

Artificial Intelligence in Classical and Quantum Photonics

Federico Vernuccio, Arianna Bresci, Valeria Cimini, Alessandro Giuseppe, Giulio Cerullo, Dario Polli,* and Carlo Michele Valensise*

The last decades saw a huge rise of artificial intelligence (AI) as a powerful tool to boost industrial and scientific research in a broad range of fields. AI and photonics are developing a promising two-way synergy: on the one hand, AI approaches can be used to control a number of complex linear and nonlinear photonic processes, both in the classical and quantum regimes; on the other hand, photonics can pave the way for a new class of platforms to accelerate AI-tasks. This review provides the reader with the fundamental notions of machine learning (ML) and neural networks (NNs) and presents the main AI applications in the fields of spectroscopy and chemometrics, computational imaging (CI), wavefront shaping and quantum optics. The review concludes with an overview of future developments of the promising synergy between AI and photonics.

intelligence, and surprisingly it is still an open point of discussion among experts.^[2] The general consensus defines AI as the science that studies artificial systems/machines that imitate human/intelligent behavior. Among the various branches of AI, the one that had the most impact on several different scientific and engineering fields is machine learning (ML), the science that studies how computers can be automatically trained to solve complex tasks starting from the analysis of data. The recent rise in popularity of ML is related to the new opportunities opened by deep learning (DL), a methodology that exploits the advancements in computing power to solve highly complex tasks,

such as computer vision, speech recognition, and self-driving cars, and that aims to approximate nonlinear transfer functions, leveraging their data-driven nature.

Photonics is among the most active and promising fields in science, technology and engineering. The combination of AI techniques and photonics has led to groundbreaking developments in many applications and provides huge opportunities for both fields. Indeed, on the one hand photonics can be used to generate rich data sets for ML computational tasks, on the other hand photonic systems are an interesting platform for AI implementations.^[3,4] In this context, two of the most investigated fields have been fiber optics communications and image processing for medical diagnosis.

Several AI-based techniques have been developed with the purpose of improving the performances of optical communication systems, mainly focused on the control and management of photonic devices. Many research articles and reviews have been published on this topic.^[5–10] Mata et al.^[5] reviewed different AI implementations in optical networks communication. Some techniques can help improving the configuration and operation of network devices, others are used for optical performance monitoring, modulation format recognition or fiber nonlinearities mitigation and quality of transmission estimation. As for the equalization of nonlinear wavefront distortion in optical communications, a relevant contribution has been provided by the works of the Nakamura group. They proposed a neural network (NN) model featuring a sole hidden layer to compensate for self-phase modulation distortions in optical multi-level signals.^[11] Interestingly, they investigated the effect of the hidden layer size on the nonlinear equalization task. As the input power increases and the self-phase modulation effect distorts the transmitted signals more severely, a higher number of hidden layer neurons is

1. Introduction

Artificial Intelligence (AI) is undoubtedly one of most active research fields of the recent years, able to gather unprecedented investments and generate large economic impacts.^[1] The definition of AI is very broad, just as is the definition of

F. Vernuccio, A. Bresci, G. Cerullo, D. Polli
Department of Physics
Politecnico di Milano
P.zza Leonardo da Vinci 32, Milan 20133, Italy
E-mail: dario.polli@polimi.it

G. Cerullo, D. Polli
Istituto di Fotonica e Nanotecnologie (IFN) CNR
Milano 20133, Italy

V. Cimini
Department of Physics
Sapienza Università di Roma
P.le Aldo Moro 5, Rome 00185, Italy

A. Giuseppe
Department of Computer
Control and Management Engineering
Sapienza Università di Roma
Via Ariosto 25, Rome 00185, Italy

C. M. Valensise
Enrico Fermi Research Center
Piazza del Viminale 1, Rome 00184, Italy
E-mail: carlo.valensise@cref.it

 The ORCID identification number(s) for the author(s) of this article can be found under <https://doi.org/10.1002/lpor.202100399>

© 2022 The Authors. Laser & Photonics Reviews published by Wiley-VCH GmbH. This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

DOI: 10.1002/lpor.202100399

required for an efficient compensation of the optical communication. Their further studies demonstrated how a four-layer NN nonlinear equalizer is more overfitting-prone than a three-layer NN model^[12] and how multi-level 4-ary pulse-amplitude modulation signals strongly limit the NN equalizer overfitting that typically occurred in the case of pseudo-random binary signals.^[13] Ji et al.^[6] proposed a novel multi-tasking architecture able to handle several aspects of optical networks management, leveraging AI techniques to produce self-adaptive and self-managed operations. Optical networks feature huge dynamicity, complexity and heterogeneity due to the use of advanced coherent techniques, so that AI proved a fundamental tool for their management. Wang and Zang^[7] focused on state-of-the-art DL algorithms and highlighted the contributions of DL to optical communications. In particular, they reviewed multiple DL applications in optical communications, such as convolutional neural networks (CNNs) for image reconstruction and recurrent neural networks (RNNs) for sequential data analysis. Moreover, they introduced a data-driven channel modeling method to replace the conventional block-based approach and improve the end-to-end learning performance and a generative adversarial network (GAN) for data augmentation (DA). Eventually, they described a deep reinforcement learning (DRL) algorithm used for network automation.

Artificial intelligence has also played a crucial role in medical diagnostic practices thanks to its ability to restrict the impact of human bias and increase the diagnosis reliability and accuracy.^[14,15] Rather than replacing the role of medical doctors, AI algorithms proved to be very powerful in supporting them, providing image preselection, preprocessing, and classification while increasing time-effectiveness with low cost.^[16–22] Moreover, optical data may reach high levels of complexity and dimensionality, leading to error-prone interpretations even in the case of experienced operators. In such cases, the employment of AI engines for optical data decoding is fundamental.^[15] CNN models have been used to increase the quality of medical images, thus enhancing the accuracy of further traditional classification procedures and limiting the occurrence of incorrect diagnoses.^[23] DL algorithms of this fashion turned out to be pivotal in diagnostics when imaging methods featuring a high signal-to-noise ratio or a complex data structure were involved, such as functional magnetic resonance imaging.^[24] In particular, deep learning has played a key role in one of the most widely employed diagnostic tools: X-ray computed tomography (CT). Since X-ray-related radiation risk is a concern, the use of a low X-ray tube current would be preferable but it would lead to poor image quality, thus preventing its routine application. CNN algorithms were employed to filter and reconstruct low-dose CT images, coupling diagnosis accuracy with a less invasive approach.^[25,26] Alongside CNNs, RNNs have been used for image diagnostics combining multiple techniques (e.g., magnetic resonance imaging coupled with positron emission tomography), thus revealing the power of such models in extracting valuable information from multimodal medical data.^[27]

Despite these promising results, the full benefits of the combination between AI and photonics have not yet been reaped. This is partly due to the lack of a deep understanding of AI methodologies by photonics researchers. This paper aims to fill this gap by providing a basic introduction to AI and reviewing the most significant contributions of AI and ML to classical and quantum

photonics. In particular, Section 2 provides an overview on the problems studied in AI and ML, giving the theoretical foundations of the algorithms employed to learn from data and environment: supervised, unsupervised and reinforcement learning. Section 3 provides the readers with the mathematical and statistical fundamentals of the most popular NN models encountered in photonics, thus introducing the basic tools required to understand the applications reviewed in the following sections. Section 4 explores DL applications in spectroscopy as powerful tools for denoising, artifact removal and spectral chemometrics. Section 5 details how AI can assist wavefront shaping when light propagates in media as well as computational imaging. In the same section, control of light propagation in multi-modal fibers (MMFs) is discussed, presenting some recent main applications related to the field. Section 6 deals with AI applied to quantum optics. It reviews the use of ML techniques for the generation of quantum states of light, their application in the field of metrology and sensing and the automated classification and characterization of optical quantum states. Eventually, Section 7 provides an overview on photonic computing, showing how photonic could play a major role in future developments of AI.

2. Machine Learning Fundamentals

A ML algorithm is able to learn information from data.^[28] Depending on the kind of data available, ML tasks are divided in the following macro-areas.

- Supervised learning, where the ML model is provided with a dataset containing input–output pairs. The output data, or labels, enable the evaluation of the model performance during the training. Supervised learning can be exploited to approximate the complex or unknown function mapping the input data to the output. Supervised learning tasks are further divided into regression tasks, where the model is required to predict some numerical output values starting from the input, and classification tasks, where the model is asked to specify which class/category the given input belongs to. Some supervised learning techniques have been used to estimate the quality of transmission of an optical communication system or for resource allocation in data centers.^[5]
- Unsupervised learning, where ML models deal with the extraction of information from the data without any target value or label available. Unsupervised learning tasks range from clustering^[29] to anomaly detection and feature learning.^[30–32] For instance, methods which belong to this type of learning have been used for optical performance monitoring, modulation format recognition and impairment mitigation.^[5]
- Reinforcement learning (RL) is a specialized ML area that deals with the control of a dynamical system,^[33] where the model is trained to find a control law for the system so that some objective is optimized. RL, and in particular its variant that employs deep neural networks (DNNs), deep RL, finds application in various complex tasks, such as robotics^[34] and autonomous driving.^[35] For instance, Q-learning, which is a reinforcement learning technique that aims to find the optimal quality value (Q-value) of an action selection policy, has been used for path and wavelength selection in the context of optical burst-switched networks.^[5]

The power of ML solutions is their capability to generalize the information they inferred from the available data over previously unseen data, functionally solving the considered task for arbitrary inputs. Over the years, different ML algorithms, models and methodologies were proposed^[36] to solve tasks from almost every scientific domain, but the recent and unprecedented rise in popularity that ML has experienced is mostly due to the results that DNNs, sometimes also referred as Artificial Neural Networks (ANNs), were able to attain in solving new and complex problems. In the following we describe the basics of ML tools and DNNs. The description will cover all the relevant aspects keeping apart the more formal ones. Nevertheless, the reader is referred to a detailed literature along the discussion. An excellent resource in this respect is ref. [37], offering also real code examples. The goal is to give the optics practitioner a background about ML, before describing the main applications demonstrated in recent literature.

The basis of every ML problem is a dataset with N entries $D = (\mathbf{X}, \mathbf{y}) = \{(\mathbf{x}_i, y_i), i = 1 \dots N\}$, where \mathbf{X} is a matrix whose rows correspond to data instances, whereas the columns are the features of the dataset, namely the variables or attributes of the instances. On the other hand, y_i is the corresponding ground-truth vector, representing the ideal output of the model. The features of \mathbf{X} can be numerical and/or categorical (i.e., input attributes encoded in the form of discrete numerical values) depending on the context, and constitute the independent variables. The goal of ML algorithms is to approximate the map $f : \mathbf{x}_i \rightarrow y_i$, acting on a set of parameters θ . For a single training example, the error between the prediction and the ground truth is quantified by means of the loss function $\mathcal{L}_i(f(\mathbf{x}_i, \theta), y_i)$. The parameters of the NN are adjusted in order to minimize the cost function $C(f(\mathbf{X}, \theta), \mathbf{y})$ (or $C(\theta)$), which is the average of the loss functions $\mathcal{L}_i(f(\mathbf{x}_i, \theta), y_i)$ over the overall training dataset. In formulae

$$C(f(\mathbf{X}, \theta), \mathbf{y}) = \frac{1}{N_{\text{train}}} \sum_{i=0}^{N_{\text{train}}} \mathcal{L}_i(f(\mathbf{x}_i, \theta), y_i) \quad (1)$$

where N_{train} is the total amount of instances in the training set. Indeed, the training of ML models requires the dataset to be partitioned in two independent sets $D = D_{\text{train}} \cup D_{\text{test}}$, respectively the training set and test set. The former is used to optimize the set of parameters θ , while the second is used to evaluate the performances of the model on new, unseen data. A typical training-test split is $\approx 80/20\%$. In addition, a portion of the training set (e.g., 20%) is used as validation set, meaning that it is used for an unbiased evaluation of the model during training and for fine tuning of the model hyperparameters, which are the untrainable parameters that define the topology of the network.

These procedures are crucial to assess the model performances and the overall goodness of fitting the data. In this respect the relevant quantities are the training error $E_{\text{train}} = C(\mathbf{X}_{\text{train}}, \mathbf{y}_{\text{train}})$ and the test error $E_{\text{test}} = C(\mathbf{X}_{\text{test}}, \mathbf{y}_{\text{test}})$. A good ML model is the one that enables one to perform a reliable prediction on previously unseen data. Model performances depend on a number of factors, among which the amount of data available (the cardinality of the training set), the number of parameters available to the model (the model complexity) and the number of optimization iterations carried out. In order to obtain good pre-

dictions from the data, both E_{train} and E_{test} should be monitored. In fact, while the minimization of E_{train} ensures that the model is learning the mapping f , the ability to perform well on new data is expressed by E_{test} . Typically E_{train} results slightly lower than E_{test} . Nevertheless if $E_{\text{test}} \gg E_{\text{train}}$ the model is overfitting the training set, that is, the model is using its representation power to store information related to fluctuations of the training set. Overfit significantly limits the predictive power of a model, and therefore has to be avoided. To mitigate overfitting one may increase the number of data points available to train the model, or reduce the model complexity by reducing the number of parameters, or applying regularization techniques (e.g., L_1 , Lasso regularization, and L_2 , Ridge regularization^[38]). This crucial aspect of ML is usually referred as bias-variance trade-off.

The goal of the model is to find a set of parameters that minimizes $C(\theta)$, thus maximizing the model performance accuracy by leading to a minimum average error between ideal and predicted outputs. Gradient descent (GD) is the typical procedure used to compute θ . The concept of this technique is that for every iteration we compute the cost function $C(\theta)$. As an example, if the chosen loss function is the mean squared error (MSE) we have

$$C(\theta) \equiv \text{MSE} = \frac{1}{N_{\text{train}}} \sum_{i=1}^{N_{\text{train}}} (f(\mathbf{x}_i, \theta) - y_i)^2 \quad (2)$$

Given the total cost for the current iteration, we can compute an update in our parameters at iteration $(t + 1)$ in the opposite direction of the gradient (∇) of $C(\theta)$ with respect to the parameters at iteration t , that is, in the direction of the minimum of $C(\theta)$. In formulae

$$\theta_{t+1} = \theta_t - \eta \nabla_{\theta_t} C(\theta_t) \quad (3)$$

where η is the learning rate of the algorithm, a hyperparameter (i.e., a parameter that controls the learning process and is not part of weights and biases to be optimized during learning) that controls how much the parameters are updated in response to the estimated error. This basic update rule has been widely investigated, and several improvements have been introduced. In particular, to avoid the computation of the gradient over the entire dataset that would be computationally very expensive, random subsets of the dataset (mini-batches) can be used at each iteration. This reduces the computational cost and introduces stochasticity in the training, which in turn reduces overfitting. To mitigate the risk of being trapped in local minima of the cost function, some optimizers add “momentum” to the update rule as an exponentially weighted average over the previous values of the gradient. Among these advanced optimizers, one of the most popular is the adaptive moment estimation optimizer (ADAM)^[39]: the learning rate is adapted based on the average first moment (the mean) and the average second moment (the uncentered variance) of the gradient of $C(\theta)$. It showed promising results in terms of regularization and acceleration of NNs convergence in a broad range of applications, adapting the learning rate during the training process. A comprehensive description of all these aspects is given in ref. [28].

The described training procedure is valid across the entire range of ML algorithms. A complete review of all of them is beyond the scope of this paper, but we mention that before the

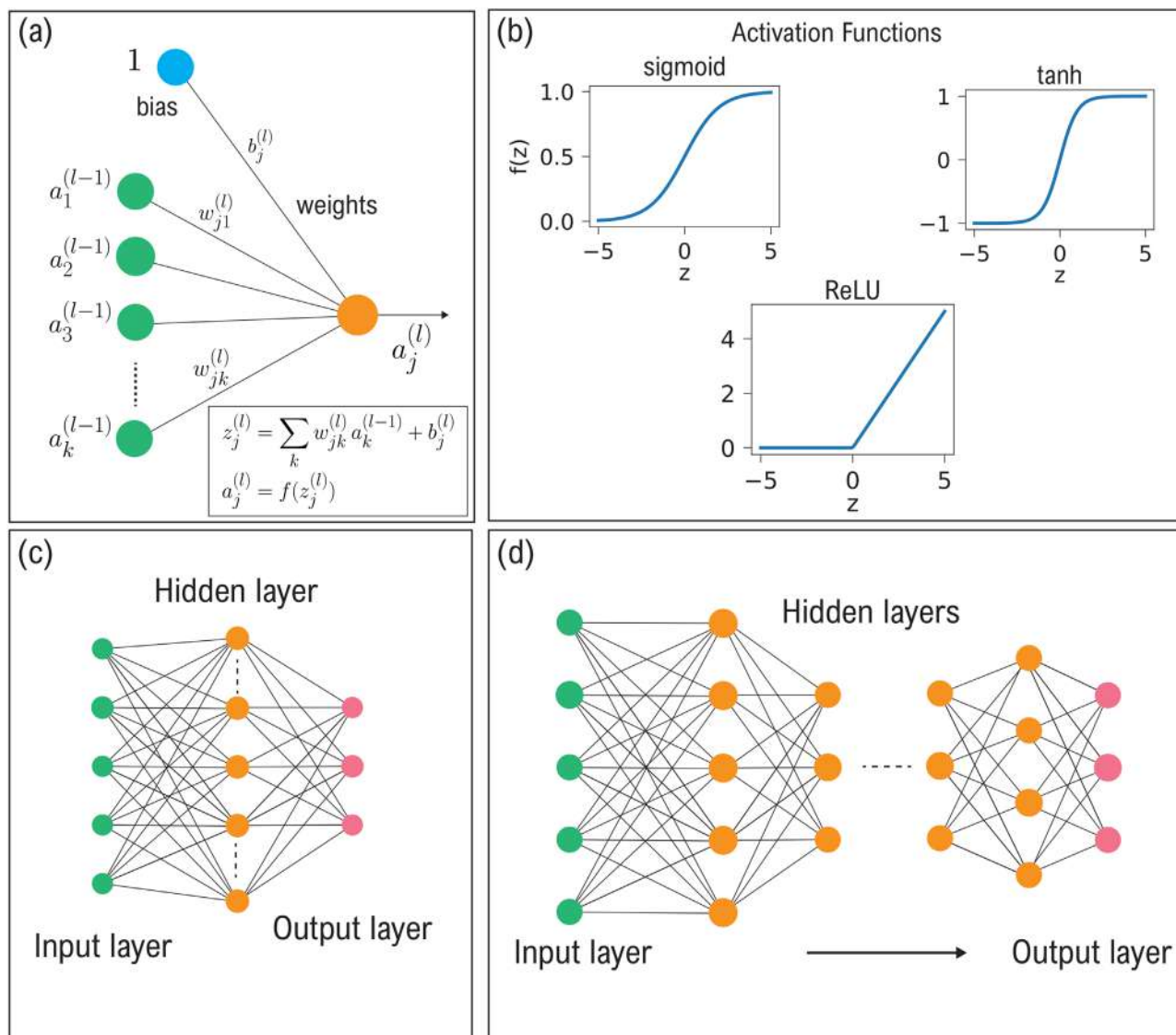


Figure 1. a) Illustration of a neuron j at layer l ; b) Plots of typical non-linear activation functions: sigmoid, tanh, ReLU; c) Scheme of a shallow NN featuring a single hidden layer; d) Scheme of a deep NN featuring several hidden layers.

rise in popularity of DL and NNs, the standard approach to solve complex regression and classification tasks was based on support vector machines (SVMs).^[40] SVMs popularity was mostly due to their usage for classification tasks of a peculiar loss function, the Hinge loss, that ensures the maximization of the geometric margin between classes, obtaining a so-called “maximal margin classifier” with unparalleled performances. Moreover, thanks to the use of kernels^[41] and the so-called “kernel-trick,” SVMs are able to efficiently re-conduct their analysis into a higher-dimensional feature space, where the classification task results simpler, greatly improving their performance. In fact, kernels allow one to map the input features into a feature space without the need to explicitly compute them, but rather performing the inner product between images of the input points. By means of this “trick” it has been possible to tackle complex problems, previously intractable, such as exploring a protein landscape.^[42] In the next section we

discuss how DNNs instead are capable of defining automatically the features required to solve the assigned task.

3. Neural Networks Fundamentals

DL is a ML methodology that employs DNNs to solve ML tasks. Despite the name and the structure somehow recalls the brain,^[36] the similitude is actually rather poor. DNNs are a class of powerful function approximators. A NN is obtained by the combination of simpler objects, the neurons (see Figure 1a). Each neuron receives a series of real numbers as input \mathbf{x} , computes their weighted average with a set of weights \mathbf{w} and a bias b , $z = \mathbf{x}^T \mathbf{w} + b$, and outputs a number $f(z)$ obtained applying a non-linear function f , called activation function. The bias term b is often included in the set of weights \mathbf{w} to compact the notation, considering an extra 1 in the vector \mathbf{x} (see Figure 1a). The

nonlinear function is the basis of the approximation power of NNs. Typical nonlinear activation functions, reported in Figure 1b, are the sigmoid $\sigma(z)$, the hyperbolic tangent $\tanh(z)$, and the rectified linear-unit, $\text{ReLU}(z)$. They are defined as

$$\sigma(z) = \frac{1}{1 + e^{-z}}; \quad \tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}};$$

$$\text{ReLU}(z) = \max(0, z) \quad (4)$$

The ability of NNs to approximate complex functions is guaranteed by the universal approximation theorem, formulated by George Cybenko in 1989.^[43,44] This result states that a NN consisting of an input layer, a single hidden layer and an output layer can approximate any arbitrary function between its input and output, provided that its hidden layer is adequately large. The problem with the application of this theorem lies in the size of the hidden layer, that grows exponentially with the complexity and non-linearity of the function, rapidly reaching unfeasibly large levels. DNNs aim at solving this dimensional issue by stacking multiple hidden layers, exponentially increasing the approximation capabilities of each neuron of the deeper layers, hence compensating for the limited number of neurons available on each layer.

