Data transmission in a multimode optical fiber using a neural network

Tom A. Kuusela*

Department of Physics and Astronomy,

University of Turku, 20014 Turku, Finland

(Dated: August 16, 2022)

Abstract

In digital data transmission, single mode optical fibers are commonly used since they can carry very short optical pulses without any significant distortions. In contrast, multimode fibers support many propagation modes that travel with different speeds, thus they cannot maintain the shape of a light pulse. This feature of multiple propagation modes can be a benefit since it makes possible the transmission of data through several channels simultaneously. We demonstrate how multimode fibers can be used to transmit images. Because of the different propagation constants of the modes, the transmitted image is scrambled to apparently random speckle patterns. A simple neural network can be used to model the transmission through the multimode fiber. We show how the neural network can be trained to recognize a set of patterns with high accuracy.

I. INTRODUCTION

Optical fibers are widely used in data transmission since they are able to carry information at higher rates and to longer distances than copper wires. In telecommunications, optical fibers are commonly single mode fibers (SMF) because of their capability to transmit very short optical pulses without any distortions. However, single mode fibers can carry information only in serial form since they consist of a single channel. Ever increasing needs to expand telecommunication networks and data handling capacities have encouraged the consideration of data transmission through parallel channels in multimode fibers (MMF). These fibers can support up to thousands optical modes, each of which (at least in theory) can be independently used to carry information.

In MMF endoscopy and other image transfer applications, the fiber modes can be used to transmit the pixel data of the image. Unfortunately, this is not straightforward: an image projected onto the proximal face of the fiber is distorted at the distal face, where the different propagation constants of the fiber modes have produce a distorted image that appears to be a random speckle pattern.

There are several methods of modeling the image transmission through MMFs. For example, the distortion due to modal dispersion can be undone by phase conjugation methods.[1–3] Alternately, in the matrix method, the map between the inputs and output amplitude and phase is experimentally measured for each input pattern.[4–6] These methods are not very practical since they require an external reference beam at the output in order to extract the complex field.

In the simplest system, only the intensity of the fiber output is measured by a camera. Although there are some attempts to construct the complete transmission matrix of the fiber from the intensity data only using convex optimization,[7] the most promising approach is to apply neural networks, which are taught to recognize a set of images. In the first works on this field, the neural networks were simple three-layer perceptron with one hidden layer.[8,9] Later, convolutional neural networks with tens of various layers and deep learning strategies have been used.[10,11]

In this paper, we show how a simple two-layer perceptron can be implemented to recognize letter images transmitted through a short multimode fiber. In Sec. II, we introduce the basic concepts of optical fibers and show how the different modes can be solved. The fundamentals

of the simplest neural networks are briefly presented in Sec. III. In Sec. IV, two different sets of experiments are explained and their properties explored. Concluding remarks are given in Sec. V.

II. MULTIMODE FIBERS

In any homogeneous transparent medium, an optical wave is described by the *wave function* of position $\mathbf{r}$ and time t, denoted $U = U(\mathbf{r}, t)$, which satisfies the wave equation

$$\nabla^2 U - \frac{1}{c^2}\frac{\partial^2 U}{\partial t^2} = 0, \tag{1}$$

where $c = c_0/n$, c_0 is the speed of light in vacuum, and n is the index of refraction of the medium. The function U represents any component (x, y, z) of the optical wave's electric or magnetic field. Substituting $U(\mathbf{r},\mathbf{t}) = U(\mathbf{r})e^{i\omega t}$, where $U(\mathbf{r})$ is the complex amplitude of the wave and ω is its angular frequency, into Eq. (1) leads to the *Helmholtz equation*

$$\nabla^2 U(\mathbf{r}) + n^2 k_0^2 U(\mathbf{r}) = 0, \tag{2}$$

where the vacuum wavenumber $k_0 = \omega/c_0 = 2\pi/\lambda_0$ and λ_0 is the vacuum wavelength.

An optical fiber is a cylindrical dielectric waveguide made of silica glass.[12,13] The fiber has a central *core* where the light is guided, embedded in an outer *cladding*. In general, the refractive index of the fiber is a function $n(r)$ of the radial position r. In the *step-index fiber*, $n(r) = n_1$ in the core ($r < a$) and $n(r) = n_2$ in the cladding ($r > a$), where n_1 and n_2 are constants. In the following, we assume that the radius of the cladding is so large that it can be taken to be infinite.

