *arXiv:2108.13624*.

[29] D. Bender, U. Çakir, and E. Yüce, "Deep learning-based fiber bending recognition for sensor applications," *IEEE Sensors J.*, vol. 23, no. 7, pp. 6956–6962, Apr. 2023.

[30] F. Feng et al., "AI-assisted spectrometer based on multi-mode optical fiber speckle patterns," *Opt. Commun.*, vol. 522, Nov. 2022, Art. no. 128675.

[31] L. V. Nguyen, C. C. Nguyen, G. Carneiro, H. Ebendorff-Heidepriem, and S. C. Warren-Smith, "Sensing in the presence of strong noise by deep learning of dynamic multimode fiber interference," *Photon. Res.*, vol. 9, no. 4, p. B109, Apr. 2021.

[32] Z. Ding and Z. Zhang, "2D tactile sensor based on multimode interference and deep learning," *Opt. Laser Technol.*, vol. 136, Apr. 2021, Art. no. 106760.

Karin Bierig received her qualification as a Physical Technical Assistant in Isny, Germany. She is a Technician with the Empirical Inference Department, Optics and Sensing Laboratory, Max Planck Institute for Intelligent Systems, Tübingen, Germany.

Caroline G. L. Cao (Member, IEEE) received the Ph.D. degree in mechanical and industrial engineering from the University of Toronto, Toronto, ON, Canada, in 2002.

She is a Researcher, an Educator, and an Entrepreneur with extensive experience in the design and development of medical systems and devices, and enabling technology (AR/VR, simulation and training, decision aid, haptics, sensory substitution, navigational aid, and robotics) for minimally invasive surgery and robotic surgery. She is an expert in human factors engineering, specializing in human–machine interaction and performance evaluation.

Prof. Cao is an Elected Fellow of the Human Factors and Ergonomics Society and the International Ergonomics Association. She was a recipient of the National Science Foundation Career Award, the U.S. Fulbright Scholar Award, and the International Chair of RFI ATLANSTIC 2020.

Ivan Oreshnikov received the master's degree in physics from Saint Petersburg State University, Saint Petersburg, Russia, in 2010.

Then, he worked as a Research Engineer and a Software Engineer in various companies across the IT industry. From 2018 to 2022, he was a Research Engineer with the Max Planck Institute for Intelligent Systems, Tübingen, Germany. He is currently a Software Engineer with Bolt, Berlin, Germany.

Bernard Javot (Member, IEEE) received the master's degree in robotics and the Ph.D. degree in haptics from Université Pierre et Marie Curie, Paris, France, in 2011 and 2016, respectively.

He earned the highest-level teaching accreditation in France in 2005 and worked as a manufacturing teacher in Paris for ten years. Then, he joined the Max Planck Institute for Intelligent Systems, Stuttgart, Germany, where he now works as a Senior Research Engineer. His research interests include inventing actuators and sensors for haptics.

Shreeram Bhattarai received the master's degree in computer science from the University of Bonn, Bonn, Germany, in 2020, specializing in computer vision and deep learning.

He is currently working as a Data Scientist with CIMPA GmbH, Hamburg, Germany. He is passionate about developing innovative solutions in the field of artificial intelligence, particularly focusing on applying deep learning techniques to complex problems. His research interests include image analysis, computer vision, and machine learning.

Valentin V. Volchkov received the Ph.D. degree in quantum physics from the University of Stuttgart, Stuttgart, Germany, in 2013.

In the following years, he investigated as a Postdoctoral Marie Skłodowska-Curie Fellow the propagation of quantum matter in disordered media with the Institut d'Optique, Palaiseau, France. Since 2016, he has been a Co-Head of the Optics and Sensing Laboratory, Max Planck Institute for Intelligent Systems, Tübingen, Germany. His research interests include machine learning in and with optics covering fields from computational photography to neuromorphic computing.