In practice, the best performances are achieved by structuring multiple layers of neurons with finite width (see Figure 1c,d) obtaining a “deep” architecture, from which the nomenclature “deep neural network” stems. Each neuron is linked to all the neurons of the previous and following layer in a so-called “fully-connected” architecture. Since the information travels in one direction only, namely from the input layer to the output layer with no backward cycles in-between, such models are often referred to as “feed-forward” neural networks. As typical of most classical ML solutions (e.g., SVM), shallow networks, constituted by a sole hidden layer, base their analysis on a process called “feature extraction,” in which input data have to be significantly pre-processed and transformed to extract some nontrivial information. DNNs instead require a very limited (if any) feature extraction pre-processing, and automatically extract and weigh the relevant features from the input to perform the assigned task.

NNs are trained in the same way described above, that is, by adapting the weights in order to minimize the cost function $C(\theta)$. However, differently from other ML techniques, NNs are characterized by having a very high number of parameters; small-sized NNs have hundreds or thousands of parameters, moderately large NNs need to train a few millions of parameters, while the largest reach hundreds of millions of parameters. Despite the complex, interconnected structure of neurons and weights, the differentiability of activation functions ensures that NNs can be trained through stochastic gradient descent (SGD).^[45] The backpropagation algorithm^[28] implements very efficiently the computation of the gradient of $C(\theta)$. This algorithm leverages dynamic programming and efficient matrix multiplication, performed on graphical processing units (GPUs) or tensor processing units (TPUs). At the heart of backpropagation is an expression for the partial derivative of the cost function with respect to any weight or bias of the network. This expression gives us detailed insights on the overall behavior of the network while changing the weights and the biases. The typical notation to refer to all the weights in the networks is w_{jk}^l . It denotes the weight from the k th neuron

in the $(l - 1)$ th layer to the j th neuron in the l th layer. A similar notation is used for the biases and the activation functions. Indeed, b_j^l denotes the bias of the j th neuron in the l th layer, while a_j^l represents the activation of the j th neuron in the l th layer (see Figure 1a). Using these notations, the activation of the j th neuron in the l th layer is related to the activations in the $(l - 1)$ th layer by the equation

$$a_j^l = f\left(\sum_k w_{jk}^l a_k^{(l-1)} + b_j^l\right) \quad (5)$$

where the sum is over all the k neurons of the $(l - 1)$ th layer and f is the chosen activation function (see Equation (4)). Typically, one may refer to the input of the activation function as z_j^l , that is $z_j^l = \sum_k w_{jk}^l a_k^{(l-1)} + b_j^l$.

Defining the error introduced by the j th neuron in the l th layer as

$$\delta_j^l \equiv \frac{\partial C}{\partial z_j^l} = \frac{\partial C}{\partial a_j^l} \frac{\partial a_j^l}{\partial z_j^l} \quad (6)$$

the backpropagation algorithm is based on four fundamental equations. They are:

1. An equation for the error δ_j^L of the output layer L

$$\delta_j^L = \frac{\partial C}{\partial a_j^L} f'(z_j^L) \quad (7)$$

where $f'(z_j^L) = \left(\frac{df}{dz}\right)_{z_j^L}$ expresses how fast the activation function is changing at z_j^L .

2. An equation for the error δ_k^{l-1} of the k th neuron in the previous layer in terms of the error δ_j^{l+1} of the j th neuron in the next layer

$$\delta_k^{l-1} = \left(w_{jk}^{l+1} \delta_j^{l+1}\right) f'(z_j^l) \quad (8)$$

3. An equation for the partial derivative of the cost function with respect to any bias in the network

$$\frac{\partial C}{\partial b_j^l} = \frac{\partial C}{\partial z_j^l} \frac{\partial z_j^l}{\partial b_j^l} = \frac{\partial C}{\partial z_j^l} \cdot 1 = \delta_j^l \quad (9)$$

4. An equation for the partial derivative of the cost function with respect to any weight in the network

$$\frac{\partial C}{\partial w_{jk}^l} = \frac{\partial C}{\partial z_j^l} \frac{\partial z_j^l}{\partial w_{jk}^l} = \delta_j^l a_k^{l-1} \quad (10)$$

Eventually, after the input has been propagated to the network by computing activation functions $a_j^l = f(z_j^l)$, the output error is computed through Equation (7). Applying the chain rule of partial derivatives from the output layer backward,^[46] the error is backpropagated using Equation (8). Finally, Equations (9) and

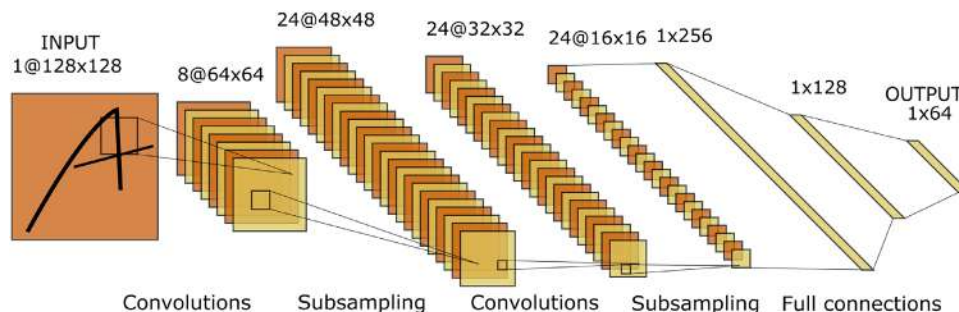


Figure 2. LeNet architecture, featuring two sets of convolutional and subsampling layers, followed by two fully-connected layers and finally an output layer. As highlighted in the image, convolutional filters or kernels have a local connectivity with their input, which enables feature extraction in a spatially invariant way. Adapted with permission.^[50] Copyright 2020, Springer Nature B.V.

(10) are used to compute the partial derivatives of the cost function with respect to any weight and bias.

3.1. Convolutional Neural Networks

CNNs are one of the most popular Network architectures utilized in DL, and in particular in imaging-related tasks. CNNs were first introduced in ref. [47]. Unlike standard NNs, CNNs include the so-called convolutional layers, which may either constitute their entire end-to-end model architecture or precede standard fully-connected layers. Two are the distinctive features of convolutional layers:

- Each neuron has only a local connectivity with the previous layer, in the sense that its inputs come from a small set of (neighboring) neurons from the previous layer.
- In every layer, all neurons share the same weights. This set of weights takes the name of filter or “kernel,” and several different filters may be placed in the same convolutional layer to operate on the same input, as in **Figure 2**.

The combination of these two properties, depicted in **Figure 2**, allows the characterization of CNNs in terms of the number of filters employed at each layer. Among the advantages of CNNs, we mention that their number of weights is significantly reduced compared to standard NNs, depicted in **Figure 1c,d**, allowing much deeper and complex architectures to be deployed. Additionally, the local connectivity of the neurons allows the network to better localize features (e.g., a face) in their input, while the sharing of the weights provides spatial invariance properties to their analysis (e.g., a face is recognized independently from its location in the input image). CNNs automatically extract features from data in the form of the so-called feature maps (**Figure 2**), which are the result of a kernel being cross-correlated to its input. It was observed that, thanks to the local connectivity of neurons, deeper layers tend to capture in their feature maps more complex concepts (e.g., a smiling face, a particular animal) with respect to the shallower ones that focus on basic features (e.g., a color pattern, an edge), directly mapping the number of their layers with their analysis capabilities.^[48] A complete survey of CNNs architectures is beyond the scope of this work (we refer the interested reader to ref. [49]), but we mention in the following the most popular ones.

For image analytic tasks, such as image segmentation, object detection & tracking^[51,52] and complex tasks such as cell

counting,^[53] encoder–decoder architectures as the one depicted in **Figure 3** proved to be among the most effective solutions. In this classic architecture, the CNN is divided into two sections: the first part, named encoder, reduces the dimensions of the feature maps at each layer (in **Figure 3** this is done by the “MaxPooling” layers that in combination with convolutional layers reduce the 128×128 pixels input image to a mere 32×32 pixels image) and increases their amount (in **Figure 3** the orange blocks have 32, 64, and 128 filters, each one generating a feature map). The second portion of the CNN, the decoder, inverts this process by decoding the information summarized in the low-dimension feature maps to produce an output image (typically of the same size of the input image) that contains the requested analysis (in **Figure 3a** segmentation and identification of the objects at the input). The decoder upsamples the input of its convolutional layers after an element-by-element addition to the output of its previous layer of the corresponded downsampled layer of the encoding path that must have the same dimensions and the same number of features. By training the network, the feature maps produced by the last encoding layer must contain a synthesis of all the information needed by the CNN to reconstruct the output image. This means that they are produced by a complex and automatic feature extraction process that the network learned during the training, making the encoder–decoder architecture a powerful tool for analyzing raw data of any nature.

U-Net and ResNet are two of the most important and state-of-the-art models of CNNs for image segmentation, with many applications in biomedical image analysis. The U-net architecture^[55] consists of a contracting path and an expansive path. The contracting path has the typical structure of a convolutional network for downsampling. The expansive path consists of an upsampling to propagate context information to higher resolution layers, realized by transposed convolutional layers, and concatenation with a cropped feature map from the contracting path. As a consequence, the expansive path is more or less symmetric to the contracting path, and leads to a u-shaped architecture. Hence, the U-Net architecture can be seen as a sort of encoder–decoder architecture, but differs in the expansive path because of the presence of concatenations.

On the other hand, ResNet paved the way for the class of residual neural networks (ResNNs)^[56] models. These algorithms are a technological breakthrough that effectively allowed the deployment of DNNs with over 100 layers. The constituting element of a ResNN is the residual block, reported in **Figure 4**, which is

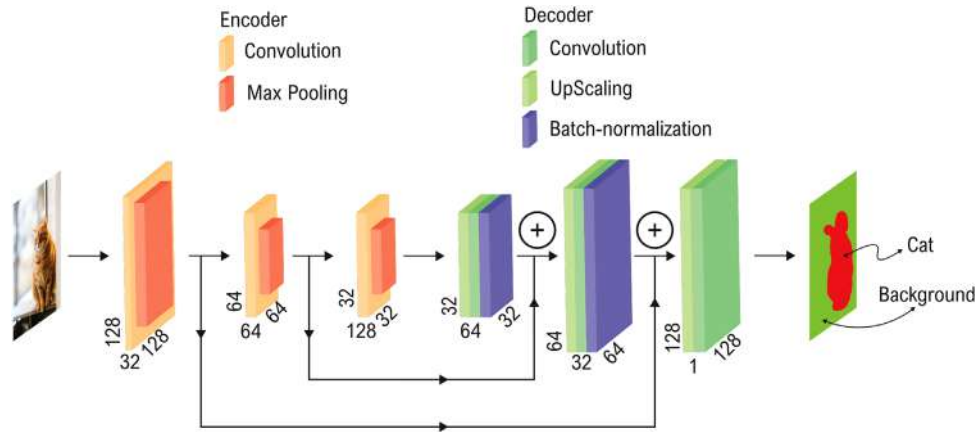


Figure 3. Example of an encoder–decoder architecture used for an image segmentation task on the data from ref. [54]. Encoder convolutional layers make use of pooling layers to reduce the input dimensionality, whereas decoder convolutional layers upsample the input feature maps and produce an output that contains the solution to the problem tackled. Moreover, the arrows point out a transfer of encoder information to the decoder layers by means of concatenations. Thus, first the model retrieves and encodes the hidden patterns in the input data, then decodes these informative feature maps to predict the final solution.

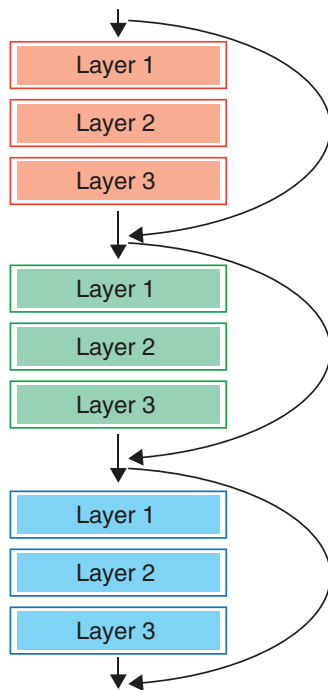


Figure 4. Illustration of a residual neural network. Each convolutional block of the network features a skip connection between its input and its output, resulting in the original information being passed on along with the processed one. Hence, deeper layers have access to the unprocessed informative content of shallower layers: the combination of shallow and deep feature maps is crucial to achieve high performances in extremely deep models, solving the problem of exploding and vanishing gradient during the training procedure.

characterized by a skip connection between its input and output. The presence of these connections and their combination creates a path where the input is propagated without passing through any convolutional layer, hence preserving its informative content that is more easily provided to deeper layers. The

combination of higher level feature maps with the deeper level ones proved to be a powerful tool to train extremely deep CNNs, making the performances of ResNN unrivalled in several image-related tasks such as computer vision. Most importantly, ResNNs can efficiently solve the problem of vanishing and exploding gradients. In fact, when extremely deep architectures are trained, the gradient computed by backpropagation tends to shrink to zero or become too big after several applications of the derivative chain rule. As a result, the network parameters fail to update efficiently. The skip connections typical of ResNet allow the flow of backpropagation directly on previous layers, which proved able to solve the gradient degradation issue. He and colleagues^[56] presented ResNet as a 34-convolutional layers network with a short-cut connection to each pair of filters, showing how such a model provides an effective gain in accuracy from an increased network depth.

3.2. Recurrent Neural Networks

RNNs are a specialized class of NNs that are used to deal with sequential data and time series.^[57,58] While the NN architectures presented above assume that inputs and outputs are independent of each other, RNNs allow for arbitrary neuron connectivity. This peculiarity causes the output of a RNN to be influenced not only by its current input, but also by the previous elements of the input sequence. Hence, the prior inputs operate as the hidden-state of a RNN. A graphical representation of this concept is reported in **Figure 5**, where a stream of outputs is produced sequentially on the basis of the analysis of an input time series. The RNN stores internally a “state” that encodes all the relevant information obtained from the previously examined elements of the input series. For this reason, RNNs are commonly said to have a “memory” and are hence among the most suitable NN architectures to study temporal data (e.g., sensor readings) and sequences (e.g., text bodies).

One of the most popular RNN architectures is the so-called long short-term memory (LSTM) network, a solution presented

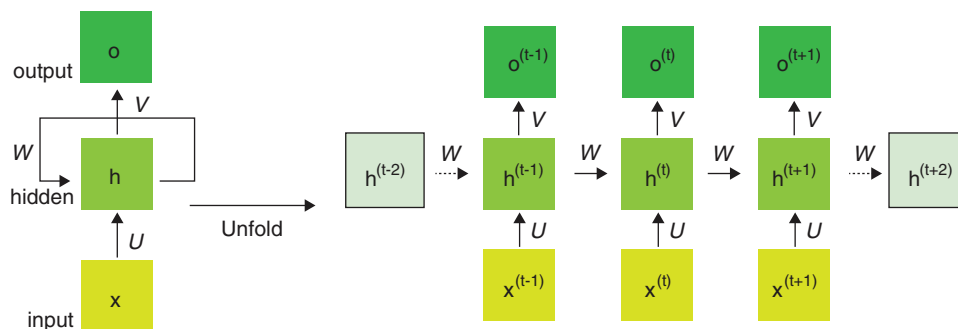


Figure 5. Illustration of a recurrent neural network architecture. The left-hand side diagram is the “rolled” visual of the RNN which represents the whole neural network: X is the input, h represents the hidden layers, W represents the connection between the hidden layers, O is the output. The right-hand side diagram visualizes the “unrolled” RNN with the individual layers, where the W connections ensure that the current output (e.g., $O^{(t)}$) is influenced not only by the current input ($X^{(t)}$) but also by all the previous samples in the sequence ($X^{(t-1)}, \dots$).

in ref. [59] to better capture long-term dependencies between the output and input values (e.g., a particularly slow dynamics may cause a control action to affect the evolution of a system only after a long time). LSTMs introduce the concept of “memory cells” to store and preserve portions of their internal state, demonstrating unprecedented capabilities on complex tasks such as speech recognition^[60] and epidemic forecasts.^[61]

4. Applications of AI to Spectroscopy

The application of AI in spectroscopy has proven powerful both to remove noise and undesired artifacts embedding the physically relevant spectral signals and to perform an accurate and efficient chemical analysis of spectral data. DL was also employed to overcome the instrumental calibration bias of spectrometers, which may strongly affect the reliability of the chemical interpretation. This section will first focus on AI-based spectral denoising applied to vibrational nonlinear spectroscopy and pump-probe ultrafast spectroscopy. It will then describe the main drawbacks of conventional spectral chemometric methods and DL applications aimed at overcoming such issues, providing end-to-end approaches able to surpass the accuracy of traditional data-processing while performing directly on raw data. In particular, we will review DL chemometric methods addressing both 1D spectra and 3D spectral images. Eventually, the final section will review an AI-driven approach to achieve calibration-agnostic spectrometers.

CNNs constitute the most frequently employed AI model in spectroscopy. It is worth pointing out that, despite 2D images constitute the main data type onto which CNNs have been applied and for which they have become popular among the scientific community, the reader will see in this section how such models operate on 1D single spectra as well. In fact, the ability of convolutional layers to extract hidden patterns from their matrix input is a valid statement regardless of the specific dimensionality of such a matrix. 1D kernels convolved on 1D inputs are a peculiar case of 2D kernels convolved on 2D inputs, so that the feature extraction task is carried out in a totally analogous manner and with comparable excellent performances with kernels sliding along a sole direction as the other one is unitary.

4.1. Spectral Denoising

4.1.1. Denoising of Spectral Profiles

Coherent anti-stokes Raman scattering (CARS) is one of the signals investigated within coherent Raman scattering^[62] (CRS) spectroscopy, together with stimulated Raman scattering (SRS). In CRS, two synchronized laser pulses are used to coherently drive and probe molecular oscillations in matter. The outcome of the measurement is a vibrational spectrum containing information about the chemical composition of the sample in the laser focus. The nonlinear third-order vibrational susceptibility responsible for CRS can be written as the superposition of two terms

$$\begin{aligned} \chi^{(3)}(\omega) &= \chi_R^{(3)}(\omega) + \chi_{NR}^{(3)}(\omega) \\ &= \text{Re} \left(\chi_R^{(3)}(\omega) \right) + i \text{Im} \left(\chi_R^{(3)}(\omega) \right) + \chi_{NR}^{(3)}(\omega). \end{aligned} \quad (11)$$

The resonant complex term $\chi_R^{(3)}$ contains the chemical information about the sample and can be modeled as a sum of Lorentzian peaks

$$\chi_R^{(3)}(\omega) = \sum_i \frac{A_i}{\Omega_i - \omega - i\Gamma_i} \quad (12)$$

where the sum runs over the vibrational resonances. The amplitude $A_i \propto \sigma_i C_i$ is proportional to the cross section (σ_i) and to the concentration of scatterers (C_i); Ω_i is the vibrational frequency, and Γ_i the linewidth. On the other hand, the nonresonant term $\chi_{NR}^{(3)}$, also known as nonresonant background (NRB), is generally assumed as a purely real contribution, that rules the nonlinear interaction of excitation beams with the sample and surrounding environment in a four-wave mixing process not mediated by any vibration. The relevant vibrational information is contained in $\text{Im}(\chi_R^{(3)}(\omega))$ and corresponds to the vibrational information obtained through spontaneous Raman. While SRS provides a signal directly proportional to $\text{Im}(\chi_R^{(3)}(\omega))$, in CARS the measured signal is $I_{\text{CARS}} \propto |\chi^{(3)}|^2 = |\chi_R^{(3)}(\omega)|^2 + \chi_{NR}^{(3)}(\omega)^2 + 2 \text{Re}(\chi_R^{(3)}(\omega)\chi_{NR}^{(3)}(\omega))$, producing a mixing of real and imaginary components, which introduces a relevant distortion of spectral features, especially when $\chi_R^{(3)} \ll \chi_{NR}^{(3)}$.

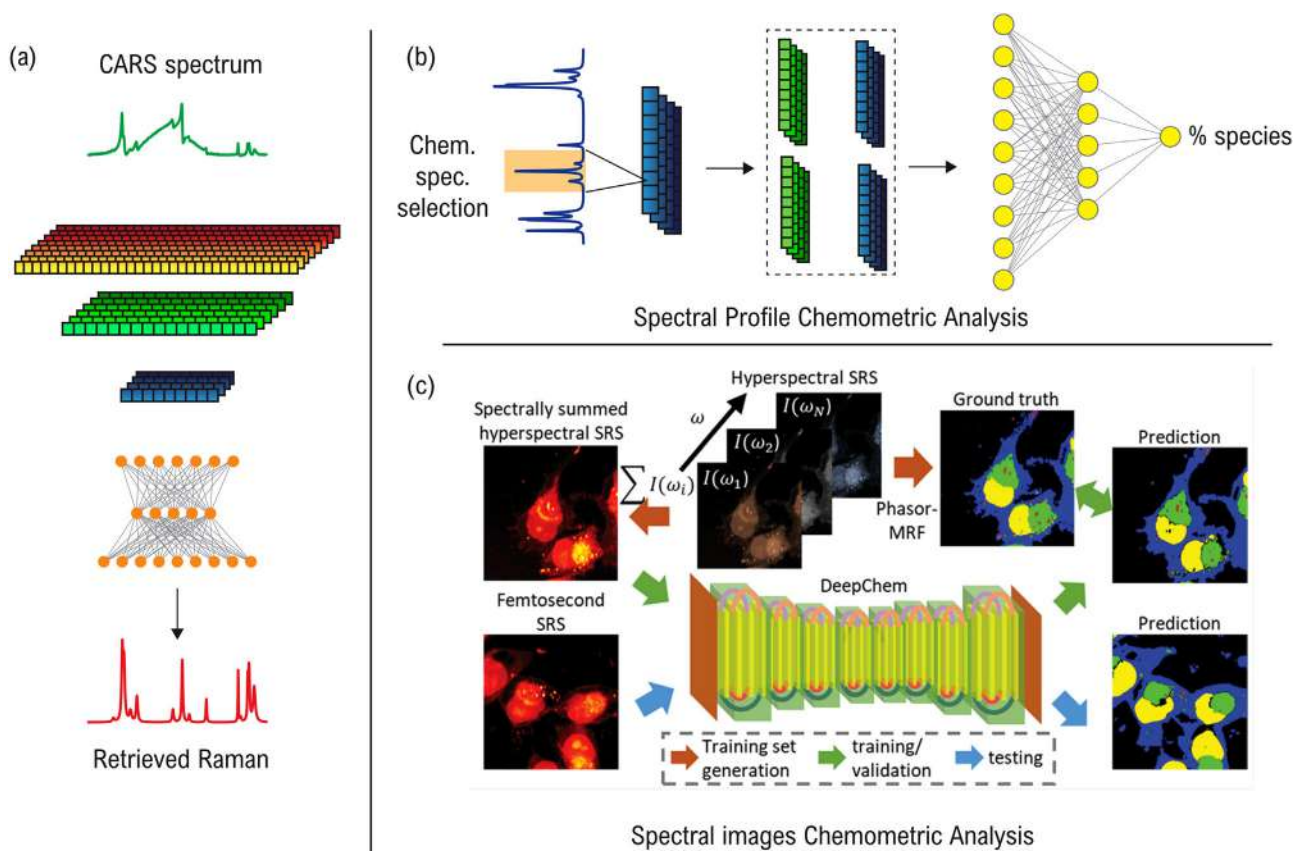


Figure 6. Examples of AI applications in spectroscopy. a) De-noising and removal of undesired artifacts of spectral profiles. A CNN model can remove the nonresonant background in CARS spectra, thus unveiling the Lorentzian peaks of the resonant Raman signal able to uniquely identify chemicals under investigation.^[63] b) Classification of chemical species from spectroscopy signals can be carried out via DL, by means of an end-to-end approach operating directly on raw spectral data, thus avoiding human-biased preprocessing.^[66] c) DL models can efficiently and accurately perform chemical segmentation of hyperspectral images. A CNN algorithm processes SRS images to generate a map of sub-cellular components based on the chemical information provided by spectral pixels. Reproduced with permission.^[67] Copyright 2020, American Chemical Society.