Each component of the monochromatic electric field obeys Eq. (2). In a cylindrical coordinate system (r,ϕ,z), this equation is written as

$$\frac{\partial^2 U}{\partial r^2} + \frac{1}{r}\frac{\partial U}{\partial r} + \frac{1}{r^2}\frac{\partial^2 U}{\partial \phi^2} + \frac{\partial^2 U}{\partial z^2} + n^2 k_0^2 U = 0, \tag{3}$$

where $U = U(r, \phi, z)$. Light waves in the fiber are traveling in the z-direction with the propagation constant β, where the z-dependence of U is of the form $e^{-i\beta z}$. In addition, because of the circular symmetry of the waveguide, each field component must not change when the coordinate ϕ is increased by 2π. We thus assume that the angular solution takes

the harmonic form $e^{-il\phi}$, where l is an integer. By substituting $U(r,\phi,z) = u(r)e^{-il\phi}e^{-i\beta z}$ into Eq. (3) we obtain

$$\frac{\partial^2 u}{\partial r^2} + \frac{1}{r}\frac{\partial u}{\partial r} + \left(n^2(r)k_0^2 - \beta^2 - \frac{l^2}{r^2}\right)u = 0, \quad l = 0, \pm 1, \pm 2, ... \tag{4}$$

In the case of the step-index fiber, Eq. (4) should be solved separately in the regions $r < a$ and $r > a$. Excluding solutions that go to infinity at $r = 0$ in the core or at $r \to \infty$ in the cladding, we obtain Bessel functions of the first kind and order l in the core (oscillating functions with decaying amplitude), and modified Bessel functions of the second kind and order l in the cladding (decaying exponentially at large x).

In order to solve for the unknown propagation constant β, the boundary conditions should be considered. Most practical fibers are weakly guided ($n_1 \approx n_2$) and all rays are paraxial, i.e., approximately parallel to the fiber axis.[12] The longitudinal components of the electric and magnetic fields are then much weaker than the transverse components and we can obtain only transverse electromagnetic (TEM) waves. In this approximation, the propagation constant (and corresponding solutions) can be found by demanding that the scalar function $u(r)$ and its derivative are continuous at the boundary $r = a$. It turns out that, for each index value l, there are only certain possible values of β. We can thus index these solutions, which are called *modes*, as $u_{lm}(r)$. Each of these modes has a distinct propagation constant β_{lm} and a characteristic field distribution in the transverse plane. It should be noted that β_{lm} depends on the n_1, n_2, and λ_0. Further, it can be shown that the number of modes M is

$$M \approx \frac{4}{\pi^2}V^2, \tag{5}$$

where $V \equiv 2\pi\frac{a}{\lambda_0}\mathrm{NA}$ is the so-called fiber parameter and $\mathrm{NA}=\sqrt{n_1^2 - n_2^2}$ the numerical aperture of the fiber. For typical step-index fibers, the number of modes can be several thousands.

The complete solutions of the transverse electric field are now

$$E_{lm}^T(r,\phi,z) = u_{lm}(r)e^{-i(l\phi+\beta_{lm}z)}. \tag{6}$$

Since all modes of a fiber are known, the propagation of a monochromatic beam with an arbitrary field distribution along the fiber can be calculated. Any initial field satisfying the boundary conditions at the input of the fiber can be decomposed into fiber modes and

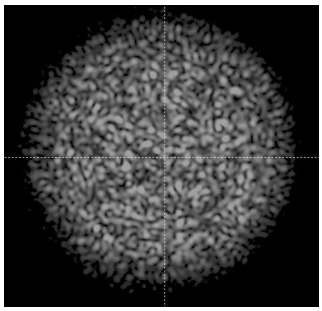


FIG. 1. Speckle pattern measured at the end of the step-index fiber.

presented as a linear combination

$$E_0(r,\phi,0) = \sum_{l,m} \alpha_{lm} E^T_{lm}(r,\phi,0), \tag{7}$$

where α_{lm} are complex coefficients. The mode solutions [Eq. (6)] give the field at the distal face of the fiber. In practical situations, many modes with various initial phases are launched into the fiber. Further, each mode propagates with a different velocity. As a result, light emerging from the far end of the fiber will be a combination of a number waves that differ from each other in phase and field distribution. At any point on the fiber end, these waves may add or cancel due to interference and produce complex speckle patterns. An example of the intensity $I = |E|^2$ pattern measured at the end of the fiber is shown in Fig. 1. Although the output intensity distribution can seem quite random, it should be noted that this intensity pattern is solely determined by the input field. However, in practice, it is quite difficult to calculate the output field because typically the input field is not well known. Hence, instead of an analytical model, in our application we use a neural network that is taught to model the transmission properties of the fiber.

III. NEURAL NETWORKS

Artificial neural networks, often simply called *neural networks*, are computing systems that mimic the functions of biological neural systems.[14] A neural network consists of artificial neurons that model, in a very simple form, the neurons of biological brains. An artificial neuron is a mathematical function having one or more inputs and one output. The output

is a linear or nonlinear sum of the inputs. Usually the inputs are separately weighted before summing and, during the teaching process of the neural network, optimal values of the weights are searched. By connecting the outputs of the artificial neurons to inputs of other neurons, complex neural networks can be built. In the machine learning, neural networks are used in various pattern recognition, data classification, and regression tasks.[15–17]

The *perceptron* is the simplest form of neural networks.[18] It has many inputs and one output, which is a weighted sum of the inputs. In many practical cases, perceptrons are used in parallel (see Fig. 2). It has inputs x_j, $j = 1, ..., d$ and outputs y_i, $i = 1, ..., K$. The K outputs are

$$y_i = \sum_{j=1}^{N} w_{ij} x_j + w_{i0}, \tag{8}$$

where w_{ij} is the $K \times (d + 1)$ weight matrix. The terms w_{i0} are intercept values to make the model more general. They can be interpreted as an extra bias unit x_0, which is always set to +1. In our applications, inputs x_j are the pixel values of the speckle pattern and the outputs are used to recognize or classify these patterns.