In this scenario, DNNs have been employed to solve the inverse problem related to Equation (11), that is, to retrieve $\text{Im}(\chi_R^{(3)}(\omega))$ from the measurement of I_{CARS} . Two different solutions have been proposed, tackling the problem of spectral de-noising from two different standpoints. In ref. [63], Valensise et al. employed a CNN, with the architecture inspired by classical LeNets,^[64] to leverage the richness of the input representations obtained through the convolutional layers and the flexibility of dense layers (Figure 6a). Convolutional layers perform very well in detecting peaks regardless of their position in the spectrum. In ref. [65], Houhou et al. used instead a LSTM architecture, looking at the spectrum as a sequence of values and at the line distortion as a pattern recurring in the data. we recall that LSTMs fall in the branch of recurrent NNs, meaning that unlike in standard feed-forward NNs, feedback connections are present among neurons. The training of the models is performed on synthetic datasets obtained through random sampling of the parameters in Equation (12) and the generation of smooth NRB traces, based on sigmoid and Gaussian functions. The simulated CARS spectra are used as inputs for the network, and the corresponding imaginary part as target variables. Both methods have been validated on CARS spectra experimentally measured on solvents.

In the context of nonlinear optical spectroscopy, pump–probe spectroscopy has been employed as a gold standard technique to study ultrafast electronic dynamics of material systems. It is typical to use high peak power laser sources to measure ultrafast pump–probe delay time traces, which may give rise to coherent artifacts under a broad range of experimental conditions. Among those, the cross-phase modulation (XPM) artifact causes strong signal distortions around time zero that hide a significant part of the dynamics of interest, causing loss of fundamental information to characterize the material under investigation. That is why the development of efficient methods to tackle the issue of XPM artifact removal urges, but the literature on the topic is somewhat restricted. In this framework, Bresci et al.^[68] reported an AI-driven model to retrieve pump–probe ultrafast electronic dynamics embedded in XPM artifacts. The CNN model, “XPMnet,” was trained on 10^5 inputs made up of data-augmented experimentally measured XPM artifacts superimposed on simulated exponential electronic cooling dynamics. Such electronic dynamics constituted the ideal ground truth to reconstruct. The model was able to operate with excellent figures of merit on both simulated and experimental pump–probe signals: $\text{MSE} = 5 \times 10^{-5}$, mean absolute percentage error=1% and $R^2 = 0.99$ (R^2 is defined as $\frac{\sum_i (f_i - \bar{y})^2}{\sum_i (y_i - \bar{y})^2}$,

where f_i is the model i th prediction, γ is the i th ground truth and $\bar{\gamma}$ is the overall mean ground truth). The experimental validation of the model on indium tin oxide (ITO), a key semiconductor for the development of infrared plasmonic devices, showed that the CNN predicted electronic dynamics in perfect agreement with expected outcomes in terms of material-specific time constants. Since “XPMnet” operates with high accuracy and an execution time as short as 30 ms, the AI model could be integrated in real time in pump–probe setups to increase the amount of information one can obtain from ultrafast spectroscopy measurements.

4.1.2. Denoising of Hyperspectral Images

SRS is a CRS technique widely employed to perform chemically selective imaging.^[62] However, low signal-to-noise ratio (SNR) and light scattering in dense samples (e.g., biological tissues) limit biomedical applications of SRS. Low SNR issues can be typically mitigated through longer acquisition times, that are nevertheless not applicable when a high imaging speed is required or can induce sample damage. In ref. [69] a CNN is trained to perform denoising of SRS images. The model is trained through pairs of images collected at low (input) and high (target) SNR, obtained varying the laser power. The trained model is demonstrated to perform well even on images whose SNR is limited by other factors, such as imaging depth and zoom.

Lin et al.^[70] presented a DL model for SNR improvement of hyperspectral SRS images in the fingerprint region. They developed an encoder–decoder ResNN which exploited upsampling and skip connections to achieve a high performance with a low amount of training data (<20 spectroscopic images of 200×200 pixels with 128 spectral channels). The input–output pairs were chosen as experimental SRS hyperspectral cubes with low pixel integration time (low SNR) and high pixel integration time (high SNR), respectively. The model improved the SNR of hyperspectral SRS images by one order of magnitude, allowing an acquisition speed of 20 μ s per pixel. The proposed ResNN features spatial and spectral parallel filtering in convolutional layers able to maintain information about correlations both in the spectral and in the spatial domain. The AI-driven denoiser was able to outperform conventional unsupervised algorithms for image restoration (e.g., block-matching 4D filtering). When compared with U-net, the model was able to denoise images with better detail preservation avoiding the introduction of artifacts.

Among the most widely used techniques in biomedicine, materials science and metrology, one can find spectral interferometry. Measured single-shot interference patterns are employed to retrieve the phase and amplitude of the optical electric field, typically by means of the Hilbert transform. Because of the deterministic nature of such an approach, this method is not robust with respect to nonlinear optical distortions, shot noise and is limited by the sampling rate in detection. Pu et al.^[71] demonstrated how deep learning may be a valid solution to perform spectral interferometry measurements with high accuracy also in case of such nonideal experimental conditions. They proposed a five-layer fully-connected NN trained on 6000 experimental spectral interferograms, as measured by a time stretch single-shot spectrometer. Interferograms are derived from electric fields spectrally modulated with known causal profiles of phase and am-

plitude that constitute the model ground truth. Indeed, the regression model outputs a vector that concatenates phase and amplitude spectra of the complex electric field. Remarkably, the algorithm, which operates on single shot measurements, outperforms the Hilbert transform technique as a time-averaged result over multiple frames by a grating-based spectrometer, in terms of both prediction accuracy and time-effectiveness. In fact, the Hilbert transform suffers from distortions induced by optical nonlinearities, mainly self-phase modulation, taking place in the dispersive media used for time stretching. The amplitude and phase RMSE for the AI-driven predictions are 0.03 and 0.04, respectively. On the other hand, the traditional Hilbert transforms features a poorer amplitude and phase RMSE of 0.16 and 1.1, respectively.

4.2. Spectral Chemometrics

Vibrational spectroscopy, such as infrared (IR) and Raman techniques, is able to extract chemically specific information with high speed, accuracy and non-invasiveness. However, the derivation of quantitative chemical data relies on the employment of chemometrics, a data-driven mathematical and statistical approach to extract the chemically relevant information from spectral measurements of light-matter interactions. One of the most popular traditional chemometric methods is partial least squares (PLS) regression,^[72] which is able to unveil the embedded chemically relevant linear relationships in highly multivariate spectroscopic data. Principal component analysis (PCA)^[73,74] is another commonly employed analytical method in chemometrics: it performs a change of basis on the data, projecting them onto the principal components space. SVMs^[75,76] constitute another class of powerful methods. In particular, such models have been widely employed due to their high performance in classification tasks. However, all the aforementioned traditional chemometric methods require spectral data preprocessing in most cases in order to achieve a robust and accurate result (e.g., baseline correction, scatter correction, signal smoothing and scaling^[77]). Data preprocessing, besides requiring an a priori knowledge of the sample and being expensive in terms of time, resources and computing, may also cover patterns of interest and generate artifacts, thus compromising the reliability of the chemical classification and quantification.^[78] A valid solution to the drawbacks associated with data preprocessing is the use of AI:DL algorithms, such as CNNs, act as end-to-end approaches able to operate directly on raw spectral data, extracting more and more complex patterns of interest as the input traverses convolutional layers.

4.2.1. Chemometrics of Spectral Profiles

In the context of AI-driven spectral chemometrics, Zhang et al.^[66] proposed “DeepSpectra” (Figure 6b), an end-to-end CNN integrating data pre-processing and analysis in a single-stage architecture. They compared the performance of “DeepSpectra” with popular multivariate calibration methods (i.e., PLS, PCA, and SVM) on both raw and preprocessed visible and near-infrared (NIR) spectra of pharmaceutical tablets, wheat, soil, and corn. The CNN input consisted in a raw spectrum, whereas the

output was the single object character to be predicted. “DeepSpectra” featured a peculiar structure in the convolutional layers, the so-called inception module: after a first classical convolution, the second and the third layers combined parallel kernels and pooling steps. Increasing the convolutional width in such a way granted a higher adaptability to diverse spectral patterns in the input. On the other hand, increasing the convolutional depth improved feature extraction, even in the case of extremely hidden nonlinear patterns. The final fully connected layers computed the output prediction in the form of a single node. “DeepSpectra” on raw data outperformed PCA, PLS, and SVM methods on pre-processed data, featuring a root-mean-square error (RMSE) in the range $< 0.1 \div 0.3$ in any case studied. Indeed, predictions on preprocessed data exhibited a RMSE in the range $0.1 \div 1.4$ with SVM, $0.3 \div 1.1$ with PCA and $0.1 \div 0.6$ with PLS algorithms. This work demonstrated the possibility to achieve a more accurate quantitative analysis via DL rather than classical calibration approaches, with the advantages of acting directly on raw data, reducing the computational and temporal cost and avoiding the human bias in the choice of pre-processing methods.

Liu et al.^[79] developed a CNN to classify chemical species from Raman spectroscopy data, exploiting the RRUFF mineral dataset.^[80] Their goal was to combine preprocessing, feature extraction and classification in a single model, in order to avoid any manual tuning and to achieve a higher accuracy with respect to the common chemometric pipelines for Raman spectra (i.e., cosmic ray removal, baseline correction, smoothing, PCA, SVM-based final classification). The proposed CNN takes a 1D experimental spontaneous Raman spectrum in wavenumbers as an input. The output consisted in a fully connected layer with a number of nodes equivalent to the number of chemical species into which data had to be classified: they tested either 512 or 1671 different minerals multi-classifications. The model architecture was a variant of LeNet, a popular CNN for classification tasks: it featured three pyramid-shaped convolutional layers and two fully connected layers. The algorithm achieved an accuracy of 93.5% on raw spectral data, outperforming by a large margin the higher accuracy of the SVM algorithm on raw data (52.2%). Interestingly, while the SVM increased its performance up to an accuracy of 82.1% with preprocessing (e.g., asymmetric least squares baseline correction), the CNN performance accuracy dropped by 0.5–2.5% when preprocessing was applied to its inputs. This fact proved that the DL-based model could retrieve discriminant information for an accurate prediction by managing baseline interference rather than working on baseline-corrected data. A similar problem was tackled by Ho et al.^[81] by means of a ResNN model applied on single spontaneous Raman spectra of infectious bacteria with poor SNR. The output of the algorithm is a probability distribution over 30 different classes of bacterial species, further grouped according to the recommended treatment, which is indeed the ultimate goal in the fight against infectious diseases. The model features six skip connections and unlike previous works on similar architectures replaces pooling layers with strided convolutions. This particular choice led to a better spatial localization of Raman peaks, overall improving the model performance. The model was trained on a total amount of 6×10^4 Raman spectra, 2×10^3 for each one of the 30 infectious bacterial species considered in the study. For 1 s measurements, featuring a SNR ratio of 4.1, the ResNN model achieves an accuracy of 82%

on the 30-class task, which increases with the input SNR. Interestingly, when considering the further treatment classification, the model reaches an accuracy of 97%. The authors employed logistic regression and SVM classification methods for benchmark, both of which were outperformed by the ResNN-driven classifier. In fact, in the 30-class task and in the treatment choice task the accuracy of logistic regression was 75.5% and 93.3%, respectively. Similarly, SVM achieved an accuracy of 74.9% and of 92.2%, respectively.

Combining CNNs with ensemble learning, as reported for the first time by Yuanyuan et al.,^[82] turned out to be a valid strategy to outperform single CNN chemometric analysis. Ensemble learning consists in training and testing several models on randomly sampled subsets, then aggregating each individual prediction into a final output by means of an averaging method. The purpose of such an approach is to grant the stability and the robustness of the quantitative model: each sub-model analyses the local distribution of data so that their combination is an actual depiction of the blueprint of the dataset. The proposed ensemble CNN (ECNN) consisted in an aggregation of 10 CNN models, whose predictions were combined into a final output by weighted average. They trained, validated and tested the architecture on three raw IR spectroscopy datasets of corn, gasoline and mixed gases. The ECNN input consisted in 1D experimental IR spectra, while the ECNN output layer nodes provided the predicted content of a variable number of chemical components, according to the experimental dataset (e.g., moisture, oil, starch, and protein content for the corn dataset). The coefficient of determination R^2 for the ECNN model classification performance was higher with statistical significance than a single CNN model or traditional PLS (different methods were run 50 times to provide statistical comparisons). In particular, the variance of the R^2 parameter was much smaller for the ECNN than the compared methods, which implied that the novel ECNN approach actually met the goal of increasing the model stability.

Even though DL-based chemometrics may be considered a preferable but optional method, it is actually essential when spectral signals are overwhelmed by optical background effects and high noise, which prevents data interpretation via traditional methods. In this context, surface-enhanced Raman scattering (SERS) is one of the most promising tools for highly sensitive bio-imaging: it is able to amplify and detect low-density molecular vibrational signals otherwise poorly resolvable. The main drawback of SERS is the challenging interpretation of the spectral data because of strong interference effects, as pointed out by Guselnikova et al.^[83] in their work on DL-assisted SERS detection of minor UV-induced DNA damage. In particular, they proposed a CNN taking as an input the SERS spectra of UV-damaged oligonucleotides grafted to an Au plasmonic surface, whereas the CNN output consisted of a classification into four different categories of damage related to different UV-exposure durations. The relevant result of this CNN model was the ability to classify DNA damage with a 98% accuracy from SERS spectra measurements avoiding optimization procedures, such as baseline correction, optimal sample area, optimal laser intensity and acquisition time.

Convolutional layers of CNNs applied to spectroscopy signals were demonstrated to be able to find optimal data manipulation and feature extraction methods by tuning kernel variables automatically: kernels serve as spectral pre-processors.^[84–86] The

process through which CNN kernels perform feature selection for chemometrics is not a total “black box:” it can be qualitatively understood and recognized by making the kernels output explicit, as reported by Bjerrum et al.^[87] in a study on CNN-based chemometrics of NIR spectroscopic data of drug tablets. They compared the five most active kernels in the first two convolutional layers with their input spectrum. In the first convolutional layer, the most active areas in four out of five kernels were the ones with higher input spectral intensity, with kernel activation intensity dependent on the slope of the spectral peaks of the input. The fifth kernel of layer one was instead activated in correspondence of the lower intensity areas of the input spectrum, thus highlighting orthogonal features with respect to the other filters. Hence, the first convolutional layer was able to apply threshold and derivative activation, which are well-known spectral signal processing techniques (e.g., Savitsky–Golay filtering). Similarly, the analysis of orthogonal components is also a key feature of popular chemometric processing algorithms, such as PLS and PCA. The next convolutional layer acted on the output of the first layer, thus achieving a higher level of abstraction and a more complex feature selection: the most active kernel areas were related to the peak surroundings of the input spectrum. This indicates that the second convolutional layer performed spectral region and variable selection, another well-known technique for spectral analysis.

Bjerrum and colleagues thus proved how CNNs applied on spectral signals are able to automatically mimic and optimize popular chemometric pre-processing methods without the need for human decision in the process. They also reported an interesting and effective procedure to increase the robustness of CNNs employed in chemometrics: spectral DA. It consists in generating expected variations of the existing training spectra and exploiting them for a more diverse training, thus reducing the risk of overfitting the model and increasing its accuracy on unseen spectral instances. They applied this novel spectral DA on NIR spectroscopic data used to train a CNN model for drug content prediction in tablets. The model received a 1D spectrum as input and predicted the drug content value via a single-node output. They presented three valid spectral augmentation techniques: random offsets variation of ± 0.1 times the training set standard deviation, random slope change in the range $0.95\div 1.05$ and random intensity multiplication by 1 ± 0.1 times the training set standard deviation. Such changes were applied nine times for every instance in the training set, which resulted into a tenfold increase of the training and validation set dimension. The performance of the CNN model with and without spectral DA was investigated in terms of RMSE: the standard dataset resulted in a training RMSE of 3.02 and a testing RMSE of 4.01, whereas the DA dataset achieved a training RMSE of 2.21 and a testing RMSE of 3.97. Along with a decrease in the RMSE value, the DA procedure allowed to obtain smoothed loss function decreasing curves in both training and testing over the course of the epochs.

It is worth noticing that chemometric analysis accuracy is strictly dependent on the instrumental calibration of spectrometers employed for spectral measurements. Indeed, spectrometers are commonly used in spectroscopy to obtain quantitative data from light–matter interactions. However, they suffer from a time-varying calibration and absolute calibrations may not suit a variety of devices. Chatziadakis et al.^[88] proposed an AI-based

method to extract chemometric data from electron energy loss spectra, which could be readily applied to spectral profiles from optical spectroscopy measurements as well. They validated the model on three different electronic environments of manganese, addressing only the relative position of absorption and emission peaks rather than their absolute one, which may be strongly affected by calibration. As a matter of fact, only the spectral shape was used by the algorithm to identify the chemicals: the peaks in the training examples were shifted as a DA procedure, in order to achieve spectral translation-invariance by the AI classifier. They employed this DA dataset to train and test three different NN architectures: a densely connected NN featuring 11,000 weights, a fully convolutional NN without dense layers featuring 650 weights and a convolutional feature extractor followed by a dense NN featuring 1100 weights. The algorithms took as input the spectral data and produced a classification into three different classes, implemented via a final three-nodes output layer. The proposed fully CNN with the lower number of weights proved to be the sole agnostic architecture with respect to the translation of spectral peaks.

4.2.2. Chemometrics of Hyperspectral Images

AI-based chemometrics applications involve also higher dimensional data: whereas spectra are regarded as 1D, 3D hyperspectral images with an arbitrary number of spectral channels can be employed as well as inputs for DL models. Hence, an hyperspectral 3D image of this fashion consists in 1D spectra for each pixel of a 2D image of the sample.

Krauß et al.^[89] proposed a hierarchical CNN for a highly accurate spectral and spatial chemical analysis and classification of Raman spectroscopic images of urine cells for urothelial bladder cancer screening. The CNN model took as input a spectral image and processed it to produce a cell classification into tumorous or not, achieved via a final output layer featuring two nodes associated with the diagnosis result certainty. This work proved how sequential convolutional layers applied to 3D inputs were able to cope not only with spectral but also with morphological information by hierarchical merging: kernel by kernel, a broader area of spectral pixels was integrated so that chemical features inherent to the Raman spectrum were associated with the corresponding spatial region. They also demonstrated that hierarchical CNNs on spectral images may be efficiently applied on a reduced number of spectral points, thus down-sizing the 3D input and reducing the time and computational cost. The max-relevance min-redundancy (MRMR) algorithm was employed to select wavelengths relevant for the classification from the Raman spectra. As a result, the hierarchical CNN algorithm outperformed with an accuracy of 0.99 both conventional pixel-by-pixel full-spectra classifiers (0.96 accuracy) and conventional morphological features extraction methods (0.89 accuracy).

In the same context, Zhang et al.^[67] reported an example of DL-based chemical imaging from high-speed femtosecond SRS. They proposed a CNN, named DeepChem, which was able to produce as output a subcellular organelle 2D map with chemical selectivity on four components (i.e., nuclei, cytoplasm, lipid droplets, and endoplasmic reticulum) given a single-frame femtosecond SRS image (Figure 6c). Indeed, hyperspectral SRS

measurements with chirped pulses require a longer acquisition time than hyperspectral single-shot SRS measurements with nonchirped pulses, but the latter modality is associated with low spectral resolution and deteriorated SNR. DeepChem served as a crucial tool to tackle the trade-off between spectral and chemical selectivity, SNR and measurement speed. The CNN training was done employing as inputs spectrally summed chirped-pulses SRS hyperspectral images collected via spectral scanning by a motorized translational stage (acquisition time: 110 s), whereas the associated ground truths consisted in chemical maps obtained by hyperspectral image segmentation by Phasor–Markov Random Field. Subsequently, DeepChem was tested on single-shot femtosecond SRS images obtained via nonchirped femtosecond pulses (acquisition time: 1–2 s) and lacking of spectral resolution: the CNN was able to predict the chemical map with a F1 score (i.e., the harmonic mean of precision, which is the number of true positives over the number of all positive results, and recall, which is the number of true positives over the number of all samples that should have been identified as positives) of a 0.787 for nuclei, 0.645 for lipid droplets, 0.805 for endoplasmic reticulum, and 0.789 for cytoplasm, which could be considered as a promising result compared to conventional chemical segmentation methods of fluorescence images (about 0.7 F1 score).

5. Application of AI to Optical Wavefront Shaping

Light that propagates in media experiences a distortion of the wavefront due to the inhomogeneous profile of the refractive index across the material. The consequent random interference produces the so called optical speckle, a pattern composed by dark and bright spots. The process of light propagation can be analytically described by a transmission matrix T , such that the output electric field \mathbf{y} is linked to the input one \mathbf{x} by

$$\mathbf{y} = T\mathbf{x}. \quad (13)$$

The transmission matrix T describes the interaction between the medium and the optical field and it is responsible for phase distortions leading to speckle formation. Optical wavefront shaping^[90] encompasses a series of techniques that are employed to compensate for this. Generally, a spatial light modulator (SLM) is used to modify the wavefront of the incoming light and achieve optical focusing or imaging through the scattering medium. In order to optimize light control by the SLM, several approaches may be employed, as optical phase conjugation, iterative optimization or transmission matrix measurement.^[91] These conventional methods suffer from a time-consuming optimization procedure, whose complexity scales linearly with the number of pixels of the SLM of the order of 10^6 and, among them, the transmission matrix method is too sensitive to noise and sample perturbations. Indeed, it relies on the assumption that optical processes are linear and can be modeled as a single matrix, which is not always true in noisy environments. Vellekoop and Mosk employed for the first time optical wavefront shaping to focus light through^[92] and inside scattering objects^[93] with an algorithm that constructs the inverse diffusion wavefront exploiting the linearity of the scattering process. A complete survey of all the techniques employed for optical wavefront shaping is beyond

the aim of this work. Nevertheless, we refer the interested readers to the work of Vellekoop^[94] that focuses on feedback-based wavefront shaping approaches and on some of the fundamental properties of these techniques as well as to the work of Mosk et al.,^[95] who reviewed the field of optical phase conjugation in disordered media and novel wave modalities such as ultrasound and radio waves. Finally, the review of Horstmeyer et al.^[96] focuses on guidestar-assisted techniques for controlling light inside in vivo tissue and provides a description of some biological applications of such approaches.

5.1. Computational Imaging

A field related to wavefront shaping is computational imaging (CI).^[97] The aim of CI is to overcome the limitations of imaging systems, mainly due to the physical measurement and the medium through which light travels before reaching the detector. Hence, after the computation, CI provides information which is not readily available from intensity images, such as tomography and quantitative phase retrieval. In this context, the scattering problem is stated as the Tikhonov–Wiener optimization function

$$\hat{f} = \operatorname{argmin}_f \|Hf - g\|^2 + \alpha\Phi(f) \quad (14)$$

where $\|\cdot\|$ denotes the L^2 norm, \hat{f} is the estimate of the target image f , and g is the measured intensity; H is the scattering operator and Φ is used to encode the a priori knowledge about the correlation patterns that are present in the scene being imaged, that may help in the recovery of the original object, with α a parameter ruling its weight. A computational imaging scheme with the ML engine is schematically shown in Figure 7.

In these contexts AI, and in particular ANNs, can be conveniently exploited thanks to their capability to approximate complex functions. Namely, by looking at input–output pairs coming from the transmission or imaging setup, a NN can be trained to approximate the operators T , H , and Φ ruling light propagation. The first demonstration of solution of imaging inverse problem in presence of scatterers by means of ML tools was given by Horisaki et al.^[98] In this work, a SVM was trained to well approximate the inverse relation from speckle pattern to object images. The algorithm was trained on a dataset of human faces and showed major improvements with respect to previous methods, such as TM measurement techniques. Later, Lyu et al.^[99] employed a DNN to approximate the mapping between the image displayed on the SLM and the corresponding speckle pattern recorded by a camera after a scattering slab. The proposed NN was a fully-connected DNN with five hidden layers, with ReLU activation function. Four thousands images from the MNIST handwritten digits dataset^[100] were used as training set for the experiment. The digit (ground truth) was displayed by the SLM, and the corresponding speckle pattern used as network input. In ref. [101] the same task was accomplished by means of a CNN built with the encoder–decoder architecture. Interestingly, beside the conventional mean absolute error loss function, also the negative Pearson correlation coefficient (NPCC) was investigated as loss function. NPCC is defined as $\text{NPCC} = -\text{cov}(Y, G)/\sigma_Y\sigma_G$, where Y and G are respectively the NN output and the ground truth, $\text{cov}(\cdot)$ is the covariance, σ_Y and σ_G the standard deviation

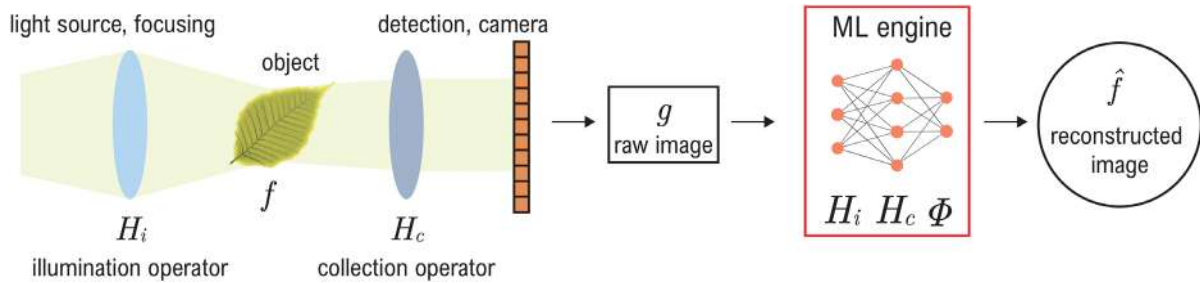


Figure 7. General computational imaging system. An illumination system, here represented by the illumination operator H_i , guides light onto the object f . After the object, light is shaped by an optical encoding through a collection system, represented by the collection operator H_c , that delivers light onto a camera which registers the image intensity pattern g . The acquired raw image is sent to the ML engine to obtain the reconstructed image \hat{f} . The ML engine generally includes a multilayered architecture and is informed on the physics of the illumination and collection optics. The three components H_i , H_c and the prior knowledge Φ are incorporated in the ML engine either explicitly as approximants or implicitly through training with examples. Adapted with permission.^[97] Copyright 2019, Optical Society of America.

of Y and G . Training the network using this loss function, they demonstrated that the DNN showed superior performances in case of sparse objects and strong scatterers. The network was trained and tested exploiting several datasets: Faces-LFW^[102] and Faces-ATT^[103] (i.e., dataset of web face images and photographed faces, respectively, each labeled with the correspondent identity), ImageNet^[104] (i.e., dataset of images of objects with correspondent hand-annotated object classification), MNIST^[100] and CIFAR^[105] (i.e., color images of 10 object classes). In ref. [106] a network made of convolutional layers followed by dense layers was used to tackle the same problem.

Finally, Li et al.,^[107] differently from previous works, trained the CNN not to reconstruct the transmission matrix of a single scattering medium but to learn a “one-to-all” mapping using multiple scattering media for the training. They showed that the CNN model trained on a few diffusers can sufficiently support the statistical information of all diffusers having the same mean characteristics, thus performing well on speckle patterns from unseen diffusers for high-quality object predictions.

As already mentioned above, these reviewed DL-based methods use arbitrary images to train the DNNs for reconstruction tasks. Moreover, they do not require to measure the complex field amplitude as they just need input–output field intensities for the training. Hence, compared to traditional TM methods, they show higher accuracy and reduced optical setup complexity. Finally, the proposed architectures exhibit promising results also when tested on images of objects not included into the training set.

A seminal work for the application of AI to light control was performed by Horisaki et al., exploiting the same approach described above. In ref. [108] the SVM was trained to focus light onto a 5×5 pixel grid through a SLM. Later, Turpin et al.^[109] demonstrated light control through scattering materials by means of a single-layer NN and a CNN made of three convolutional layers and three max-pooling layers. Pairs of binary illumination patterns were generated via a SLM, and the corresponding speckle pattern was recorded through a CCD. Given this dataset, the NN was trained to infer the relationship between the scattered light distribution and the illumination pattern. Once this was established, a desired output pattern was input in the NN and the corresponding illumination pattern, to be displayed on the SLM, was obtained. Good agreement between the desired pattern and the one actually displayed on the CCD was demonstrated for sev-

eral scattering media: glass diffusers, multi-modal fibers and paper. Moreover, exploiting other NNs, the authors demonstrated the capability of controlling transmitted light through the light portion that is reflected by the scatterer. This is accomplished exploiting two NNs: one approximating the relationship between transmitted and reflected light by the scatterer, the other inferring the relationship between the reflected light and the illumination pattern displayed on the SLM.

Recently, Luo et al.^[110] combined a NN with a genetic algorithm (GA).^[111,112] GAs are metaheuristic algorithms used to solve real-life complex problems belonging to different fields such as economics, engineering, politics, and management. They mimic the Darwinian theory of survival of the fittest in nature using as basic elements chromosome representation, fitness selection and biological-inspired operators such as selection, mutation, and crossover. While NNs are not guaranteed to reach a global optimum of the parameter set for approximating the target function, GA guarantees the convergence to the global optimum, provided that the starting point is in proximity of the global optimum. In ref. [110], a GA is used to improve the illumination pattern provided by the CNN. The CNN takes as input the SLM pattern, the corresponding speckle and the desired focus pattern. The latter two inputs are processed by two independent set of convolutional layers, while the SLM pattern is directly passed to a fully connected layer. Finally, a series of fully connected layers is used to concatenate all the inputs and produce the SLM pattern to obtain the desired light arrangement.

In ref. [113] the inverse problem of CI (see Equation (14)) is formulated in terms of a purely phase-modulated field. Let $E(x, y, z = 0) = e^{i\phi(x,y)}$ represent the optical field with unitary amplitude and phase modulated by the unknown object, which introduced a phase $\phi(x, y)$ at the lateral Cartesian coordinates x, y at position $z = 0$. After propagating a distance z , the measured intensity image is $I(x, y) = |E(x, y, z)|^2 = H\phi(x, y)$, where H is the forward operator that relates the phase $\phi(x, y)$ at the origin $z = 0$ to the intensity image at distance z . To retrieve the optical phase it is then required to solve the inverse problem

$$\hat{\phi}(x, y) = H^{\text{inv}} I(x, y) \quad (15)$$

with $\hat{\phi}$ denoting the estimate of the phase rather than the exact solution. The NN used in this experiment is based on a ResNet

architecture. A SLM is used to generate phase images, by randomly sampling images from the Faces-LFW^[102] or the ImageNet database.^[104] The corresponding raw intensity was detected by a CMOS camera. The DNN was first trained using images of faces and then images of natural objects. The two trained models were able to reconstruct distinct images, such as handwritten digits, characters from different languages and images from a disjoint natural image dataset. Both trained networks reached accurate results, suggesting that they have actually learned a generalizable model approximating the inverse operator H^{inv} .

5.2. Digital Holography

Digital holography (DH) is a CI technique that allows retrieving the phase of a light field, exploiting the presence of a reference beam that interferes with the field under investigation.^[114] The presence of the reference field produces artifacts that require in turn additional intensity information for proper phase recovery, that are generally obtained by scanning physical degrees of freedom of the experiments (e.g., sample-to-sensor distance) or varying the properties (wavelength, phasefront) of the reference beam. In ref. [115], Rivenson et al. proposed a solution for phase recovery and holographic image reconstruction using DNNs. After proper training of the network with 150 training instances, they demonstrated that a single measured hologram is enough to extract the phase information. The employed neural network is a deep CNN with residual blocks. The network input is composed by a single hologram image, divided into amplitude and phase contribution, while the corresponding ground truth is obtained applying the state-of-the-art algorithm in DH, namely the multi-height phase recovery method.

A CNN, inspired by the U-Net, was also exploited by Wu et al.^[116] to automatically perform autofocusing and phase recovery to retrieve the 3D information from a single hologram image. This approach allows to extend the depth of field (DOF) and reconstruction speed in holographic imaging. The CNN is trained by using pairs of defocused back-propagated holograms and their corresponding in-focus phase-recovered images, as ground truth. Once trained, given a single back-propagated hologram, the CNN is able to reconstruct the in-focus image of the sample over a DOF of $\approx 90\mu\text{m}$. This DL method for DOF extension is non-iterative and significantly improves the algorithm time complexity of holographic image reconstruction from $O(n \cdot m)$ to $O(1)$, where n refers to the number of individual object points or particles within the sample volume, and m represents the focusing search space within which each object point or particle needs to be individually focused.

As this section has pointed out, artificial intelligence can be used to assist DH to perform automatic auto-focusing and image reconstruction. Nevertheless, DH can be a useful tool to provide input–output pairs to train NNs. Next section will extensively describe this last aspect.

5.3. Multi-Modal Fibers: Control of Non-Linear Wavefront Distorsions

Optical wavefront shaping is also strictly related to MMFs, which support tens of thousands of optical modes and can deliver spatial

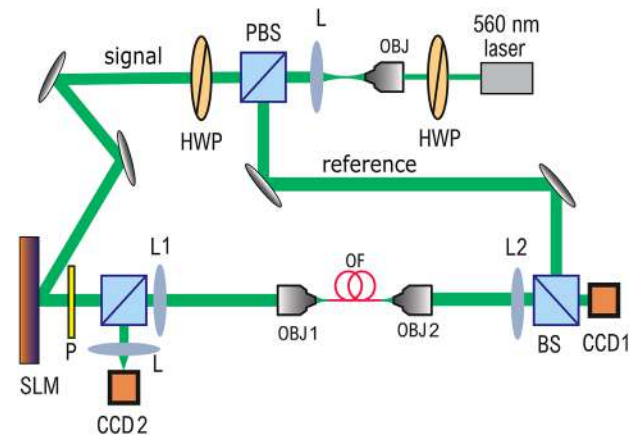


Figure 8. Scheme of a typical holographic setup, commonly employed to collect input–output pairs to train DNNs. The light of a 560 nm laser is split into two beams at the polarizing beam splitter (PBS): signal and reference beams. The laser beam in the signal arm is modulated by a spatial light modulator (SLM). Then, the SLM pattern is imaged onto the proximal facet by means of a 4f-system, constituted by the lens L1 and the objective OBJ1. A second 4f-system, constituted by the objective OBJ2 and the lens L2, is employed at the distal end of the fiber to image the output speckle pattern at CCD1. CCD2 measures the SLM output pattern. The system can also record the corresponding digital hologram which is formed when the reference beam interferes with the speckle pattern on the CCD1. Adapted with permission under the terms of a CC-BY-NC-ND license.^[139] Copyright 2019, The Authors, published by Elsevier Inc. HWP: half waveplate; OBJ: objective; L: lens; P: polarizer; OF: optical fiber; BS: beam splitter.

information. Coherent light that propagates into MMFs, which behave as scattering media, experiences distortion in the wavefront shapes of the input fields due to modal dispersion. Since 1967, MMFs have been used for imaging. The very first application of MMFs in imaging was done by Spitz and Wertz,^[117] who phase conjugated the light transmitted through a MMF and obtained a recognizable image back at the input. After them, several groups^[118–125] exploited MMF for imaging using optical phase conjugation as the mechanism to undo the modal distortion introduced by the MMF. Other research groups have tried to compensate for the modal dispersion in MMF using digital iterative algorithms^[126–130] or interferometric methods^[131–135]

Thanks to the development of DH,^[122,123,136–138] imaging in MMF has changed. Indeed, DH provides several input–output pairs that can be used to characterize the transmission matrix of a MMF. Nevertheless, the experimental setup (see **Figure 8**) used for holography to generate the input–output pairs requires a calibration step in advance, since it involves different optical elements. Once the transmission matrix is reconstructed from the input–output pairs, it is used to establish the input wavefront shape at the frontal end of the fiber in order to obtain a desired target image at the distal end of the fiber. The retrieved transmission matrix is also used to interpret the light at the distal end of the fiber knowing a given SLM input pattern. The problem is that the time at which the calibration has been performed and the time when the retrieved matrix is used are different and this may lead to a poor reliability of the performance of the system. In particular, the bending of the fiber is one of the most detrimental problems for applications to imaging. Another drawback of DH is the complexity of the experimental setup used to acquire the

input–output pairs. Indeed, the holography system requires an external reference beam brought to the output of the fiber to generate an interference pattern from which the complex optical field (amplitude and phase) can be extracted. Although some studies have shown that the reference beam can also be sent through the same MMF,^[135] multiple quadrature phase measurements must be done to extract the phase. In order to assess the fidelity of the transmission matrix approach, in the system constituted by MMF and SLM, measurements with amplitude modulation only, phase modulation only and combination of both have been executed. The main problem is that the MMF is treated as a black box which cannot be modeled by a simple physical system, that is the reason why to characterize it one needs a number of orthogonal input modes at least equal to the degrees of freedom of the system.

In the past few years, the development of AI has led to the growing usage of DL approaches to solve highly nonlinear problems.^[140] Indeed, DL allows one to reconstruct the transmission matrix feeding the network with a number of non-necessarily orthogonal inputs. Since DL deals with nonlinear systems, NNs are suitable to retrieve the transmission properties of the MMFs reconstructing the nonlinear relationships between inputs and outputs. Hence, to retrieve what input phase (or amplitude) has generated an intensity pattern, one can refer to intensity measurements only, rather than holographic measurements, thus simplifying the experimental setup. It is relevant to notice that when a phase term $\phi(x, y)$ is introduced to the input complex optical field, the system has two nonlinearities: the first is the exponential law $e^{i\phi(x,y)}$ due to the way the SLM displays the input phase pattern $\phi(x, y)$ and the second is the square law introduced by the detector, which takes the square of the modulus of the output complex optical field $|E(x, y)|^2$. Finally, the NN can also take into account temporal drifts of the transmission properties of the fiber, solving also bending issues.

The studies for adapting a NN for image recognition in a MMF started in 1991 by Aisawa et al.^[141] They proposed a way to perform image classification through MMF using a NN. In 2001 Marusarz et al.^[142] wrote a paper where they retrieved the transmission property of the fiber using a NN-based technique. In these preliminary works, the constructed NN algorithms were primitive compared to the modern DNNs.

5.4. Recent Applications to Multi-Modal Fibers

In 2017, Takagi et al.^[143] presented a ML algorithm for object recognition in MMF that can be treated as a strongly scattering medium. They experimented the performance of three well-known supervised learning algorithms when studying propagation in MMF: SVM, adaptive boosting (AB)^[144] and a NN. The experimental setup is realized with a SLM and a MMF. Face and nonface images are used as input object images. They are displayed onto the SLM, which is illuminated by a laser diode. A image sensor, which measures the output intensity, collects the speckle patterns used for training and testing the different supervised learning algorithms which classifies the speckle in face and nonface speckles. The training set size was 2000 (1000 face images and 1000 nonface images) and the testing set size was 400 (200 face images and 200 nonface images). The accuracy of the

three algorithms is calculated by considering randomly selected sampling pixels in the speckle patterns, that were set at the same positions for all the speckle patterns both in the training and test processes. Their results showed that the SVM had the highest accuracy rates at all numbers of sampling pixels and also at all numbers of training sets, demonstrating also that all the supervised learning methods achieved high accuracy rates around 90 % for classification.

In 2018, Borhani et al.^[145] used a DNN to interpret the transmission properties of a MMF using the speckle patterns produced by launching images in a MMF. The database, used to train and assess the recognition and reconstruction performances of the network, is constituted by handwritten digits. The experimental setup is constituted by a SLM and a graded-index (GRIN) MMF (62.5 μm core diameter, 4500 spatial modes). The input images are displayed on the SLM, illuminated by a laser diode, and are imaged on the proximal facet of the MMF by mean of a 4f-system. The output images are collected through a CCD which measures the output intensity. A half-wave plate and a linear polarizer are placed before and after the SLM, respectively, in order to test both phase and amplitude patterns as inputs to the GRIN fiber. The NN used to classify the distal end images and the reconstructed SLM input images is a visual geometry group (VGG) type CNN. It consists of a convolutional front-end with downsampling for encoding and a fully connected back-end for classification. A U-net CNN, exploiting the architecture developed by Ronneberger et al.,^[55] is used to reconstruct the SLM input images from the output speckle patterns. They showed that the fidelity of the reconstruction, which is based on the reconstructed images obtained from the experiments, decreases from 97.6% for a 0.1-m fiber to 90.0% for a 1-km fiber. Moreover, the classification accuracy, defined as the percentage of correctly recognized digits, decreases with increasing fiber length for both amplitude and phase-modulated proximal facet input modes from 90% for a 2-cm fiber to 30% for a 1-km fiber.

In the same year, Rahmani et al.^[146] showed that a DNN can learn the input–output relationship in a 0.75-m long MMF. Specifically, they demonstrated that a CNN can learn the nonlinear relationships between the amplitude of the speckle pattern (phase information lost) obtained at the output of the fiber and the phase or the amplitude at the input of the fiber. The training set used to retrieve amplitude-to-amplitude and amplitude-to-phase relationships is constituted by the speckle patterns of handwritten Latin alphabet. They used two different architectures: a 22-layers CNN, based on a VGG type network, and a 20-layers CNN based on a ResNN. The dataset is constituted by 60 000 images for the training set and 1000 images for the validation set and the learning algorithm uses a MSE loss function. They showed that the VGG type network can generate input SLM amplitude and phase patterns with average 2D correlations of $\approx 93\%$ for amplitude patterns and $\approx 79\%$ for phase patterns. Besides, the model can reach fidelities on the validation set as high as $\approx 98\%$ for amplitude patterns and $\approx 85\%$ for phase patterns. On the other hand, the ResNN type architecture reproduces input amplitudes with a fidelity of $\approx 96\%$ and input phases with a fidelity of $\approx 88\%$ with a much faster convergence rate. The novelty of this work is the capability of the networks of transferring the learning, testing it for phase and amplitude reconstruction in images different from the ones used in the

training and validation set. Just using the VGG-net architecture, the reconstruction accuracy reached $\approx 90\%$. Finally, from the measured transmission matrix they computed the inverse matrix in order to compute the input phase pattern displayed on the SLM from desired speckle patterns. Hence, the NN is fed with these input images. The output of the CNN will give the image captured on the camera, that were generated at the output of the MMF with a fidelity that could be as high as $\approx 94\%$. In 2019, Kürüm et al.^[147] used a multi-core, MMF (MCMMF) array as a multiplexed speckle spectrometer, achieving real-time spectral imaging over several thousands of individual fiber cores. Indeed, a MCMMF bundle can be used as a frequency characterization element in a high-throughput imaging spectrometer for snapshot spatial and spectral measurements with sub-nanometer spectral resolution, using a compressive sensing (CS) algorithm to retrieve the spectral information. They showed how DL can perform the same work as well as CS. The experimental setup is characterized by a supercontinuum light source which is spectrally filtered by an acousto-optic tunable filter achieving a spectral resolution of 5 nm. The spectrally filtered light is first sent to a single-mode fiber to avoid any other spectral drift introduced in the setup and then sent to the MMF array of 3012 fibers with individual core diameters of $50\ \mu\text{m}$ by means of a SLM. A CMOS camera is used to retrieve the output speckle patterns. Since all fiber cores are different, they give rise to different speckle patterns at the output. Moreover, the speckle patterns for every wavelength are stored into a multispectral transmission matrix for every core, which in principle allows to retrieve spectral information from arbitrary superposition states using number different techniques. Spectra consisting in many wavelengths correspond to the superposition of different speckle patterns.