The training of the perceptron (or, in general, any neural network) means that we try to find the optimal values of the weights w_{ij} that minimize a given error or cost function. An error function measures how far the outputs differ from specified target values. A commonly used error function is the sum of squared errors

$$E(w) = \frac{1}{2} \sum_{t} (r^{(t)} - y^{(t)}(w))^2, \tag{9}$$

where $r^{(t)}$ are the target values indexed by t. The target values with the corresponding input values makes our training set of the perceptrons. Next we should select the method to update the weights. The *gradient descent* method is a simple yet useful optimization algorithm that is often used in machine learning to find the local minimum of linear systems. In this approach, we climb down the slope of the local or global minimum of E.[19] During the iteration process, the step is taken in the direction of the negative gradient of E. This guarantees that each change in weighting factor drives the error closer to the minimum. The step size is determined by the value of the learning factor η as well as the slope of the gradient. At the minimum, the derivative is zero and the procedure terminates.[20] Thus, we

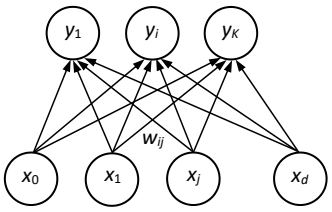


FIG. 2. K parallel perceptrons where x_j, $j = 0, ..., d$ are the inputs and y_i, $i = 1, ..., K$ are the outputs. The bias unit $x_0 = 1$.

obtain the change of the weight w_{ij}

$$\begin{aligned}
\Delta w_{ij} &= -\eta \frac{\partial E}{\partial w_{ij}} = -\frac{1}{2}\eta \frac{\partial}{\partial w_{ij}} \sum_t (r_i^{(t)} - y_i^{(t)})^2 \\
&= -\frac{1}{2}\eta \sum_t \frac{\partial}{\partial w_{ij}} (r_i^{(t)} - y_i^{(t)})^2 \\
&= -\eta \sum_t (r_i^{(t)} - y_i^{(t)}) \frac{\partial}{\partial w_{ij}} (r_i^{(t)} - y_i^{(t)}) \\
&= -\eta \sum_t (r_i^{(t)} - y_i^{(t)}) \frac{\partial}{\partial w_{ij}} \left(r_i^{(t)} - \sum_{j=1}^{N} w_{ij} x_j^{(t)} + w_{i0} \right) \\
&= \eta \sum_t (r_i^{(t)} - y_i^{(t)}) x_j^{(t)}.
\end{aligned} \tag{10}$$

The error term $(r_i^{(t)} - y_i^{(t)})$ measures how far we are from the target value. For example, if the output is less than the desired output, the update Δw_{ij} is positive if the input $x_j^{(t)}$ is positive and negative if the input is negative, thus after update the error is decreased. The magnitude of the update depends also on the input: if $x_j^{(t)}$ is very small its contribution on the weighted sum (8) is also very small, therefore it is not wise to allow large changes in this direction. After each step, we set $w_{ij} = w_{ij} + \Delta w_{ij}$ and repeat the process until the error function E is small enough (see Sec. IV). The learning factor η is somewhat critical: if too large, the process may never converge, while if very small, extremely many updates are needed and the convergence process is very slow. Before the training procedure, the weights are set to small random values.

A perceptron can only approximate the linear functions of the inputs. By adding a nonlinear intermediate or *hidden layer*, this multilayer perceptron can model nonlinear functions.[18] The input x_i is fed to the input layer which act as an input of the second hidden layers (see

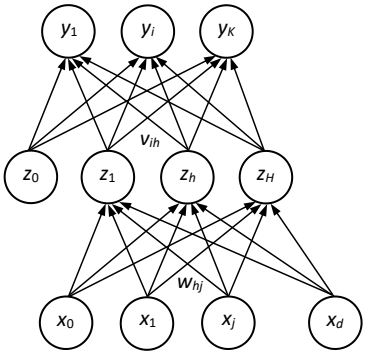


FIG. 3. Neural network with one hidden layer.

Fig. 3). Each hidden unit is a perceptron by itself, and their output is the *nonlinear* weighted sum

$$z_h = \frac{1}{1+\exp\left[-\left(\sum_{j=1}^{d} w_{hj}x_j + w_{h0}\right)\right]}, \quad h = 1, ..., H. \tag{11}$$

The output function $f(x) = 1/(1+e^{-x})$ is just one possible choice for the sigmoid (S-shaped) function acting as a continuous, differentiable version of thresholding function that ranges from 0 to +1 (or from −1 to +1 as $\tanh(x)$). The outputs y_i are again normal perceptrons taking the hidden units as inputs

$$y_i = \sum_{h=1}^{H} v_{ih}z_h + v_{i0}, \tag{12}$$

where v_{ih} are the weights of the hidden layer. Again we can use the same error function as used previously. Because of the hidden layer, the updating of weights w_{hj} and v_{ih} is more complicated (see Appendix A).

IV. EXPERIMENTS

A. Experimental setup

The experimental setup is shown in Fig. 4. As a light source, we use a He-Ne laser (Uniphase 1507P, the power 0.5 mW). The output power of the laser is reduced by the neutral density filter because the intensity is far too high for the camera, even if very short

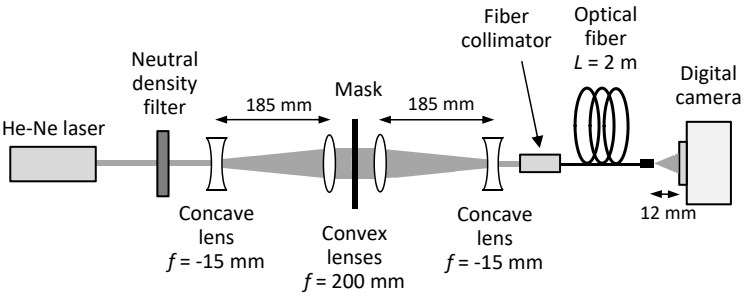


FIG. 4. Experimental setup.

shutter times are used. Since the beam diameter of the laser is quite small, about 1 mm, we use the combination of the concave and convex lens to increase the collimated beam diameter to 8 mm. A larger beam diameter makes it easier to implement masks, which are used to produce various input fields for the optical fiber. In the first experiments, we use thin microscopy cover glasses, where the letters (height about 4 mm) are handwritten in permanent ink. In the second experiments, the mask is an opaque plastic sheet with four small holes in a rectangular arrangement. After the mask, the light beam is converted back to the collimated beam with the diameter of 1 mm. This beam is coupled to the optical fiber with a fiber collimator (Thorlabs F220FC-B). As a multimode fiber, we use a step-index fiber (Thorlabs M43LO2, core diameter 105μm, NA 0.22) of the length of 2 m. According to Eq. (5), we can estimate that, at the laser wavelength of 632 nm, this fiber supports about 5300 different modes.

At the output of the fiber, we did not use any lenses, but projected the speckle pattern directly on the surface of the camera chip. The distance from the end of the fiber is adjusted in such a manner that the speckle pattern covers as large area as possible. The digital monochrome camera (IDS UI-1242L) has a CMOS sensor of 1280 $\times$ 1024 pixels (with area 6.784 mm $\times$ 5.427 mm). Before any data analysis, we crop the pictures to the size of 1024 $\times$ 1024 pixels. The shutter time should be optimized in order to fully utilize the dynamical range of the camera chip. The laser should be allowed to warm up long enough to obtain stability (1-2 hours), since even minor changes in the laser wavelength (or in the wavelength distribution) can significantly affect the speckle pattern. Likewise, even modest strain on the fiber or temperature variation will change the refractive index of the fiber and

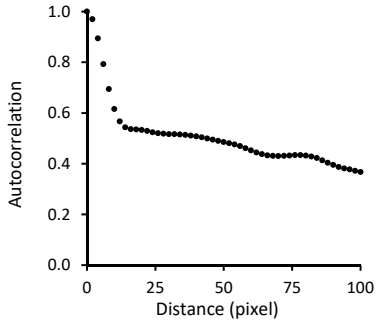


FIG. 5. Autocorrelation of the pixel data as a function of the distance.

alter mutual phases of the modes producing dramatic changes in speckles.

B. Training of the neural network with the letter masks

Before we can use our neural network to model the fiber transmission, it is necessary to reduce the amount of data. In principle, it would be possible to use the data from every one of the more than one million camera pixels but this approach would not be very practical. Visual inspection of the speckle patterns reveals that the speckles have a certain average size. To study this in more detail, we calculate the autocorrelation function of the pixel data:

$$C(\delta) = \frac{\langle X_{ij} X_{i+\delta,j+\delta} \rangle}{\langle X_{ij} \rangle^2}, \tag{13}$$

where X_{ij} is the pixel data, δ the pixel distance and $\langle \ldots \rangle$ means the average over all pixels. The autocorrelation as a function of the distance is shown in Fig. 5, where we observe that the autocorrelation drops rapidly up to the distance of about 10 pixels and at longer distances the autocorrelation does not change much. We interpret that the average size of speckles corresponds to the quick drop in the autocorrelation. Therefore, we first averaged all images over 8×8 pixel rectangles, producing images of the size 128×128 pixels. Visually this averaged image looks very similar as the original one, so obviously no essential information is lost. This averaged image is further reduced by cropping the central area of 32×32 pixels. Visual inspection of the images revealed that most significant changes are in this region. Finally, we have image data of 1024 pixels, which are formed as an input vector x_j of the neural network.

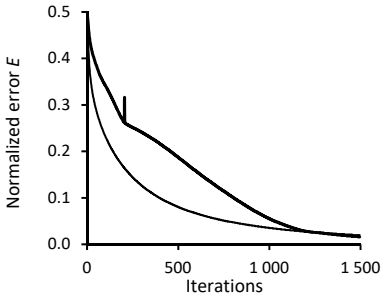


FIG. 6. Normalized error as a function of iterations of the perceptron type of neural network (thin line) and nonlinear neural network (thick line). The significant spike in the thick line means that the error has temporarily increased but the adaptive learning factor has decreased and in further iterations the error continues decreasing.