In this work, Kürüm et al. performed spectral reconstruction via DL by using a CNN constituted by a series of convolutional layers followed by two fully connected layers and a final dense output layer with 43 neurons, correspondent to the size of the retrieved spectrum. The size of the network is adjusted according to each tested sampling condition, that is, to the number of pixels of the CMOS camera selected to feed the network. Each convolution is followed by batch normalization and a leaky ReLU activation layer. To test the performance of each network, multiple patterns were digitally added up together to simulate a real signal made of a given number of nonzero wavelength components with randomly varying intensities. For each MMF, the dataset was constituted by 31 000 images, 29 000 used for training, 1000 for validation and 1000 for the final evaluation. The performance of the NN were assessed both in the case of downsampling and oversampling. For downsampling, DL yields a very good performance and even clearly outperforms CS for dense spectra in the under-sampling case. For the oversampling case, the DL shows weaker performances. This is due to the statistical nature of the NN compared to the CS approach, which is an analytical approach which in any case yields to the optimal solution, even if at a higher computational cost.

In 2019, Caramazza et al.^[148] implemented a method that statistically reconstructs the inverse transformation matrix for propagation in MMF. The main goal is to transmit natural scenes at high frame rates, high resolutions and in full color. They used a shallow network constituted by a fully connected complex-valued

matrix. The output speckle patterns of the MMF with amplitude distribution, x , that corresponds to the square root of the measured speckle intensity patterns, are fed to the fully connected layer together with the intensity images, I , that have generated each speckle pattern. The algorithm approximates the inverse of a complex transmission matrix, W , such that $I = |Wx|^2$. The obtained W is then used to retrieve images that were not part of the sample dataset starting from intensity measurements of their output speckle patterns. The training dataset consists of images from the ImageNet database.^[104] During the training procedure, the matrix W is changed through a stochastic gradient descent approach thus ensuring convergence of the loss function to a minimum value.

In 2020, Rahmani et al.^[149] proposed an online learning approach for the projection of arbitrary shapes through a MMF when a sample of intensity-only measurements is taken at the output. They used a NN to solve the highly ill-posed problem of predicting a scattering medium system's forward and backward response functions. With respect to previous works, in which DNNs have been used to predict the input field from amplitude-only speckle pattern at the output of the fiber, they realized a NN able to learn the correct inputs that will generate a desired output of a MMF. This is challenging because they did not have a training set that consisted of desired output of the MMF with the corresponding input fields to feed the MMF. The novelty of their work is that they used a combination of two networks to generate the inputs that created a desired target. Indeed, the whole network, called projector network, is made of two sub-networks: actor and model. These two subnetworks work synergistically: starting from speckle patterns, the actor produces at its output the SLM input patterns to feed the fiber. The model then is meant to mimic the forward propagation of light into the MMF producing from the inputs (the SLM patterns) the desired output (the speckle images), backpropagating the error between the desired target and the speckle pattern measured at the distal end of the fiber. When this error reaches the actor, its parameters are adjusted thus reducing the error given by the model.

The entire learning process is divided into three main steps, sketched in **Figure 9**: 1) a number of input control patterns are sent to the system and recorder to the camera; 2) the model is trained on this input patterns in order to learn the forward path of light from the SLM toward the MMF till the camera at the distal end of the fiber; 3) While the model is fixed, the actor is fed with a desired output to generate a given SLM image correspondent to that target image. The actor-produced SLM image is passed to the fixed model now mimicking the fiber. Finally, the error between the output of the model and the target image is backpropagated via the model to the actor to update its trainable weights and biases. After this training, the test procedures to assess the accuracy of the neural network is performed by feeding the projector with a target image, obtaining the SLM image which will be given to the fiber to obtain the real output, which is then compared with the desired one.

The set of images used to produce the training set consists of handwritten Latin alphabet from EMNIST.^[150] After the training, the NN is used directly to project a different category of images, thus showing the generalization ability of the projector network reaching accuracy as high as $\approx 90\%$ even with images

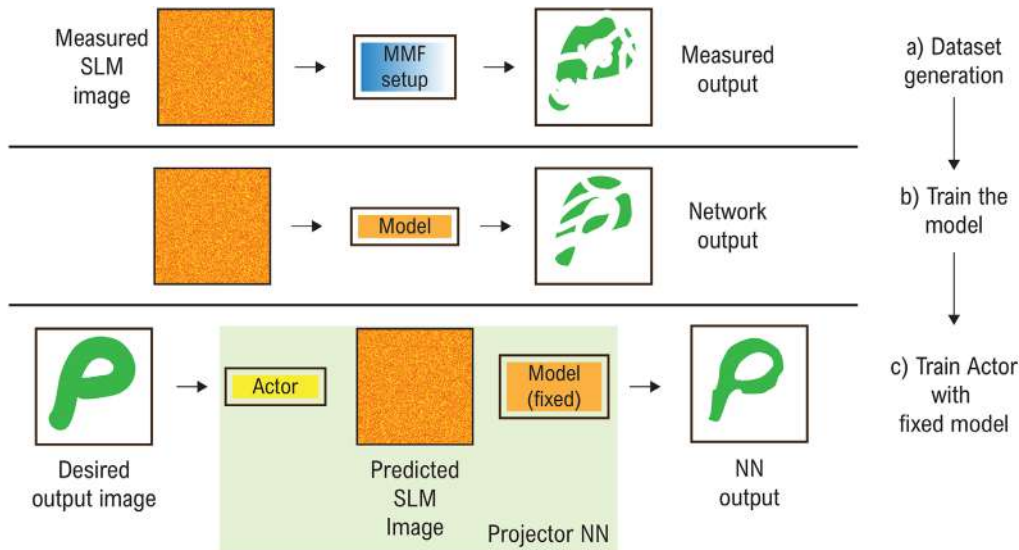


Figure 9. The projector neural network consists of two subnetworks: actor and model. The actor, once trained, takes as input a target image, that corresponds to the output speckle pattern collected at the distal end of the MMF, and gives at its output the correspondent SLM pattern, that experimentally is then delivered at the frontal end of the fiber and propagates into it giving the output speckle pattern. The role of the model is to help the actor in this operation by mimicking the forward propagation of light into the MMF. a) The experimental generation of the training set, made of input SLM patterns and output speckle patterns. The experimental setup, as the one shown in Figure 8, creates the input–output pairs. b) The training of the model using as input the measured SLM patterns and getting at the output the speckle patterns, so that the DNN can learn the rules of propagation through the MMF. c) The training of the subnetwork actor while the subnetwork model is kept fixed. The actor starting from desired output image will produce at its output a predicted SLM image that will go through the model to generate the NN output then compared with the input one. Hence, the error is backpropagated via the fixed model to the actor to update its weights and biases. Eventually, the network is tested by feeding the actor with desired output images and delivering the predicted SLM images directly to the MMF system measuring the output speckle pattern with a CCD. Adapted with permission.^[149] Copyright 2021, The Authors, under exclusive license to Springer Nature.

not included in the training set. It is also shown that the performance of the network in inferring the SLM images are strictly related to the complexity of the target image. Nevertheless, training the projector network on complex images, even though the convergence speed is lower, has shown that it is able to provide SLM images with fidelities comparable with those of full-measurement schemes.

Another work on image transmission through a MMF via DL was done by Kurekci et al.^[151] They built three different CNNs based on U-net, VGG-net e RES-net architectures trained with 31 200 grey-scale handwritten letters of the Latin alphabet and using a MSE cost function. After the training, the networks performances are assessed using 5200 images of handwritten Latin letters not included in the training set. The result of their studies shows that the RES-net architecture is the best compared to the other two architecture both in terms of accuracy and of computational time. After the reconstruction of the input field, once the networks are fed with the speckle patterns obtained from experiments, it is shown that the ResNet and the U-Net both converge (minimize the validation set loss) in less than 20 epochs, after which their validation set losses increase slightly and their training losses keep decreasing. This is not the same in the VGG-net architecture, which is due to the fact that the ResNet and the U-Net are both architectures where the input features get stacked and represented in smaller matrices in the encoder part and then progressively decoded to reconstruct the MMF input, while the VGG-net preserves the input shape through the network by re-shaping layers.

5.5. Highly Stable Information Retrieval in Perturbed Multi-Modal Fibers

In 2019, Fan et al.^[152] developed a CNN with the capability to accurately predict unknown information at the other end of a MMF at any state. Indeed, any change of the fiber geometry leads to different MMF transmission matrices (TM) and so to different states. Introducing high variability in the MMF shapes, they developed a CNN network able to: predict the MMF transmission feeding the NN with the output speckle patterns obtained by the measured TM, perform image retrieval at different states of a stationary MMF and also when continuous shape variations in the MMF occur. The experimental setup is characterized by a digital micromirror device (DMD), which is illuminated by a laser at 632.8 nm and which can display different binary patterns switching on and off the single DMD modules. This binary pattern modulates the light, which reaches the proximal end of the MMF, while the output speckle pattern of the MMF is sent to a CMOS camera.

The training set is constituted by 28×28 pixels images which are converted to 36×36 pixels images to match the DMD pattern requirements, where they are then converted to binary images. After calculating the TM matrices with 8000 input–output pairs, they used 7800 speckle patterns calculated from experimental TM. To estimate the performance of the whole network other 780 speckle patterns were derived from other images taken from a different database. The average prediction accuracy between the predicted binary images and the ground truth is 98.74%. For the

second experiment, the experimentally acquired input–output pairs are used to train directly the CNN, without deriving them from the TM. The collection of input–output pairs is repeated for different MMF geometric shapes. Starting with a dataset of 40 000 images, they used them in a ratio 9:1 for the training and testing set, respectively. They showed that, using for the training process all the images obtained at the different MMF states, the trained CNN has an accuracy above 96% for all possible geometric shapes. Finally, they fed the NNs with a dataset of speckle images acquired while continuous variations were induced in the MMF. Out of these images, some speckle input images and their corresponding binary DMD labels are randomly selected to be used as training data set, while the remaining speckle inputs and their corresponding patterns are used for testing. The average prediction accuracy is 96.48%.

In 2020, Kakkava et al.^[153] proposed an alternative approach for recovering the information through the MMF system in the presence of a wavelength drift in the light source using DNNs. Indeed, perturbations of the system caused by wavelength, thermal or mechanical drift, may be catastrophic for a calibration-based technique such as TM. Therefore, it is needed to look for a technique which is calibration free. The dataset is obtained by taking 10 000 images of handwritten digits from MNIST^[100] and considering the input field at the SLM of the experimental setup and the output speckle pattern at the CCD set at the distal end of the MMF. The wavelength drift is induced through a Matlab script, generating an array of 100 different wavelengths, which are sorted in an ascending order to guarantee fast stabilization of the laser.

The NN used for image classification through MMF is a VGG-type CNN. It is trained with batches of 100 images and it relies on a MSE cost function. In order to test the performance of DNNs in the presence of wavelength drift, it is essential that the classification accuracy is first determined for the different wavelengths within the drift bandwidth in no-drift conditions. This approach takes into account that for different wavelengths there is a different number of supported modes that could result in low capability of the system to support the input images. In this way, the accuracy is only related to the noise induced in the dataset by the wavelength drift. The training and test sets consist of images captured at a single wavelength, without any drift during recording. After establishing that in the range from 700 to 1000-nm there is a good number of spatial modes coupled with the MMF that do not affect the input images accuracy, they considered different bandwidths (from 6 to 96-nm) around a central wavelength of 800-nm. The training is performed in two different cases. In the first case, the wavelength is kept fixed during the training and then tested on speckle patterns where it drifts. In the second case, the training is performed for speckle patterns related to different values of the wavelength. For the first case they show that the DNN is efficient only for a very narrow range of wavelengths, after which the accuracy drops abruptly to 10% at 812-nm. Instead, for the second case, a classification accuracy of 70% is achieved even with a drift of 100-nm. Moreover, varying the dataset size, they showed that the more severe the wavelength drift, the more samples are needed for achieving higher classification accuracies. But in any case, a dataset larger than 6000 is not needed for 100 nm wavelength drift, since the accuracy saturates even using larger datasets.

5.6. Nonlinear Frequency Conversion Control

All the previously reviewed works about DL applications in MMF deal mainly with spatial control of light propagation via wavefront shaping with a SLM. Recently, Tegin et al.^[154] have proposed a ML approach to learn and control nonlinear frequency conversion inside MMF. The physical processes involved in the creation of new optical frequencies in MMF are cascaded SRS as well as supercontinuum generation. They studied the effect of the initial spatial excitation condition of a GRIN MMF on the output spectrum. In particular, they showed that they showed that two highly nonlinear phenomena such as supercontinuum generation and broadening of the spectrum based on cascaded SRS can be experimentally controlled for the first time in the literature with ML tools. Numerical calculations allowed the authors to determine the preliminary excitation patterns to feed the MMF, numerically solving the multi-modal nonlinear Schrödinger equation with a Raman scattering term and a third-order dispersion term. Simulating the propagation of different modes inside the fiber, they showed that spectral broadening can be obtained either by favoring lower order modes or, keeping the pump parameters fixed, when equal excitation of all the modes is provided. Finally, smoother supercontinuum formation is achieved when coupling most of the energy to higher-order modes.

The experimentally measured spectra at the output of the fiber (generated by the propagation of 10 ps short pulses in a 20-m GRIN MMF with 62.5- μm core diameter) are fed to the network as inputs, and for each spectrum, the coefficients to generate the corresponding beam profiles are the output variables of the network, constituted by four hidden layers. By adjusting the peak power of the pulses to 85 and 150 kW, they can study the two nonlinear broadening phenomena. Indeed, the latter peak power favors spectral broadening induced by SRS processes, while the former favors supercontinuum generation at the output of the MMF. The experimentally collected datasets are divided with a ratio of 9:1 for training and validation. To assess the performance of the trained network, a collection of synthetic spectral shapes is generated via summations of Gaussian distributions with different amplitudes and widths. These synthetic spectra are fed to the DNN which provides the parameters for the input field at the proximal end of the MMF. Comparing the experimental result with the designed spectra, to which noise is also added to simulate real measurements, an accuracy higher than $\approx 80\%$ is achieved. In this way, they showed how ML can be used also to predict highly nonlinear effects with a good accuracy.

Another approach in which AI has been used for automated control of highly nonlinear optical processes has been proposed by Valensise et al. in 2021.^[155] They used a Deep RL algorithm to control and optimize white light continuum (WLC) generation in bulk media without a-priori knowledge of the system dynamics or functioning. WLC generation is a very complex task which involves many nonlinear optical processes such as self-phase modulation, self focusing, self-steepening, space–time focusing, group velocity dispersion as well as femtosecond filamentation. The high complexity of these processes, combined with fluctuations of the parameters of the driving laser, call for a time-consuming optimization procedure to obtain a broad and long-term stable WLC. The experimental setup (see **Figure 10a**) is made of a fiber-based ytterbium laser system, generating

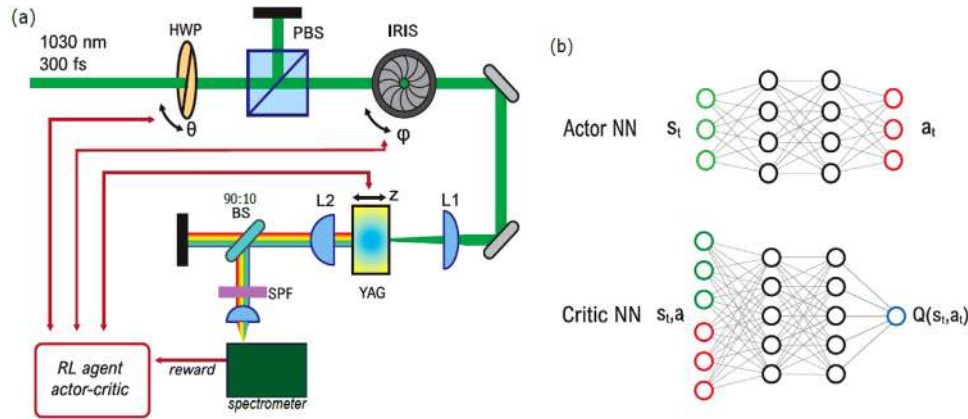


Figure 10. a) WLC generation setup modeled as a Markov decision process. The state of the system s_t is a three-component vector (z, θ, ϕ) . The action a_t is the absolute movement of the three actuators. The reward is provided to the RL agent, calculating it from the measured spectra acquired while the three actuators are moving. b) Actor–critic architecture constituted by two NNs trained for different purposes. The actor NN approximates the policy function π , which maps the relationship between the state s_t and the action a_t . The critic NN approximates the state-action value function $Q(s_t, a_t)$, that is an estimation of the cumulative reward, once the state s_t and the action a_t are provided to the network. Adapted with permission.^[155] Copyright 2021, Optical Society of America.

300-fs pulses at 1030-nm and 2-MHz repetition rate, which are tightly focused with a 5-cm lens on a 6-mm YAG crystal mounted on a motorized translational stage to adjust the position of the laser focus z . Before the lens, a combination of a HWP, mounted on a motorized rotational stage θ , and a polarizing beam splitter (PBS) is used to control the pulse energy, while a second rotary stage controls the aperture of a iris ϕ to regulate the beam divergence. Finally, the collimated beam is sent to a visible spectrometer which records the WLC spectra.

In this work, the WLC generation system is designed as a Markov decision process (MDP) in which an agent, after the observation of the actual state s of the system, acts on the three parameters z, θ , and ϕ following a policy π . This action a brings the system to a new state s' . During each action, WLC spectra are acquired and a reward, corresponding to a single scalar number, is given to the agent. Dealing with continuous parameters, a twin delayed deep deterministic policy gradient (DDPG) actor–critic architecture^[156] (see Figure 10b) is used for training the system. The training procedure is divided into episodes each lasting 50 steps of the three actuators. After sampling the parameter space (four episodes), an evaluation phase (three episodes) is performed to see what the agents learned. Typically, the agent needs two further explorations (two episodes), each one followed by an evaluation phase (three episodes), to acquire the correct knowledge. At each exploration step, when an action is taken, random noise is added to the actuator thus allowing the agent to explore new states which may lead to the optimum policy π^* . Its only at the third evaluation phase that the RL agent is able to give at each episode positive rewards. After training, the RL agent was able to learn how to switch on and to obtain a long-term stable WLC, showing that the broader and most intense spectrum is not always the one chosen by the agent since it aims at maximizing the cumulative rewards.

In ref. [157] Salmela et al. presented a solution to the problem of ultrashort pulses propagation in optical fibers using a machine-learning based paradigm with a RNN. Specifically, they demonstrated how a RNN with LSTM accurately predicts the

temporal and spectral evolution of higher-order soliton compression, studying the propagation of picosecond pulses in the anomalous dispersion regime of a highly nonlinear fiber. Then, they extended their analysis to more complex propagation dynamics, such as the generation of a broadband supercontinuum by injecting femtosecond pulses into a highly nonlinear fiber. Their work demonstrated that the NN was able to reproduce the dynamics both in the temporal and spectral domain both for soliton compression and supercontinuum generation. Moreover, their results for the case of higher-order soliton compression were in excellent agreement with experiments, showing that NNs can be used as an important and standard tool for analyzing complex ultrafast dynamics.

6. Application of AI to Quantum Optics

The development of quantum technologies has now reached the stage in which some form of automated data processing is strongly desirable. This need arises from the large amount of data that a complex quantum system can generate as well as the necessity of not relying on an operator who acts on the system. ML thus appears as an appealing technique to handle such problems. In particular, in the field of quantum optics,^[158] the complexity of the new experiments is constantly increasing. We now have the equipment and the platforms to generate high dimensional, multipartite entangled states, involving physical systems composed of more than two subsystems, which can be manipulated to achieve different tasks. One of the direct consequences of dealing with such complex systems is that the control and the characterization of the generated states require bigger efforts both in terms of computational costs and in the ability of modeling their behavior. Indeed, while the full characterization of a classical system requires a number of parameters that scales linearly with the systems size, the number of measurements and parameters needed to describe the produced quantum states scales exponentially with their dimensions. Such exponential scaling is intrinsically linked to specific properties of quantum phenomena.^[159]

Therefore, the use of ML methods appears especially useful in noisy experimental conditions, where the application of the theoretical model can fail and the development of a specific model results to be extremely hard in particular for high dimensional systems. The increased complexity of the available photonic quantum resources is the main reason why in the last years the number of experiments resorting to ML has rapidly spread in this field. Its use has been demonstrated to be beneficial in different aspects that we will examine individually in the following, that is, the generation of quantum states, their use in metrological applications and ultimately their characterization.

6.1. Generation of Quantum States of Light

Photonic platforms represent a promising candidate to produce a huge variety of entangled multiphoton states. However, the difficulties found in the design of new and efficient optical experiments increase both with the dimensions and the complexity of the desired states. Lately, AI protocols have been employed to find the optimal configuration of optical elements producing the quantum state of interest from the initial state available.^[160,161] This kind of problems can be efficiently solved by RL algorithms where an agent is trained to search the interesting configurations producing the desired states.

In ref. [162], Melnikov et al. developed a RL protocol, formulated within the projective simulation framework, to design complex quantum photonics experiments. The quantum state is encoded in the orbital angular momentum (OAM) of photons produced by a double spontaneous parametric down-conversion (SPDC) process in two nonlinear crystals. The authors give the agent two different tasks: the first is to find the simplest setup which allows to produce a quantum state with a certain set of properties while the second consists in finding as many experimental configurations as possible bringing to the generation of the same state. To achieve such tasks in each iteration of the algorithm, the agent has access to a set of optical elements including beam splitters, mirrors, shift-parameterized holograms, and Dove prisms which it can sequentially place on the optical table. After the analysis of the state obtained upon the evolution through the chosen elements, the agent either receives a reward or not as illustrated in **Figure 11**. The reward is linked to the generation of the targeted multipartite entangled states and it depends on which one of the two tasks we are looking at. The obtained configurations demonstrate how AI algorithms can be employed even during the design of new optical experiments. Interestingly, this kind of approach, which allows the investigation of millions of different quantum optical experiments, brought to the discovery of new unconventional setups which have been used to obtain the first experimental realization of higher dimensional highly entangled states and new quantum techniques. As new protocols and applications are found, this way of accessing arbitrary states becomes a key asset.

The combination of ML algorithms and photonics can also be exploited to improve the efficiency of the ML algorithms themselves. Concerning RL algorithms, recently there is a growing interest also to their implementation on photonic architectures.^[163] In ref. [164] Saggio et al. demonstrated a speed-up in learning

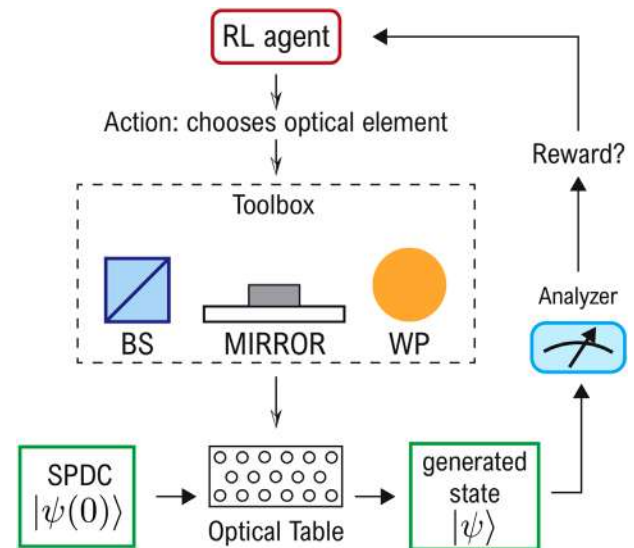


Figure 11. An initial quantum state generated via a SPDC process passes through a series of optical elements chosen by a learning agent. The agent has access to a toolbox with different optical elements that can be placed on the optical table through actuators. Depending on the agent choice, a great variety of different quantum states can be generated and, according to a specific task, the agent will be rewarded or not.

time exploiting quantum resources, paving the way for quantum-enhanced RL algorithms.

6.2. Applications to Metrology and Sensing

The second important application of ML methods to quantum experiments is their use to avoid all the difficulties arising from the development of a theoretical model able to describe the quantum system behavior in a noisy environment. In this scenario, NN and other ML algorithms can be used to map inputs to outputs resulting in a faster and simpler solution than finding an explicit model, since they represent an effective description learned directly from data.

In this context, ML has found an application in quantum phase estimation protocols which represent an important benchmark in the metrology field. The parameter of interest is an optical phase shift, introduced by the investigated sample, among two different modes of the optical state used as probe. The task of metrology experiments consists in estimating such phase with the smallest uncertainty achievable, which has a fundamental lower bound introduced by quantum mechanics laws,^[165–167] measuring the optical probe after its interaction with the sample. To reach this goal, it has been demonstrated that the use of quantum resources plays a fundamental role. Indeed, using optical states with nonclassical features, such as entanglement, it is possible to achieve the ultimate limit of measurement precision.^[168,169] A standard class of single-photon states used for optical metrology purposes is the one of $N00N$ states, where a fixed number of N photons are distributed in a superposition of two modes

$$|\Psi\rangle = \frac{1}{\sqrt{2}} (|N\rangle|0\rangle + e^{i\varphi}|0\rangle|N\rangle) \quad (16)$$

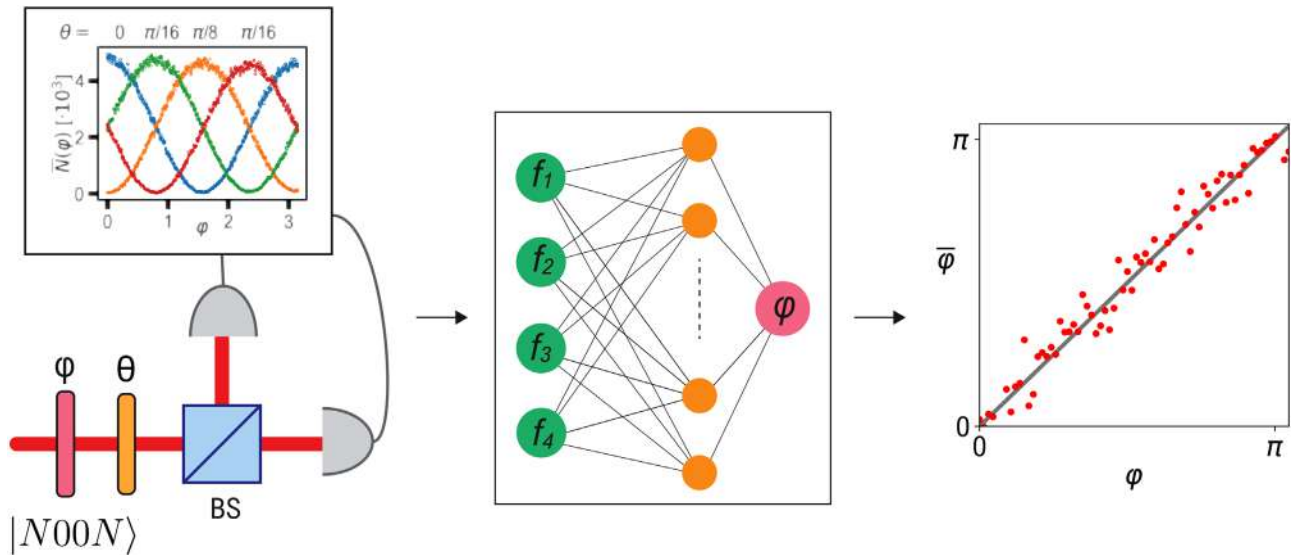


Figure 12. A two-photon $N00N$ state acquires a phase shift φ among its two optical modes that can be detected studying the coincidences counts obtained at the output of a PBS for four different projection angles θ . During the training procedure, a calibrated HWP is inserted in the setup to introduce a known value of the phase φ . The training is performed associating to each of the inspected phase values the four normalized measured photon counts which are fed to the input layer of the NN. The training allows the NN to correctly estimate the value of $\bar{\varphi}$ directly from the four measured outcomes when a new value of the phase is introduced in the setup.

giving a superposition of either all the photons in the first mode and the vacuum state $|0\rangle$ in the second one or vice-versa.

The use of such maximally entangled states for quantum enhanced phase estimation entails that, when passing through the sample, all the N photons acquire at the same time the phase φ of interest, resulting in an oscillation of the registered photon-counting outcomes with a phase $N\varphi$. The improved metrological capabilities of this class of states derive from such faster change of the photon-counting probabilities compared to the one obtained with a separable state, resulting in a superior precision than the one attainable with classical light of the same average energy. However, to achieve such superiority, it is essential to reach an accurate description of the quantum state of the probe. Indeed, the generated states are easily degraded by the presence of unavoidable experimental noise. Therefore it is necessary to develop a reliable calibration procedure to fully exploit the introduced quantum advantage. Usually, a detailed theoretical model of the device operation is developed to reach a high measurement accuracy. In general, this represents a complicated problem since the task of modeling all the noise sources and how they affect the optical probe state becomes harder the more complex and the bigger is the quantum state exploited.

In ref. [170], Cimini et al. demonstrated how ML algorithms can represent a convenient solution for such calibration tasks, explaining how to characterize a quantum phase sensor based on a two-photon $N00N$ state generated exploiting the Hong–Ou–Mandel (HOM) effect.^[171,172] The parameter of interest is the polarization rotation introduced by a HWP at the angle $\frac{\varphi}{2}$, which results in a phase shift between the right- and left-circular polarization state of the probe $N00N$ state. The detection scheme, shown in **Figure 12**, consists of a second HWP and a PBS, allowing the state projection on arbitrary linear polarizations via the choice of the angular position θ of the HWP. Photon count-

ing is performed by fiber-coupled avalanche photodiodes (APD) placed at each of the two outputs of the PBS. The electric signals converted by the APDs are then carried to a field programmable gate array (FPGA) board, which allows one to obtain coincidence counts. The usual modus operandi consists in building a model which links the coincidences detection probability, for four different settings θ of the measurement HWP, to the parameter of interest φ . In this specific case, such conditional probability relies on the precalibration of the visibility ν of the HOM dip and an incorrect determination of the precalibrated visibility can affect the value of the phase parameter, introducing a bias in the estimation.

A feed-forward NN is used to map the obtained coincidences probability for four different measurement settings, fed to the input layer of the NN, to the relative optical phase φ . The network is trained with a calibrated HWP registering the four detection outcomes for different rotation angles of the measurement HWP. The bigger is the training set size, the better will be the reconstruction performed by the NN. Moreover, to obtain a robust model against Poissonian noise affecting the photon counts, a bootstrapping procedure to augment the training set has been implemented.

After the network has been trained, the calibrated HWP is substituted with the sample which introduces the unknown phase shift φ between the two modes of the optical $N00N$ state. The estimation of such phase is achieved feeding the NN with the coincidences counts registered after the probe interaction with the sample achieving a near-optimal estimation independently on the level of the probe signal exploited.

The ability to calibrate sensing devices, circumventing the need of developing an accurate theoretical model describing the corresponding response function, becomes vital when dealing with more complicated devices, whose operation depends on

multiple parameters. Such dependence would indeed imply to take into account all the cross-talks among the different parameters, resulting in a quite intricate problem to solve. In this context, a NN has been used to calibrate the operation of an integrated three-mode interferometer whose response function depends on the application of two voltages which regulate the relative phase shifts of two arms with respect to the reference one. In ref. [173], the NN approach has been used to successfully perform the calibration of the integrated device in the single photon regime, finding a map between voltages and output photons probabilities.

The great advantage of using ML to handle quantum systems is that, thanks to these algorithms, it is possible to develop mass-production devices, ready to be used even from nonexpert users. In fact, AI allows to rely on an autonomous calibration which does not require either the use of additional states or the development of an explicit model describing the system's behavior. This capability is the main reason why AI is starting to be involved in quantum metrology and sensing applications. Finally, the use of AI algorithms for metrological applications has proved successful for the optimization of the feedback strategy in adaptive phase estimation protocols.^[174,175]

6.3. Classification and Characterization of Optical Quantum States

Quantum photonic states represent an important resource not only for sensing applications but also for quantum communication^[176–178] and computation protocols.^[179,180] To exploit the power of quantum effects in all these fields, it is often necessary to obtain a reliable characterization of the employed quantum states. The generated states are indeed affected by noise and experimental imperfections, therefore the knowledge on the actual state available is acquired only through its complete tomographic reconstruction, that is, reconstructing the density matrix of the state using measurements on an ensemble of identical quantum states. The knowledge of the density matrix ρ indeed allows to fully specify the inspected quantum state.^[159] However, the number of measurements needed to obtain a full tomography scales exponentially with the dimension of the investigated state, therefore for high-dimensional systems it becomes a computationally hard task to solve, requiring the analysis of a huge amount of data. To overcome the problems linked to the exponential scaling, generative models have been employed when it is reasonable to assume that the investigated quantum state satisfies some specific regularity properties.

Of particular interest are those states whose wavefunction can be approximated by a restricted Boltzmann machine (RBM) which is known to be a universal approximator able to learn a general complex distribution just from the inputs. RBMs are an unsupervised learning method which allows one to reconstruct the probability distribution associated to its set of inputs. They consist of a two-layer NN: the visible layer and the hidden layer and connections exist only between visible nodes and hidden nodes. In recent years they have been proved to be an efficient tool to solve quantum-physics problems, as demonstrated in ref. [181]. Restricting to the RBM *ansatz*, it is possible to obtain the quantum state tomography solving an unsuper-

vised ML task. In the photonic framework, the idea developed by ref. [181] has been implemented to obtain the tomography of an experimental two-qubit state and for the reconstruction of a continuous-variable optical state from homodyne quadratures measurements.^[182]

In ref. [183] the task was to perform the tomography of the following two-photon Bell state

$$|\psi\rangle = \frac{1}{\sqrt{2}}(|HH\rangle + |VV\rangle) \quad (17)$$

where H and V refer to the horizontal and vertical polarizations states respectively. In quantum mechanics, due to the collapse of the wavefunction after the measurement process, to obtain information about the initial state, before it is altered by the measurement, it is necessary to have a large number of identically prepared copies of it. The density matrix ρ of the unknown initial state before it collapses, is therefore reconstructed from the statistics of the measurement outcomes registered on all the copies. Considering a series of quantum generalized positive-operator valued measures (POVM) $\hat{\Pi}_a$, where each index a corresponds to a possible measurement outcome, the probability of obtaining the outcome a after the measurement on the state ρ is given by the Born's rule

$$P(a) = \text{Tr}[\hat{\Pi}_a \rho] \quad (18)$$

If the matrix of elements $T_{aa'} = \text{Tr}[\hat{\Pi}_a \hat{\Pi}_{a'}]$ is invertible then it is possible to reconstruct the density matrix of the inspected state. Generative models can be exploited to approximate the probability distribution $P(\mathbf{a})$ resulting from a tomographically complete measurement.

In ref. [183] each qubit of the two-photon state is measured in one of the three Pauli bases ($\hat{\sigma}_x$, $\hat{\sigma}_y$, and $\hat{\sigma}_z$) obtaining nine possible configurations. The algorithm is trained minimizing, over all the measured bases, the Kullback–Leibler (KL) divergence between the measured probabilities, reconstructed upon the outcomes of the Pauli POVM on the two-photon Bell state, and the corresponding model distribution (see **Figure 13**). Since the qubits are encoded in the polarization degree of freedom of the photons, the Pauli measurement basis is selected through quarter-wave plates and half-wave plates. The RBM consists of three hidden neurons and after the training the KL divergence reached a value of 10^{-4} indicating the ability of the network to learn accurately the distribution.

However, there are situations in which either we do not have access to the complete set of measurements necessary for the full tomography or we are interested in only specific properties of the quantum state, therefore the employment of alternative techniques is desirable. There are different examples exploiting NN based algorithms in the photonic context for the identification of a specific quantum characteristic on the investigated state. This ranges from the identification of a negativity in the Wigner function of an optical continuous variable multi-mode state,^[184] to the classification of different optical states with nonclassical features arising from the measured click-counting statistics^[185,186] and to the characterization of vector vortex beams.^[187]

ML algorithms have been demonstrated to be successfully employed also for developing benchmarking techniques to validate

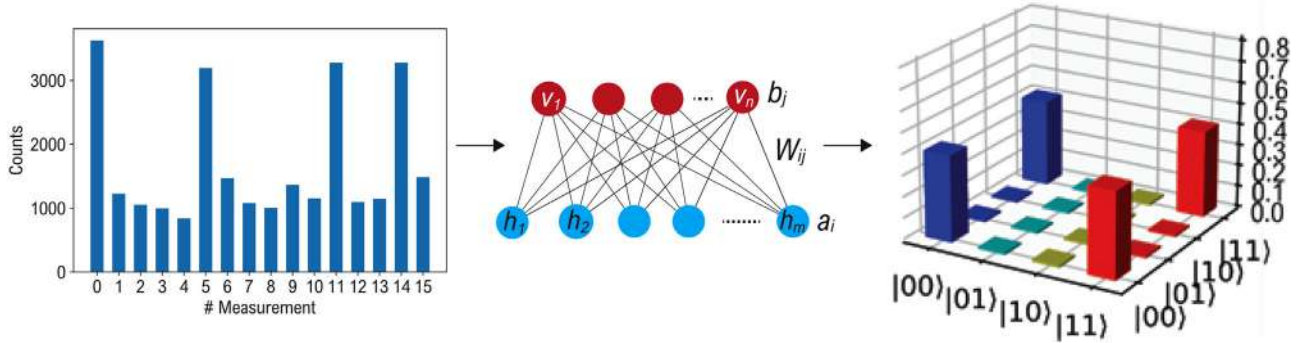


Figure 13. Coincidence count rates obtained measuring with a specific setting the entangled two-photon state generated via a SPDC process. The measurements probability retrieved repeating such procedure for all the elements of the POVM is fed to the visible layer of the RBM which after being trained allows to reconstruct the density matrix of the state.

the correct operation of quantum devices. In this spirit, in ref. [188] Agresti et al. implemented a K-means clustering algorithm to distinguish between boson samplers that use indistinguishable photons from those that do not. As discussed, the use of ML methods appears useful in the quantum optics framework especially for dealing with noisy experimental conditions, where the application of theoretical models can fail and the development of specific ones results to be extremely complex and computationally heavy to solve. Moreover, the great versatility of ML algorithms makes it feasible to adapt them to different systems and experimental conditions, a necessary prerequisite when developing quantum technological devices.

7. Photonic Computing

In the above sections we have extensively discussed applications of AI in the field of optics, highlighting its capability to assist classical and quantum photonics in a wide-range of experimental applications. Recently, a large and growing research area has explored the opposite point of view, namely how to develop photonic platforms that can perform computations and AI tasks. This attempt is motivated by the Bosonic, noninteracting nature of photons that allows one to perform parallel computation with ultra-broad bandwidth. A detailed review of this topic is beyond the scope of this paper. We refer the reader to excellent and detailed reviews^[189–191] that already exist. In any case, we believe that highlighting the main steps in this field, with the latest research results, can be useful in order to understand all the aspects of the fruitful interplay between photonics and AI.^[192]

The first implementations of optical neural networks date back to the 80's.^[193–195] Nevertheless, the available photonic technology^[196] did not allow an easy manipulation of networks weights nor a straightforward implementation of the nonlinearities required for activation functions. These two aspects are the two pillars of research in neuromorphic computing,^[190] a research stream that aims at developing hardware that reflects the features of neural models. Huge advancements in this respect were enabled by the wide development of silicon integrated photonics and by the demonstration that multiply-and-accumulate (MAC) operations, the cornerstone of DNN, may be efficiently computed via optical platforms.^[197] These findings gave new nerve to the neuromorphic computing field.^[198–203]

In parallel to the development of silicon photonics, another intersection between optics and AI emerged in the early 2000s, after the first demonstrations of echo-state networks^[204] and liquid-state machines.^[205] These objects are particular RNN whose neurons are connected by fixed weights. Only the final layer is trained to predict the output, via simple linear regression. These architectures were unified in the concept of reservoir computing (RC):^[206] the fixed weights of the RNN are now replaced by a generic reservoir that can be implemented by any system with rich and stable dynamics.^[207] This new computational paradigm was readily exploited by photonic researchers to implement RC via optical hardware.^[208–214] These advancements in RC are reviewed in refs. [215, 216]. In the last years, research on photonic computation,^[217] neuromorphic engineering^[218–220] and RC are growing.^[207,221,222] Along this research path, new application of photonic hardware and especially SLMs have been demonstrated, such as computing the ground state of systems of interacting spins^[223,224] or perform classical ML task leveraging similar computational frameworks such as extreme learning machines^[225] exploiting light propagation in free-space^[226] or through fibers.^[227]

The idea of exploiting physical systems to overcome the computational limitations encountered by ML algorithms when dealing with high-dimensional datasets is giving rise to a new research branch which exploits the properties of quantum systems in order to optimize classical ML algorithms.^[228–230] Quantum computation indeed has allowed to develop specific algorithms reaching an exponential speed-up compared to their best known classical counterparts,^[231,232] therefore a quantum platform could offer to ML a number of resources inaccessible with classical computers. The use of quantum photonics platforms seems promising to investigate this last aspect as demonstrated in refs. [233–237].

8. Conclusions and Outlooks

AI technologies are becoming more and more ubiquitous and widespread in a broad range of contexts, from commercial applications to advanced scientific research. In this respect, photonics occupies a prominent position.

Numerous experimental contexts may benefit from the unique capability of AI-based algorithms to approximate complex

relationships. This fact allows researchers to boost the performance and ease the complexity of experiments by including AI tools directly in experimental routines. Also, ML models offer novel and powerful data-analysis tools, which are specifically tailored, thanks to the training procedure, for the problem at hand and lead to unprecedented results in most of the cases. In this review, we described the theoretical foundations of ML and DL and showed their application in several experimental photonic contexts.

In spectroscopy, DL models have been used to perform denoising of spectral traces, for example, for removing spurious signals in coherent Raman spectroscopy measurements or for cross-phase modulation removal in ultrafast pump–probe dynamics, and spatial and spectral denoising of hyperspectral data, as the ones generated in coherent Raman imaging applications. Other DL models have been adopted for chemometrics both in spectroscopy and imaging experiments. As spectroscopy is the gold standard for materials and biomaterials characterization, the integration of AI tools for data analysis could accelerate and improve the adoption of advanced spectroscopic techniques at the industrial level for different applications, such as drug screening and material quality control, as well as for medical diagnostics and astronomic research.

NNs are particularly effective to assist optical wavefront shaping when dealing with the nonlinear inverse problem of controlling light after propagation through a diffusive medium. They bypass the need for extended information, such as the complex amplitude of the fields, and for complex experimental systems required by standard analytical approaches based on the solution of Maxwell's equations. For example, ML models have been used in computational imaging to approximate the transmission matrix from speckle patterns after a scatterer to the image or the relationship between the illumination pattern shown on a spatial light modulator and the corresponding light intensity distribution read by a camera, or in digital holography to retrieve the phase information from a single measured hologram. A similar framework is found when light propagates in MMFs, where modal dispersion distorts the wavefront of coherent light propagating inside the fiber. NNs offer valid solutions to reconstruct the nonlinear relationship between input and output in MMFs, to perform object recognition and spectral reconstruction, to reconstruct the input given the speckle pattern at the output of a MMF and for information retrieval in noisy experimental situations. In this sense, ML-driven wavefront shaping techniques constitute a valuable resource for coherent tomography, optical sensing, and quantitative phase imaging in biological applications deriving morphological and mechanical properties at sub-cellular level.