The pixels intensity values are in the range of $0 \ldots 255$. Usually, it is a good idea to normalize the input values so that they are all centered around 0 and have the same scale. We use the fixed (not individual) scale by first subtracting the value of 128 from the pixel value and then dividing it with 128. The initial value of all weights is set on a random value between -0.01 and $+0.01$. The learning factor η is either fixed or adaptive. In the latter case, the error function is calculated after each iteration step, and if error increases the learning factor is decreased by the factor of 0.75.

In the first set of experiments, we use 15 masks with hand-written letters from A to O and one blank frame, totally 16 frames. We code each letter by a binary number $0000, 0001, \ldots, 1111$ (named as DCBA), so we have 4 outputs y_i. The error function normalized by the initial error (the error without any iterations) summed over all outputs is shown in Fig. 6 using two different structures: the simple perceptrons with the fixed learning factor of 0.0002 and the nonlinear one with 20 hidden units and adaptive learning factor. At first, we find that with the linear model the error decreases more rapidly than with the nonlinear model, but later the difference is not significant. However, if very small final error is needed, the nonlinear model reaches the level with fewer iterations than the linear one. The normalized error of 0.001 is reached at about 6000 iterations (linear model) and 3500 (nonlinear model). At this error level the maximum deviation from the target values (0 or 1) is ± 0.05 with some letter masks, but in most cases it is much smaller. We conclude that

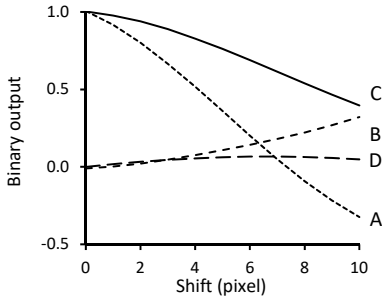


FIG. 7. The binary outputs of the letter E (the target 0101) as a function of the frame shift.

both the linear and nonlinear neural network can be easily trained to recognize 16 letter figures with high accuracy. This is a remarkable result since visible differences in the speckle patterns of different letter masks are surprisingly small.

In practical applications, it is important to check how sensitive the neural network is to various disturbances. First, we digitally displaced the original camera image (before any averaging and cropping) in the example of the letter E (the corresponding target 0101) in the horizontal direction (orthogonal to the laser beam axis) and results are shown in Fig. 7. Clearly we can no longer recognize this letter correctly (for example, by setting the threshold value of 0.5 for each bit) if the displacement is larger than 4 pixels, about 20 μm. This is half of the average size of the speckle pattern based on the spatial correlation analysis. We can see that the neural network modeling is highly sensitive to this kind of pattern displacement at the camera end.

In the next experiment, we moved the mask horizontally (orthogonal to the laser beam axis), by use of the translation stage upon which the mask is installed. Again, we use the letter E as an example. The mask is moved a total of ± 1.75 mm, but in the training phase of all letters, the mask at the position of 0 mm (i.e., the letter is approximately in the center of the expanded laser beam) is used. Surprisingly, this letter can be recognized correctly when the shift is in the range -1 to $+0.5$ mm, 50 times more than the allowed displacement in the camera position. Obviously moving the mask inside the collimating beam does not dramatically change the input field or rather the composition of the field modes launched into the fiber. The actual mode composition of the real mask is very difficult to calculate. In general, the modes form a spatial function base: with low values of indexes l and m the

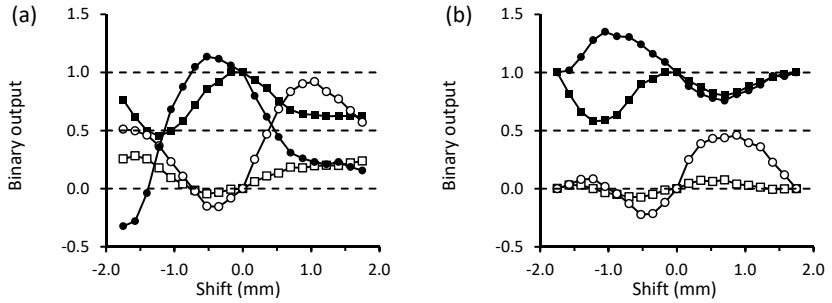


FIG. 8. The binary outputs of the letter E (the target 0101) as a function of the mask shift. The bit A - solid circle, B - open circle, C - solid rectangle, D - open rectangle. (a) Only one mask used in training of this letter positioned at the shift of 0 mm (b) Three masks used in the training of this letter positioned at the shifts -1.75, 0 and $+1.75$ mm.

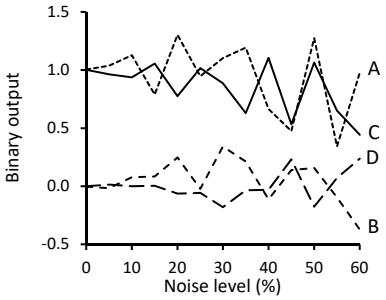


FIG. 9. The binary outputs of the letter E (the target 0101) as a function of the noise level.

oscillations are slow, with high index values oscillations are rapid. Since speckle patterns in our application are very complex images, mainly fast components dominates. We assume that slow components are mostly responsible on the mask position and the fast components on the details of the mask shape.

We can even increase the allowed range of the mask displacement by using the additional mask positions of -1.75 and $+1.75$ mm in the training phase of the letter E. In this case, as seen in Fig. 8(b), this letter can be recognized correctly over the whole range. In principle, it would be possible to train our model with shifts in other directions and for all letters, but final performance could be difficult to predict since some interaction between different patterns could arise.

Another important feature of our neural network modeling is the tolerance against noisy inputs. We added white noise to each pixel of the averaged and cropped picture and tested how well the letters can be recognized. The results of the example letter E are shown in Fig. 9. The noise level is the percentage of the maximum value 255 of the pixel. We can see that even 50–60% additional noise can not prevent correct recognition of the letter. This insensitivity against the noise can be explained by the fact that each output depends on very many inputs and typically all weights are small. Therefore, even large changes in few inputs due to the noise do not affect much on the output value.

C. Training binary masks

In the second set of experiments, we used the mask with four holes in a rectangular arrangement. We named the holes as A, B, C and D. First, we took four images when only one of these holes was open and all the rest were blocked with a small piece of an opaque tape. Next, we taught our neural network with only the blank frame and these four images with corresponding binary targets 0000, 0001, 0010, 0100 and 1000 using both the simple perceptron and nonlinear neural network with 20 hidden units. With this trained neural network, we tested how well all other 11 combinations of the open hole patterns can be recognized. The results are shown in Fig. 10. With the patterns used in the teaching phase, the recognition is, of course, perfect. We obtain perfect recognition also with all the rest of the patterns if we put the threshold value of 0.5 for each bit. The most critical case is the pattern 1111. There were no significant differences in results between the perceptron and nonlinear model.

At first, the results of Fig. 10 seem to be quite surprising since for example the output *intensity distribution* with two open mask holes is not the sum of output intensity distributions of the single open mask holes: we cannot sum intensities but we can sum electric fields. Let us have two input fields $E_0^{(1)}(r,\phi,0)$ and $E_0^{(2)}(r,\phi,0)$ corresponding to these two mask holes. At the distal face of the fiber we have $E = E^{(1)}(r,\phi,L) + E^{(2)}(r,\phi,L)$, L is the length of the fiber. Now the corresponding intensity is $I = |E|^2 = |E^{(1)} + E^{(2)}|^2 = |E^{(1)}|^2 + |E^{(2)}|^2 + E^{(1)}E^{(2)*} + E^{(1)*}E^{(2)} = I_1 + I_2 + I_{12}$. The neural network can, of course, recognize the intensity patterns I_1 and I_2 very well but clearly it efficiently ignores the interference term I_{12}. Both $E^{(1)}$ and $E^{(2)}$ consist of many modes but with mutually different

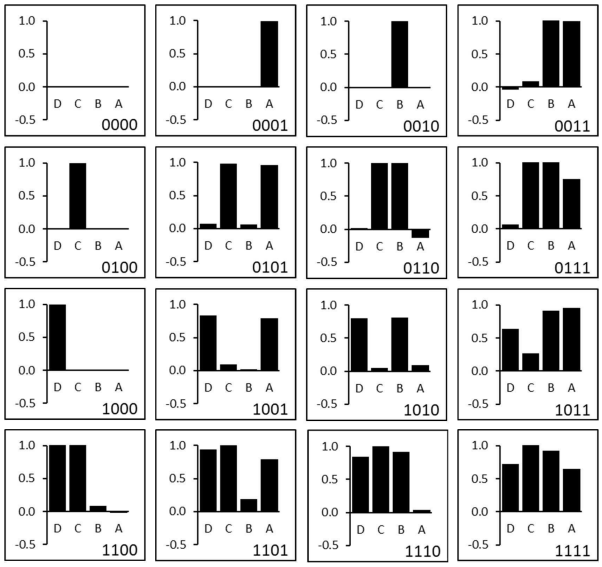


FIG. 10. The binary outputs of the perceptron when the input mask consists of the different combinations of open holes. The cases 0000, 0001, 0010, 0100 and 1000 were used in the training.

(and almost random) phases, thus their interference produces such an intensity distribution which appears as a noise in I_1 and I_2. As we know about previous experiments, the pattern recognition is rather tolerant against noise. As we can see in Fig. 10, this "noise" increases when several holes are open and the accurate pattern recognition is more difficult.

V. CONCLUDING REMARKS

We have shown how a neural network can be trained to recognize letter patterns transmitted through a multimode fiber. It should be noted that the neural network actually models the whole optical path including the light source, all lenses, the masks, the fiber and even the camera. The neural network greatly simplifies the system since there is no need to know any details or physical properties of the optical components. On the other hand, usually the pattern recognition works only with those patterns that have been used in the training set. The more advanced deeply hierarchical neural network structures can recognize the basic graphical elements of the images, such as lines, circles, arcs, and their combinations, and

therefore they are more capable of "understanding" images that are not in the training set. On the other hand, while neural networks work nicely here, it may be difficult to understand why they are so good in this kind of task, since all important information produced in the training phase is spread in the numerous weights. Even in our simple experiments, there are features that are not so obvious. The reconstruction of the background rules from the neural network weights is still a mostly unsolved problem.