NN-based solutions have been successfully applied not only to data analysis, but also inside experimental setups. We described the use of DL for the generation of new optical frequencies in MMF, for the generation of white-light supercontinuum as well as for the generation and characterization of quantum states of light, the building blocks of all quantum information experiments performed with light. Further developments of these approaches may improve in a non-trivial way the quality and reliability of experimental setups. In quantum photonics, interesting solutions offered by AI tools involve bypassing theoretical models' estimation for metrology and sensing applications, and mod-

eling of quantum states in noisy environments, where analytical approaches may be too computationally heavy. Given enough data representing the behavior of the system in the form of input–output pairs, ML algorithms and NNs can be used to approximate the unknown description.

Eventually, photonic computing is a research field in which the opposite perspective of the relationship between photonics and AI is embraced. In this respect, photonic devices are capable to provide energy effective computing platforms. New opportunities may arise from this research area, especially with the aim of fully incorporating AI algorithms into devices capable of instantaneous and accurate responses.

Acknowledgements

F.V. and A.B. contributed equally to this work. This project has received funding from the European Union project CRIMSON under Grant Agreement No. 101016923 and from the Regione Lombardia project NEWMED under Grant Agreement No. POR FESR 2014–2020. V.C. is supported by the Amaldi Research Center funded by the Ministero dell'Istruzione dell'Università della Ricerca (Ministry of Education, University and Research) program "Dipartimento di Eccellenza" (CUP:B8118001170001).

Open Access Funding provided by Politecnico di Milano within the CRUI-CARE Agreement.

Conflict of Interest

The authors declare no conflict of interest.

Keywords

artificial intelligence, computational imaging, multi-modal fibers, neural networks, photonics, quantum optics, spectroscopy

Received: July 21, 2021

Revised: February 25, 2022

Published online: March 31, 2022

- [1] J. Bughin, J. Seong, J. Manyika, M. Chui, R. Joshi, *Notes from the AI Frontier: Modeling the Global Economic Impact of AI*, McKinsey and Company, New York **2018**.
- [2] J. N. Kok, E. J. Boers, W. A. Kusters, P. Van der Putten, M. Poel, *Artif. Intell.* **2009**, *1*, 270.
- [3] G. Genty, L. Salmela, J. M. Dudley, D. Brunner, A. Kokhanovskiy, S. Kobtsev, S. K. Turitsyn, *Nat. Photonics* **2020**, *15*, 91.
- [4] D. Piccinotti, K. F. MacDonald, S. A. Gregory, I. Youngs, N. I. Zheludev, *Rep. Prog. Phys.* **2020**, *84*, 012401.
- [5] J. Mata, I. de Miguel, R. J. Durán, N. Merayo, S. K. Singh, A. Jukan, M. Chamania, *Opt. Switching Networking* **2018**, *28*, 43.
- [6] Y. Ji, R. Gu, Z. Yang, J. Li, H. Li, M. Zhang, *Sci. China Inf. Sci.* **2020**, *63*, 160301.
- [7] D. Wang, M. Zhang, *Front. Commun. Networks* **2021**, *2*, 656786.
- [8] F. N. Khan, Q. Fan, C. Lu, A. P. T. Lau, in *Optical Fiber Telecommunications VII*, Elsevier, New York **2020**, pp. 921–978.
- [9] B. Karanov, M. Chagnon, F. Thouin, T. A. Eriksson, H. Bulow, D. Lavery, P. Bayvel, L. Schmalen, *J. Lightwave Technol.* **2018**, *36*, 4843.
- [10] D. Zibar, M. Piels, R. Jones, C. G. Schaeffer, *J. Lightwave Technol.* **2016**, *34*, 1442.

- [11] Y. Fumumoto, S. Owaki, M. Nakamura, in *2017 Opto-Electronics and Communications Conference (OECC) and Photonics Global Conference (PGC)*, IEEE, Piscataway, NJ **2017**, pp. 1–2.
- [12] J. Nakamura, K. Ikuta, M. Nakamura, in *26th Optoelectronics and Communications Conference*, Optical Society of America, Washington, D.C. **2021**, pp. JS2B.10.
- [13] K. Ikuta, Y. Otsuka, M. Nakamura, in (Eds: B. Jalali, K. Kitayama), *AI and Optical Data Sciences*, SPIE, Bellingham, WA **2020**, pp. 1129916.
- [14] P. Sajda, *Annu. Rev. Biomed. Eng.* **2006**, *8*, 537.
- [15] J. Fang, A. Swain, R. Unni, Y. Zheng, *Laser Photonics Rev.* **2020**, *15*, 2000422.
- [16] C. L. Chen, A. Mahjoubfar, L.-C. Tai, I. K. Blaby, A. Huang, K. R. Niazi, B. Jalali, *Sci. Rep.* **2016**, *6*, 21471.
- [17] J. K. Zhang, Y. R. He, N. Sobh, G. Popescu, *APL Photonics* **2020**, *5*, 040805.
- [18] A. Esteva, B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, S. Thrun, *Nature* **2017**, *542*, 115.
- [19] J. N. Kather, A. T. Pearson, N. Halama, D. Jäger, J. Krause, S. H. Loosen, A. Marx, P. Boor, F. Tacke, U. P. Neumann, H. I. Grabsch, T. Yoshikawa, H. Brenner, J. Chang-Claude, M. Hoffmeister, C. Trautwein, T. Luedde, *Nat. Med.* **2019**, *25*, 1054.
- [20] P. Lakhani, B. Sundaram, *Radiology* **2017**, *284*, 574.
- [21] Y. Jiang, C. Lei, A. Yasumoto, H. Kobayashi, Y. Aisaka, T. Ito, B. Guo, N. Nitta, N. Kutsuna, Y. Ozeki, A. Nakagawa, Y. Yatomi, K. Goda, *Lab Chip* **2017**, *17*, 2426.
- [22] Y. Li, A. Mahjoubfar, C. L. Chen, K. R. Niazi, L. Pei, B. Jalali, *Sci. Rep.* **2019**, *9*, 11088.
- [23] E. C. Yang, D. R. Brenes, I. S. Vohra, *J. Med. Imaging* **2020**, *7*, 054502.
- [24] H. Parmar, B. Nutter, R. Long, S. Antani, S. Mitra, *J. Med. Imaging* **2020**, *7*, 056001.
- [25] H. Chen, Y. Zhang, W. Zhang, P. Liao, K. Li, J. Zhou, G. Wang, *Biomed. Opt. Express* **2017**, *8*, 679.
- [26] J. Yu, B. Yang, J. Wang, J. Leader, D. Wilson, J. Pu, *J. Med. Imaging* **2020**, *7*, 051202.
- [27] G. Lee, K. Nho, B. Kang, K.-A. Sohn, D. Kim, *Sci. Rep.* **2019**, *9*, 1952.
- [28] I. Goodfellow, Y. Bengio, A. Courville, in *Adaptive Computation and Machine Learning Series*, MIT Press, Cambridge, MA **2016**.
- [29] P. S. Balakrishnan, M. C. Cooper, V. S. Jacob, P. A. Lewis, *Psychometrika* **1994**, *59*, 509.
- [30] C. Zhang, D. Song, Y. Chen, X. Feng, C. Lumezanu, W. Cheng, J. Ni, B. Zong, H. Chen, N. V. Chawla, in *Proc. of the AAAI Conf. on Artificial Intelligence*, AAAI, Menlo Park, CA **2019**, pp. 1409–1416.
- [31] A. Pumarola, A. Agudo, A. Sanfeliu, F. Moreno-Noguer, in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, IEEE, Piscataway, NJ **2018**, pp. 8620–8628.
- [32] X. Wang, A. Gupta, in *2015 IEEE International Conference on Computer Vision (ICCV)*, IEEE, Piscataway, NJ **2015**, pp. 2794–2802.
- [33] R. Sutton, A. Barto, in *Adaptive Computation and Machine Learning series*, MIT Press, Cambridge, MA **2018**.
- [34] S. Gu, E. Holly, T. Lillicrap, S. Levine, in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, Piscataway, NJ **2017**, pp. 3389–3396.
- [35] A. E. Sallab, M. Abdou, E. Perot, S. Yogamani, *Electron. Imaging* **2017**, *2017*, 70.
- [36] C. Bishop, in *Information Science and Statistics*, Springer, Berlin, Heidelberg **2006**.
- [37] P. Mehta, M. Bukov, C.-H. Wang, A. G. Day, C. Richardson, C. K. Fisher, D. J. Schwab, *Phys. Rep.* **2019**, *810*, 1.
- [38] A. Y. Ng, in *Proc. of the Twenty-First Int. Conf. on Machine Learning, ICML '04*, Association for Computing Machinery, New York, NY **2008**, p. 78.
- [39] D. P. Kingma, J. Ba, arXiv:1412.6980 **2017**.
- [40] J. Cervantes, F. Garcia-Lamont, L. Rodríguez-Mazahua, A. Lopez, *Neurocomputing* **2020**, *408*, 189.
- [41] J. Shawe-Taylor, N. Cristianini, *Kernel Methods for Pattern Analysis*, Cambridge University Press, Cambridge **2004**.
- [42] P. A. Romero, A. Krause, F. H. Arnold, *Proc. Natl. Acad. Sci. U. S. A.* **2012**, *110*, E193.
- [43] G. Cybenko, *Math. Control, Signals, Syst.* **1989**, *2*, 303.
- [44] V. Kůrková, *Neur. Networks* **1992**, *5*, 501.
- [45] W. A. Gardner, *Signal Process.* **1984**, *6*, 113.
- [46] M. Li, T. Zhang, Y. Chen, A. J. Smola, in *Proc. of the 20th ACM SIGKDD Int. Conference on Knowledge Discovery and Data Mining*, Association for Computing Machinery, New York, NY **2014**, pp. 661–670.
- [47] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, L. D. Jackel, *Neur. Comput.* **1989**, *1*, 541.
- [48] J. Yosinski, J. Clune, A. Nguyen, T. Fuchs, H. Lipson arXiv:1506.06579 **2015**.
- [49] A. Khan, A. Sohail, U. Zahoor, A. S. Qureshi, *Artif. Intell. Rev.* **2020**, *53*, 5455.
- [50] A. LeNail, *J. Open Source Software* **2019**, *4*, 747.
- [51] Y. Ji, H. Zhang, Z. Zhang, M. Liu, *Inf. Sci.* **2021**, *546*, 835.
- [52] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Piscataway, NJ **2016**, pp. 779–788.
- [53] W. Xie, J. A. Noble, A. Zisserman, *Comput. Methods Biomech. Biomed. Eng.: Imaging Visualization* **2016**, *6*, 283.
- [54] J. Shotton, J. Winn, C. Rother, A. Criminisi, *Int. J. Comput. Vision* **2009**, *81*, 2.
- [55] O. Ronneberger, P. Fischer, T. Brox, in *Lecture Notes in Computer Science*, Springer, Cham **2015**, pp. 234.
- [56] K. He, X. Zhang, S. Ren, J. Sun, in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Piscataway, NJ **2016**, 770–778.
- [57] Z. C. Lipton, J. Berkowitz, C. Elkan, A critical review of recurrent neural networks for sequence learning, **2015**.
- [58] D. P. Mandic, J. A. Chambers, *Recurrent Neural Networks for Prediction*, Wiley, New York **2001**.
- [59] S. Hochreiter, J. Schmidhuber, *Neur. Comput.* **1997**, *9*, 1735.
- [60] A. Graves, N. Jaitly, A.-r. Mohamed, in *2013 IEEE Workshop on Automatic Speech Recognition and Understanding*, IEEE, Piscataway, NJ **2013**, pp. 273–278.
- [61] V. K. R. Chimmula, L. Zhang, *Chaos, Solitons Fractals* **2020**, *135*, 109864.
- [62] D. Polli, V. Kumar, C. M. Valensise, M. Marangoni, G. Cerullo, *Laser Photonics Rev.* **2018**, *12*, 1800020.
- [63] C. M. Valensise, A. Giuseppi, F. Vernuccio, A. D. la Cadena, G. Cerullo, D. Polli, *APL Photonics* **2020**, *5*, 061305.
- [64] Y. Lecun, L. Bottou, Y. Bengio, P. Haffner, *Proc. IEEE* **1998**, *86*, 2278.
- [65] R. Houhou, P. Barman, M. Schmitt, T. Meyer, J. Popp, T. Bocklitz, *Opt. Express* **2020**, *28*, 21002.
- [66] X. Zhang, T. Lin, J. Xu, X. Luo, Y. Ying, *Anal. Chim. Acta* **2019**, *1058*, 48.
- [67] J. Zhang, J. Zhao, H. Lin, Y. Tan, J.-X. Cheng, *J. Phys. Chem. Lett.* **2020**, *11*, 8573.
- [68] A. Bresci, M. Guizzardi, C. M. Valensise, F. Marangi, F. Scotognella, G. Cerullo, D. Polli, *APL Photonics* **2021**, *6*, 076104.
- [69] B. Manifold, E. Thomas, A. T. Francis, A. H. Hill, D. Fu, *Biomed. Opt. Express* **2019**, *10*, 3860.
- [70] H. Lin, H. J. Lee, N. Tague, J.-B. Lugagne, C. Zong, F. Deng, J. Shin, L. Tian, W. Wong, M. J. Dunlop, J.-X. Cheng, *Nat. Commun.* **2021**, *12*, 3052.
- [71] G. Pu, B. Jalali, *Opt. Express* **2021**, *29*, 20786.
- [72] Y.-J. Liu, M. Kyne, C. Wang, X.-Y. Yu, *Comput. Struct. Biotechnol. J.* **2020**, *18*, 2920.
- [73] M. Hedegaard, C. Matthäus, S. Hassing, C. Krafft, M. Diem, J. Popp, *Theor. Chem. Acc.* **2011**, *130*, 1249.

- [74] M. Miljković, T. Chernenko, M. J. Romeo, B. Bird, C. Matthäus, M. Diem, *Anal.* **2010**, *135*, 2002.
- [75] N. Bergner, T. Bocklitz, B. F. Romeike, R. Reichart, R. Kalff, C. Krafft, J. Popp, *Chemom. Intell. Lab. Syst.* **2012**, *117*, 224.
- [76] T. Bocklitz, M. Putsche, C. Stüber, J. Käs, A. Niendorf, P. Rösch, J. Popp, *J. Raman Spectrosc.* **2009**, *40*, 1759.
- [77] L. Xu, Y.-P. Zhou, L.-J. Tang, H.-L. Wu, J.-H. Jiang, G.-L. Shen, R.-Q. Yu, *Anal. Chim. Acta* **2008**, *616*, 138.
- [78] J. Yang, J. Xu, X. Zhang, C. Wu, T. Lin, Y. Ying, *Anal. Chim. Acta* **2019**, *1081*, 6.
- [79] J. Liu, M. Osadchy, L. Ashton, M. Foster, C. J. Solomon, S. J. Gibson, *The Analyst* **2017**, *142*, 4067.
- [80] T. Armbruster, R. M. Danisi, *Highlights in Mineralogical Crystallography*, De Gruyter, Berlin **2015**, pp. 1–19.
- [81] C.-S. Ho, N. Jean, C. A. Hogan, L. Blackmon, S. S. Jeffrey, M. Holodniy, N. Banaei, A. A. E. Saleh, S. Ermon, J. Dionne, *Nat. Commun.* **2019**, *10*, 4927.
- [82] C. Yuanyuan, W. Zhibin, *Chemom. Intell. Lab. Syst.* **2018**, *181*, 1.
- [83] O. Guselnikova, A. Trelin, A. Skvortsova, P. Ulbrich, P. Postnikov, A. Pershina, D. Sykora, V. Svorcik, O. Lyutakov, *Biosens. Bioelectron.* **2019**, *145*, 111718.
- [84] C. Cui, T. Fearn, *Chemom. Intell. Lab. Syst.* **2018**, *182*, 9.
- [85] S. Malek, F. Melgani, Y. Bazi, *J. Chemom.* **2017**, *32*, e2977.
- [86] J. Acquarelli, T. van Laarhoven, J. Gerretzen, T. N. Tran, L. M. Buydens, E. Marchiori, *Anal. Chim. Acta* **2017**, *954*, 22.
- [87] E. J. Bjerrum, M. Glahder, T. Skov, Data augmentation of spectral data for convolutional neural network (CNN) based deep chemometrics, arXiv:1710.01927 **2017**.
- [88] M. Chatzidakis, G. A. Botton, *Sci. Rep.* **2019**, *9*, 2126.
- [89] S. D. Krauß, R. Roy, H. K. Yosef, T. Lechtonen, S. F. El-Mashtoly, K. Gerwert, A. Mosig, *J. Biophotonics* **2018**, *11*, e201800022.
- [90] S. Cheng, H. Li, Y. Luo, Y. Zheng, P. Lai, *J. Innovative Opt. Health Sci.* **2019**, *12*, 1930006.
- [91] S. M. Popoff, G. Lerosey, R. Carminati, M. Fink, A. C. Boccara, S. Gigan, *Phys. Rev. Lett.* **2010**, *104*.
- [92] I. M. Vellekoop, A. P. Mosk, *Opt. Lett.* **2007**, *32*, 2309.
- [93] I. M. Vellekoop, E. G. van Putten, A. Lagendijk, A. P. Mosk, *Opt. Express* **2008**, *16*, 67.
- [94] I. M. Vellekoop, *Opt. Express* **2015**, *23*, 12189.
- [95] A. P. Mosk, A. Lagendijk, G. Lerosey, M. Fink, *Nat. Photonics* **2012**, *6*, 283.
- [96] R. Horstmeyer, H. Ruan, C. Yang, *Nat. Photonics* **2015**, *9*, 563.
- [97] G. Barbastathis, A. Ozcan, G. Situ, *Optica* **2019**, *6*, 921.
- [98] R. Horisaki, R. Takagi, J. Tanida, *Opt. Express* **2016**, *24*, 13738.
- [99] M. Lyu, H. Wang, G. Li, G. Situ, arXiv:1708.07881 **2017**.
- [100] Y. LeCun MNIST handwritten digit classification dataset 1998, <http://yann.lecun.com/exdb/mnist/> (accessed: November 1998).
- [101] S. Li, M. Deng, J. Lee, A. Sinha, G. Barbastathis, *Optica* **2018**, *5*, 803.
- [102] G. B. Huang, M. Mattar, T. Berg, E. Learned-Miller, in (Eds: E. Learned-Miller, A. Ferencz, F. Jurie), *Workshop on Faces in 'Real-Life' Images: Detection, Alignment, and Recognition*, Springer, Berlin, Heidelberg **2008**.
- [103] F. Samaria, A. Harter, in *Proc. of 1994 IEEE Workshop on Applications of Computer Vision*, IEEE, Piscataway, NJ **1994**, pp. 138–142.
- [104] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, L. Fei-Fei, *Int. J. Comput. Vision* **2015**, *115*, 211.
- [105] A. Krizhevsky, G. Hinton, *Learning Multiple Layers of Features from Tiny Images*, University of Toronto, Toronto, Ontario **2009**.
- [106] M. Lyu, H. Wang, G. Li, S. Zheng, G. Situ, *Adv. Photonics* **2019**, *1*, 036002.
- [107] Y. Li, Y. Xue, L. Tian, *Optica* **2018**, *5*, 1181.
- [108] R. Horisaki, R. Takagi, J. Tanida, *Appl. Opt.* **2017**, *56*, 4358.
- [109] A. Turpin, I. Vishniakov, J. d. Seelig, *Opt. Express* **2018**, *26*, 30911.
- [110] Y. Luo, S. Yan, H. Li, P. Lai, Y. Zheng, *APL Photonics* **2020**, *5*, 016109.
- [111] S. Katoch, S. S. Chauhan, V. Kumar, *Multimedia Tools Appl.* **2020**, *80*, 8091.
- [112] T. Alam, S. Qamar, A. Dixit, M. Benaida, *Int. J. Eng. Pedagogy* **2020**, *10*, 57.
- [113] A. Sinha, J. Lee, S. Li, G. Barbastathis, *Optica* **2017**, *4*, 1117.
- [114] P. Picart, J. Li, *Digital Holography*, Wiley, London **2012**.
- [115] Y. Rivenson, Y. Zhang, H. Günaydin, D. Teng, A. Ozcan, *Light: Sci. Appl.* **2017**, *7*, 17141.
- [116] Y. Wu, Y. Rivenson, Y. Zhang, Z. Wei, H. Günaydin, X. Lin, A. Ozcan, *Optica* **2018**, *5*, 704.
- [117] E. S. A. Wertz, *Compt. Rend. Acad. Sci. B* **1967**, *264*, 1015.
- [118] A. Gover, C. P. Lee, A. Yariv, *J. Opt. Soc. Am.* **1976**, *66*, 306.
- [119] A. Friesem, U. Levy, Y. Silberberg, *Proc. IEEE* **1983**, *71*, 208.
- [120] O. Garibyan, I. Kompanets, A. Parfyonov, N. Pilipetsky, V. Shkunov, A. Sudarkin, A. Sukhov, N. Tabiryana, A. Vasiliev, B. Zel'dovich, *Opt. Commun.* **1981**, *38*, 67.
- [121] A. Yariv, J. AuYeung, D. Fekete, D. M. Pepper, *Appl. Phys. Lett.* **1978**, *32*, 635.
- [122] I. Yamaguchi, T. Zhang, *Opt. Lett.* **1997**, *22*, 1268.
- [123] E. Cuche, F. Bevilacqua, C. Depeursinge, *Opt. Lett.* **1999**, *24*, 291.
- [124] I. N. Papadopoulos, S. Farahi, C. Moser, D. Psaltis, *Opt. Express* **2012**, *20*, 10583.
- [125] I. N. Papadopoulos, S. Farahi, C. Moser, D. Psaltis, *Biomed. Opt. Express* **2013**, *4*, 260.
- [126] R. D. Leonardo, S. Bianchi, *Opt. Express* **2010**, *19*, 247.
- [127] T. Čížmár, K. Dholakia, *Opt. Express* **2011**, *19*, 18871.
- [128] T. Čížmár, K. Dholakia, *Nat. Commun.* **2012**, *3*, 1027.
- [129] S. Bianchi, R. D. Leonardo, *Lab Chip* **2012**, *12*, 635.
- [130] E. R. Andresen, G. Bouwmans, S. Monneret, H. Rigneault, *Opt. Lett.* **2013**, *38*, 609.
- [131] Y. Choi, C. Yoon, M. Kim, T. D. Yang, C. Fang-Yen, R. R. Dasari, K. J. Lee, W. Choi, *Phys. Rev. Lett.* **2012**, *109*, 203901.
- [132] A. M. Caravaca-Aguirre, E. Niv, D. B. Conkey, R. Piestun, *Opt. Express* **2013**, *21*, 12881.
- [133] R. Y. Gu, R. N. Mahalati, J. M. Kahn, *Opt. Express* **2015**, *23*, 26905.
- [134] D. Loterie, S. Farahi, I. Papadopoulos, A. Goy, D. Psaltis, C. Moser, *Opt. Express* **2015**, *23*, 23845.
- [135] S. Popoff, G. Lerosey, M. Fink, A. C. Boccara, S. Gigan, *Nat. Commun.* **2010**, *1*, 81.
- [136] J. W. Goodman, *Introduction to Fourier Optics*, Roberts and Company Publishers, Greenwood Village, CO **2005**.
- [137] Y.-C. Lin, C.-J. Cheng, T.-C. Poon, *Appl. Opt.* **2010**, *50*, B25.
- [138] C. J. Mann, L. Yu, C.-M. Lo, M. K. Kim, *Opt. Express* **2005**, *13*, 8693.
- [139] E. Kakkava, B. Rahmani, N. Borhani, U. Teğin, D. Loterie, G. Konstantinou, C. Moser, D. Psaltis, *Opt. Fiber Technol.* **2019**, *52*, 101985.
- [140] M. Närhi, L. Salmela, J. Toivonen, C. Billet, J. M. Dudley, G. Genty, *Nat. Commun.* **2018**, *9*, 4923.
- [141] S. Aisawa, K. Noguchi, T. Matsumoto, *Opt. Lett.* **1991**, *16*, 645.
- [142] R. K. Marusz, M. R. Sayeh, *Appl. Opt.* **2001**, *40*, 219.
- [143] R. Takagi, R. Horisaki, J. Tanida, *Opt. Rev.* **2017**, *24*, 117.
- [144] Y. Freund, R. E. Schapire, in *Proc. of the Sixteenth Int. Joint Conf. on Artificial Intelligence*, Morgan Kaufmann, Burlington, MA **1999**, 1401–1406.
- [145] N. Borhani, E. Kakkava, C. Moser, D. Psaltis, *Optica* **2018**, *5*, 960.
- [146] B. Rahmani, D. Loterie, G. Konstantinou, D. Psaltis, C. Moser, *Light: Sci. Appl.* **2018**, *7*, 69.
- [147] U. Kürüm, P. R. Wiecha, R. French, O. L. Muskens, *Opt. Express* **2019**, *27*, 20965.
- [148] P. Caramazza, O. Moran, R. Murray-Smith, D. Faccio, *Nat. Commun.* **2019**, *10*, 2029.
- [149] B. Rahmani, D. Loterie, E. Kakkava, N. Borhani, U. Teğin, D. Psaltis, C. Moser, *Nat. Mach. Intell.* **2020**, *2*, 403.
- [150] G. Cohen, S. Afshar, J. Tapson, A. van Schaik, arXiv:1702.05373 **2017**.