Finally, our basic setup can also be used for other purposes than pattern recognition, classification, or image reconstruction. As speckle patterns are highly sensitive to fiber movements, the fiber acts like a microphone or vibration sensor. Since the refractive indices of the fiber depend on temperature, the fiber can be used as a thermometer. As the speckles are produced by interference of the different modes, these patterns are sensitive to the source wavelength, leading to fiber-based spectrometer, which can achieve picometer resolution.[21,22]

Appendix A: Neural network with a hidden layer.

Since the second layer is a perceptron with the inputs z_h we already know how to update the weights v_{ih} as

$$\Delta v_{ih} = \eta \sum_t (r_i^{(t)} - y_i^{(t)}) z_h^{(t)}. \tag{A1}$$

The first layer also consists of perceptrons but the problem is that we do not know the desired values of the outputs of the hidden units. Therefore, we use the derivative chain rule in order to calculate the gradient

$$\frac{\partial E}{\partial w_{hj}} = \frac{\partial E}{\partial y_j} \frac{\partial y_j}{\partial z_h} \frac{\partial z_h}{\partial w_{hj}}. \tag{A2}$$

Now we can write

$$\begin{aligned} \Delta w_{hj} &= -\eta \frac{\partial E}{\partial w_{hj}} = -\eta \frac{1}{2} \frac{\partial}{\partial w_{hj}} \sum_t \sum_i (r_i^{(t)} - y_i^{(t)})^2 \\ &= \eta \sum_t \left[\sum_i (r_i^{(t)} - y_i^{(t)}) v_{ih} \right] z_h^{(t)} (1 - z_h^{(t)}) x_j^{(t)} \end{aligned}. \tag{A3}$$

The error term $\sum_i (r_i^{(t)} - y_i^{(t)}) v_{ih}$ is the *backpropagated* error of the hidden unit h produced by all output units. As with the simple perceptron, each error term is weighted by the responsibility v_{ih} of the hidden unit. The term $z_h^{(t)}(1 - z_h^{(t)})$ is the derivative of the nonlinear function (11). After each step we update the weights $v_{ih} = v_{ih} + \Delta v_{ih}$ and $w_{hj} = w_{hj} + \Delta w_{hj}$.

Its is important that v_{ih} is updated *after* the update of w_{hj}, i.e., we use the *old* values of v_{ih} in Eq. (A3).

---

* tom.kuusela@utu.fi

[1] A. Yariv, "On transmission and recovery of three-dimensional image information in optical waveguides," J. Opt. Soc. Am. **66**, 301–306 (1976).

[2] A. Gover, C. P. Lee, and A. Yariv, "Direct transmission of pictorial information in multimode optical fibers," J. Opt. Soc. Am. **66**, 306–311 (1976).

[3] G. J Dunning and R. C. Lind, "Demonstration of image transmission through fibers by optical phase conjugation," Opt. Lett. **7**, 558–560 (1982).

[4] Y. Choi, C. Yoon, M. Kim, T. D. Yang, C. Fang-Yen, R. R. Dasari, K. J. Lee, and W. Choi, "Scanner-free and wide-field endoscopic imaging by using a single multimode optical fiber," Phys. Rev. Lett. **109**, 203901-1–5 (2012).

[5] R. Y. Gu, R. N. Mahalati, and J. M. Kahn, "Design of flexible multi-mode fiber endoscope," Opt. Express **23**, 23845–23858 (2015).

[6] D. Loterie, S. Farahi, I. Papadopoulos, A. Goy, D.Psaltis, and C.Moser, "Digital confocal microscopy through a multimode fiber," Opt. Express **23**, 26905–26918 (2015).

[7] S. Popoff, G. Lerosey, M. Fink, A. C. Boccara, and S. Gigan, "Image transmission through an opaque material," Nat. Commun. **1**, 81 (2010).

[8] S. Aisawa, K. Noguchi, and T. Matsumoto,"Remote image classification through multimode optical fiber using a neural network," Opt. Lett. **16**, 645–647 (1991).

[9] R. K. Marusarz and M. R. Sayeh,"Neural network-based multimode fiber-optic information transmission," Appl. Opt. **40**, 219–227 (2001).

[10] B. Rahmani, D. Loterie, G. Konstantinou, D. Psaltis, and C. Moser,"Multimode optical fiber transmission with a deep learning network," Light Sci. Appl. **7**, 69–79 (2018).

[11] N. Borhani, E. Kakkava, C. Moser, and D. Psaltis,"Learning to see through multimode fibers," Optica **5**, 960–966 (2018).

[12] B. E. A. Saleh and M. C. Teich, *Fundamentals of Photonics 2nd Edition* (Wiley, New Jersey, 2007).

[13] I. R. Kenyon, *The Light Fantastic: A Modern Introduction to Classical and Quantum Optics* (Oxford University Press, New York, 2011).

[14] J. A. Hertz and R. G. Palmer, *Introduction to the theory of neural computing* (Addison-Wesley, Reading, 1991).