- [151] S. Kurekci, A. T. Temur, M. E. Odabas, G. Afshari, E. Yüce, in (Eds: C. Fournier, M. P. Georges, G. Popescu), *Unconventional Optical Imaging II*, SPIE, Bellingham, WA **2020**, p. 1135126.
- [152] P. Fan, T. Zhao, L. Su, *Opt. Express* **2019**, *27*, 20241.
- [153] E. Kakkava, N. Borhani, B. Rahmani, U. Teĝin, C. Moser, D. Psaltis, *Appl. Sci.* **2020**, *10*, 3816.
- [154] U. Teĝin, B. Rahmani, E. Kakkava, N. Borhani, C. Moser, D. Psaltis, *APL Photonics* **2020**, *5*, 030804.
- [155] C. M. Valensise, A. Giuseppi, G. Cerullo, D. Polli, *Optica* **2021**, *8*, 239.
- [156] S. Fujimoto, H. van Hoof, D. Meger, in Proc. of the 35th Int. Conf. on Machine Learning, Microtome Publishing, Brookline, MA **2018**, pp. 1–10.
- [157] L. Salmela, N. Tspinakakis, A. Foi, C. Billet, J. M. Dudley, G. Genty, *Nat. Mach. Intell.* **2021**, *3*, 344.
- [158] J. L. O'Brien, A. Furusawa, J. Vučković, *Nat. Photonics* **2009**, *3*, 687.
- [159] M. Paris, J. Rehacek, *Quantum State Estimation*, Springer, Berlin, Heidelberg **2004**.
- [160] M. Krenn, M. Malik, R. Fickler, R. Lapkiewicz, A. Zeilinger, *Phys. Rev. Lett.* **2016**, *116*, 090405.
- [161] P. A. Knott, *New J. Phys.* **2016**, *18*, 073033.
- [162] A. A. Melnikov, H. Poulsen Nautrup, M. Krenn, V. Dunjko, M. Tierisch, A. Zeilinger, H. J. Briegel, *Proc. Natl. Acad. Sci. U. S. A.* **2018**, *115*, 1221.
- [163] F. Flamini, A. Hamann, S. Jerbi, L. M. Trenkwalder, H. P. Nautrup, H. J. Briegel, *New J. Phys.* **2020**, *22*, 045002.
- [164] V. Saggio, B. E. Asenbeck, A. Hamann, T. Strömberg, P. Schiansky, V. Dunjko, N. Friis, N. C. Harris, M. Hochberg, D. Englund, S. Wölk, H. J. Briegel, P. Walther, *Nature* **2021**, *591*, 229.
- [165] D. W. Berry, H. Wiseman, J. Breslin, *Phys. Rev. A* **2001**, *63*, 053804.
- [166] W. Górecki, R. Demkowicz-Dobrzański, H. M. Wiseman, D. W. Berry, *Phys. Rev. Lett.* **2020**, *124*, 030501.
- [167] V. Giovannetti, S. Lloyd, L. Maccone, *Science* **2004**, *306*, 1330.
- [168] V. Giovannetti, S. Lloyd, L. Maccone, *Phys. Rev. Lett.* **2006**, *96*, 010401.
- [169] V. Giovannetti, S. Lloyd, L. Maccone, *Nat. Photonics* **2011**, *5*, 222.
- [170] V. Cimini, I. Gianani, N. Spagnolo, F. Leccese, F. Sciarrino, M. Barbieri, *Phys. Rev. Lett.* **2019**, *123*, 230502.
- [171] C. K. Hong, Z. Y. Ou, L. Mandel, *Phys. Rev. Lett.* **1987**, *59*, 2044.
- [172] F. Bouchard, A. Sit, Y. Zhang, R. Fickler, F. M. Miatto, Y. Yao, F. Sciarrino, E. Karimi, *Rep. Prog. Phys.* **2021**, *84*, 012402.
- [173] V. Cimini, E. Polino, M. Valeri, I. Gianani, N. Spagnolo, G. Corrielli, A. Crespi, R. Osellame, M. Barbieri, F. Sciarrino, *Phys. Rev. Appl.* **2021**, *15*, 044003.
- [174] A. Lumino, E. Polino, A. S. Rab, G. Milani, N. Spagnolo, N. Wiebe, F. Sciarrino, *Phys. Rev. Appl.* **2018**, *10*, 044033.
- [175] A. Hentschel, B. C. Sanders, *Phys. Rev. Lett.* **2011**, *107*, 233601.
- [176] S.-K. Liao, W.-Q. Cai, W.-Y. Liu, L. Zhang, Y. Li, J.-G. Ren, J. Yin, Q. Shen, Y. Cao, Z.-P. Li, F.-Z. Li, X.-W. Chen, L.-H. Sun, J.-J. Jia, J.-C. Wu, X.-J. Jiang, J.-F. Wang, Y.-M. Huang, Q. Wang, Y.-L. Zhou, L. Deng, T. Xi, L. Ma, T. Hu, Q. Zhang, Y.-A. Chen, N.-L. Liu, X.-B. Wang, Z.-C. Zhu, C.-Y. Lu, et al., *Nature* **2017**, *549*, 43.
- [177] G. Vallone, D. Bacco, D. Dequal, S. Gaiarin, V. Luceri, G. Bianco, P. Villoresi, *Phys. Rev. Lett.* **2015**, *115*, 040502.
- [178] F. B. Basset, M. Valeri, E. Rocca, V. Muredda, D. Poderini, J. Neuwirth, N. Spagnolo, M. B. Rota, G. Carvacho, F. Sciarrino, R. Trotta, *Sci. Adv.* **2021**, *7*, eabe6379.
- [179] J. L. O'Brien, *Science* **2007**, *318*, 1567.
- [180] S. Takeda, A. Furusawa, *APL Photonics* **2019**, *4*, 060902.
- [181] G. Carleo, M. Troyer, *Science* **2017**, *355*, 602.
- [182] E. S. Tiunov, V. V. T. (Vyborova), A. E. Ulanov, A. I. Lvovsky, A. K. Fedorov, *Optica* **2020**, *7*, 448.
- [183] M. Neugebauer, L. Fischer, A. Jäger, S. Czischek, S. Jochim, M. Weidemüller, M. Gärtner, *Phys. Rev. A* **2020**, *102*, 042604.
- [184] V. Cimini, M. Barbieri, N. Treps, M. Walschaers, V. Parigi, *Phys. Rev. Lett.* **2020**, *125*, 160504.
- [185] V. Gebhart, M. Bohmann, *Phys. Rev. Res.* **2020**, *2*, 023150.
- [186] C. You, M. A. Quiroz-Juárez, A. Lambert, N. Bhusal, C. Dong, A. Perez-Leija, A. Javaid, R. d. J. León-Montiel, O. S. Magaña-Loaiza, *Appl. Phys. Rev.* **2020**, *7*, 021404.
- [187] T. Giordani, A. Suprano, E. Polino, F. Acanfora, L. Innocenti, A. Ferraro, M. Paternostro, N. Spagnolo, F. Sciarrino, *Phys. Rev. Lett.* **2020**, *124*, 160401.
- [188] I. Agresti, N. Viggianiello, F. Flamini, N. Spagnolo, A. Crespi, R. Osellame, N. Wiebe, F. Sciarrino, *Phys. Rev. X* **2019**, *9*, 011013.
- [189] G. Wetzstein, A. Ozcan, S. Gigan, S. Fan, D. Englund, M. Soljačić, C. Denz, D. A. B. Miller, D. Psaltis, *Nature* **2020**, *588*, 39.
- [190] B. J. Shastri, A. N. Tait, T. F. de Lima, W. H. P. Pernice, H. Bhaskaran, C. D. Wright, P. R. Prucnal, *Nat. Photonics* **2021**, *15*, 102.
- [191] D. Brunner, D. Psaltis, *Nat. Photonics* **2021**, *15*, 323.
- [192] K. Goda, B. Jalali, C. Lei, G. Situ, P. Westbrook, *APL Photonics* **2020**, *5*, 070401.
- [193] D. Psaltis, N. Farhat, *Opt. Lett.* **1985**, *10*, 98.
- [194] N. H. Farhat, D. Psaltis, A. Prata, E. Paek, *Appl. Opt.* **1985**, *24*, 1469.
- [195] K. Wagner, D. Psaltis, *Appl. Opt.* **1987**, *26*, 5061.
- [196] D. Psaltis, D. Brady, X.-G. Gu, S. Lin, *Nature* **1990**, *343*, 325.
- [197] D. A. B. Miller, *Photonics Res.* **2013**, *1*, 1.
- [198] A. N. Tait, M. A. Nahmias, B. J. Shastri, P. R. Prucnal, *J. Lightwave Technol.* **2014**, *32*, 3427.
- [199] A. N. Tait, T. F. de Lima, E. Zhou, A. X. Wu, M. A. Nahmias, B. J. Shastri, P. R. Prucnal, *Sci. Rep.* **2017**, *7*, 7430.
- [200] Y. Shen, N. C. Harris, S. Skirlo, M. Prabhu, T. Baehr-Jones, M. Hochberg, X. Sun, S. Zhao, H. Larochelle, D. Englund, M. Soljačić, *Nat. Photonics* **2017**, *11*, 441.
- [201] J. M. Shainline, S. M. Buckley, R. P. Mirin, S. W. Nam, *Phys. Rev. Appl.* **2017**, *7*, 034013.
- [202] H.-T. Peng, M. A. Nahmias, T. F. de Lima, A. N. Tait, B. J. Shastri, *IEEE J. Sel. Top. Quantum Electron.* **2018**, *24*, 6101715.
- [203] X. Lin, Y. Rivenson, N. T. Yardimci, M. Veli, Y. Luo, M. Jarrahi, A. Ozcan, *Science* **2018**, *361*, 1004.
- [204] H. Jaeger, H. Haas, *Science* **2004**, *304*, 78.
- [205] W. Maass, T. Natschläger, H. Markram, *Neur. Comput.* **2002**, *14*, 2531.
- [206] D. Verstraeten, B. Schrauwen, M. D'Haene, D. Stroobandt, *Neur. Networks* **2007**, *20*, 391.
- [207] J. Dong, M. Rafayelyan, F. Krzakala, S. Gigan, *IEEE J. Sel. Top. Quantum Electron.* **2020**, *26*, 7701012.
- [208] K. Vandoorne, W. Dierckx, B. Schrauwen, D. Verstraeten, R. Baets, P. Bienstman, J. V. Campenhout, *Opt. Express* **2008**, *16*, 11182.
- [209] F. Duport, B. Schneider, A. Smerieri, M. Haelterman, S. Massar, *Opt. Express* **2012**, *20*, 22783.
- [210] Y. Paquot, F. Duport, A. Smerieri, J. Dambre, B. Schrauwen, M. Haelterman, S. Massar, *Sci. Rep.* **2012**, *2*, 287.
- [211] L. Larger, M. C. Soriano, D. Brunner, L. Appeltant, J. M. Gutierrez, L. Pesquera, C. R. Mirasso, I. Fischer, *Opt. Express* **2012**, *20*, 3241.
- [212] D. Brunner, M. C. Soriano, C. R. Mirasso, I. Fischer, *Nat. Commun.* **2013**, *4*, 1364.
- [213] K. Vandoorne, P. Mechet, T. V. Vaerenbergh, M. Fiers, G. Morthier, D. Verstraeten, B. Schrauwen, J. Dambre, P. Bienstman, *Nat. Commun.* **2014**, *5*, 3541.
- [214] L. Larger, A. Baylón-Fuentes, R. Martinenghi, V. S. Udaltsov, Y. K. Chembo, M. Jacquot, *Phys. Rev. X* **2017**, *7*, 011015.
- [215] G. V. der Sande, D. Brunner, M. C. Soriano, *Nanophotonics* **2017**, *6*, 561.
- [216] D. Brunner, M. C. Soriano, G. V. der Sande, editors, *Photonic Reservoir Computing*, De Gruyter, Berlin, **2019**.
- [217] J. Spall, X. Guo, T. D. Barrett, A. I. Lvovsky, *Opt. Lett.* **2020**, *45*, 5752.

- [218] J. Feldmann, N. Youngblood, C. D. Wright, H. Bhaskaran, W. H. P. Pernice, *Nature* **2019**, 569, 208.
- [219] X. Xu, M. Tan, B. Corcoran, J. Wu, A. Boes, T. G. Nguyen, S. T. Chu, B. E. Little, D. G. Hicks, R. Morandotti, A. Mitchell, D. J. Moss, *Nature* **2021**, 589, 44.
- [220] J. Feldmann, N. Youngblood, M. Karpov, H. Gehring, X. Li, M. Stappers, M. Le Gallo, X. Fu, A. Lukashchuk, A. S. Raja, J. Liu, C. D. Wright, A. Sebastian, T. J. Kippenberg, W. H. P. Pernice, H. Bhaskaran, *Nature* **2021**, 589, 52.
- [221] M. Rafayelyan, J. Dong, Y. Tan, F. Krzakala, S. Gigan, *Phys. Rev. X* **2020**, 10, 041037.
- [222] M. Miscuglio, Z. Hu, S. Li, J. K. George, R. Capanna, H. Dalir, P. M. Bardet, P. Gupta, V. J. Sorger, *Optica* **2020**, 7, 1812.
- [223] D. Pierangeli, G. Marcucci, C. Conti, *Phys. Rev. Lett.* **2019**, 122, 213902.
- [224] D. Pierangeli, M. Rafayelyan, C. Conti, S. Gigan, *Phys. Rev. Appl.* **2021**, 15, 034087.
- [225] G.-B. Huang, Q.-Y. Zhu, C.-K. Siew, *Neurocomputing* **2006**, 70, 489.
- [226] D. Pierangeli, G. Marcucci, C. Conti, *Photonics Res.* **2021**, 9, 1446.
- [227] U. Teğın, M. Yıldırım, İ. Oğuz, C. Moser, D. Psaltis, *Nat. Comput. Sci.* **2021**, 1, 542.
- [228] M. Schuld, I. Sinayskiy, F. Petruccione, *Contemp. Phys.* **2015**, 56, 172.
- [229] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, S. Lloyd, *Nature* **2017**, 549, 195.
- [230] C. Ciliberto, M. Herbster, A. D. Ialongo, M. Pontil, A. Rocchetto, S. Severini, L. Wossnig, *Proc. R. Soc. A: Math., Phys. Eng. Sci.* **2018**, 474, 20170551.
- [231] H.-S. Zhong, H. Wang, Y.-H. Deng, M.-C. Chen, L.-C. Peng, Y.-H. Luo, J. Qin, D. Wu, X. Ding, Y. Hu, P. Hu, X.-Y. Yang, W.-J. Zhang, H. Li, Y. Li, X. Jiang, L. Gan, G. Yang, L. You, Z. Wang, L. Li, N.-L. Liu, C.-Y. Lu, J.-W. Pan, *Science* **2020**, 370, 1460.
- [232] F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, R. Biswas, S. Boixo, F. G. S. L. Brandao, D. A. Buell, B. Burkett, Y. Chen, Z. Chen, B. Chiaro, R. Collins, W. Courtney, A. Dunsworth, E. Farhi, B. Foxen, A. Fowler, C. Gidney, M. Giustina, R. Graff, K. Guerin, S. Habegger, M. P. Harrigan, M. J. Hartmann, A. Ho, M. Hoffmann, T. Huang, et al., *Nature* **2019**, 574, 505.
- [233] A. Rocchetto, S. Aaronson, S. Severini, G. Carvacho, D. Poderini, I. Agresti, M. Bentivegna, F. Sciarrino, *Sci. Adv.* **2019**, 5, eaau194.
- [234] X.-D. Cai, D. Wu, Z.-E. Su, M.-C. Chen, X.-L. Wang, L. Li, N.-L. Liu, C.-Y. Lu, J.-W. Pan, *Phys. Rev. Lett.* **2015**, 114, 110504.
- [235] G. R. Steinbrecher, J. P. Olson, D. Englund, J. Carolan, *npj Quantum Inf.* **2019**, 5.
- [236] N. Killoran, T. R. Bromley, J. M. Arrazola, M. Schuld, N. Quesada, S. Lloyd, *Phys. Rev. Res.* **2019**, 1, 033063.
- [237] P. L. McMahon, A. Marandi, Y. Haribara, R. Hamerly, C. Langrock, S. Tamate, T. Inagaki, H. Takesue, S. Utsunomiya, K. Aihara, R. L. Byer, M. M. Fejer, H. Mabuchi, Y. Yamamoto, *Science* **2016**, 354, 614.



Federico Vernuccio is currently a Ph.D. candidate at the Physics Department, Politecnico di Milano, where he achieved his master degree in photonics and nanooptics in 2019. His research is mainly focused on broadband coherent Raman scattering microscopy and on artificial intelligence with the final purpose of developing an intelligent microscope for applications in biomedicine.



Arianna Bresci is currently a Ph.D. candidate in physics at Politecnico di Milano, where she achieved her master degree in biomedical engineering in 2020. Her research focuses on multimodal vibrational and multiphoton nonlinear optical micro-spectroscopy supported by artificial intelligence for applications in assisted diagnosis and intelligent biomedicine.



Valeria Cimini received her master degree in physics at La Sapienza Università di Roma in 2016. She did her Ph.D. at the University of Roma Tre in the New Quantum Optics group under the supervision of Prof. Barbieri. She is currently a postdoctoral researcher in Prof. Sciarrino's group at La Sapienza working on quantum optics and quantum metrology projects.



Alessandro Giuseppe is an assistant professor (RTDa) at the University of Rome "La Sapienza" where he received his master degree in control engineering and his Ph.D. degree in automatica respectively in 2016 and 2019. Since 2016, he has participated in 7 EU and national research projects in the field of automatic control and artificial intelligence. Since 2020 he is serving as associate editor for the International Journal of Control, Automation and Systems. His main research activities are in the fields of network control and intelligent systems, where he has published about 50 papers in international journals and conferences.



Giulio Cerullo is a full professor with the Physics Department, Politecnico di Milano, where he leads the Ultrafast Optical Spectroscopy Laboratory. His research activity covers on the one hand the generation and manipulation of ultrashort light pulses, and on the other hand the use of such pulses to capture the dynamics of ultrafast events in molecular and solid-state systems. He has published over 450 papers which have received >> 23 000 citations. He is a fellow of the Optical Society of America and of the European Physical Society and Chair of the Quantum Electronics and Optics Division of the European Physical Society.



Dario Polli is associate professor of physics. He leads the "VIBRA" Nonlinear Microscopy Laboratory within Politecnico di Milano in Italy. He published more than 100 scientific articles on coherent Raman spectroscopy and microscopy, ultrafast and nonlinear optics, Fourier-transform spectroscopy, time-resolved pump-probe spectroscopy and microscopy. He has been the coordinator of several national and international research projects, including an ERC consolidator grant, two ERC proof-of-concept grants, and a Horizon-2020 ICT collaborative project. He has filed four patents related to photonics. He is the co-founder of two startups in the field of spectroscopy. He is passionate about science divulgation to the public.



Carlo M. Valensise is postdoctoral researcher at Centro Ricerche Enrico Fermi in Rome. He got a Ph.D. in physics in 2021 at Politecnico di Milano, developing applications of deep and reinforcement learning for nonlinear spectroscopy. His research interests include photonics and data science.