[15] C. M. Bishop, *Neural networks for pattern recognition* (Oxford University Press, Oxford, 1995).

[16] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern classification* (Wiley, New York, 2001).

[17] J. Han and M. Kammer, *Data mining: concepts and technques* (Morgan Kaufmann, San Francisco, 2011).

[18] E. Alpaydim, *Introduction to Machine Learning Fourth Edition* (The MIT Press, Cambridge, 2020).

[19] The function $E(w)$ is highly complicated, thus we hardly ever find the global minimum but some of the many local ones. In practice, this means that with random initial values of w_j we finally get different set of w_j. However, this is not seriously problem as long as the final error less than the target value.

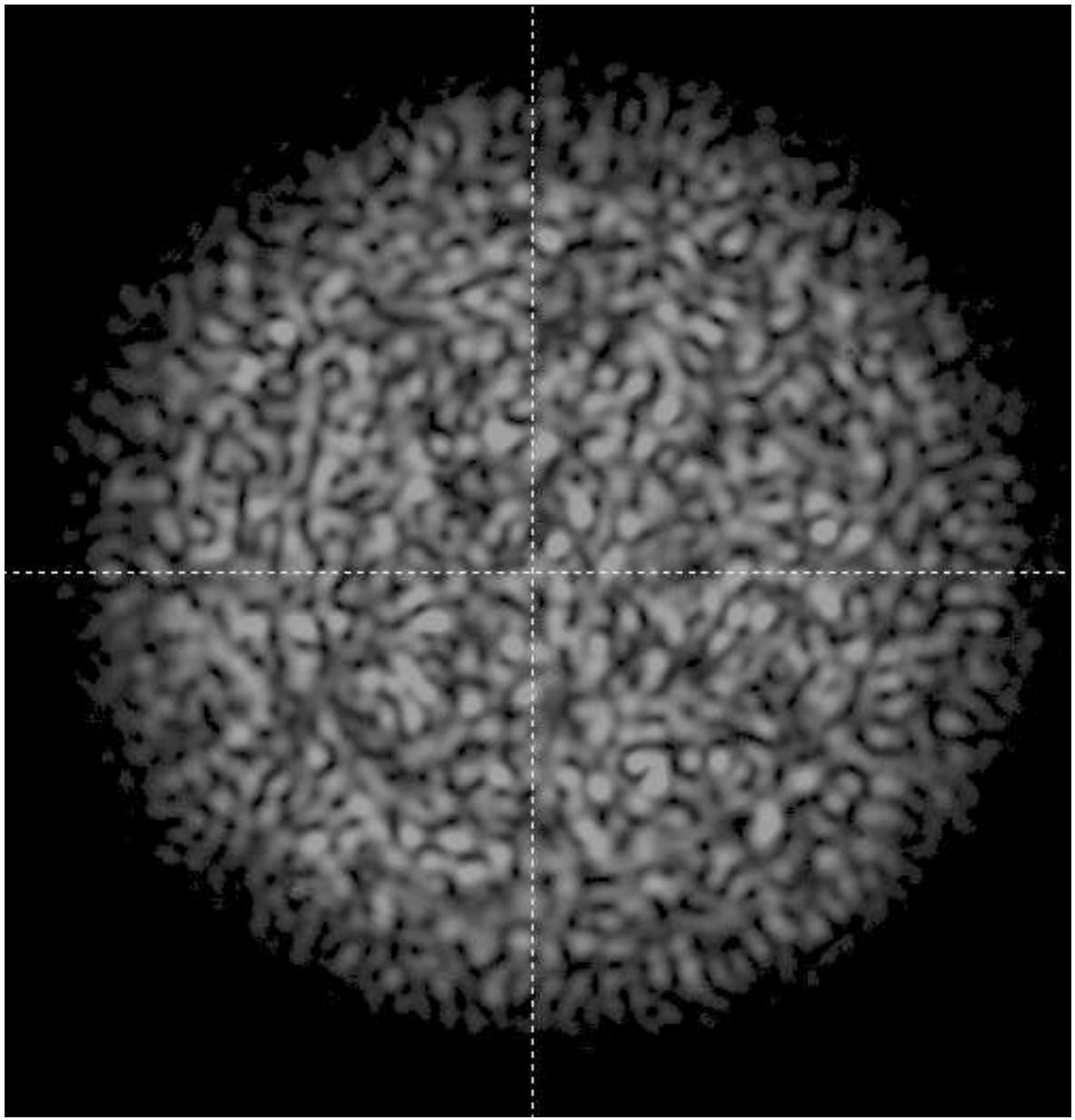
[20] It is possible, but in practice very unlikely, that we hit *exactly* to the local maximum or inflexion point where the derivative is also zero and the process terminates.

[21] B. Redding, S. M. Popoff, and H. Cao,"All-fiber spectrometer based on speckle pattern reconstruction," Opt. Express **21**, 6584–6600 (2013).

[22] B. Redding, S. M. Popoff, Y. Bromberg, M. A. Choma, and H. Cao,"Noise analysis of spectrometers based on speckle pattern reconstruction," Appl. Opt. **53**, 410–417 (2014).

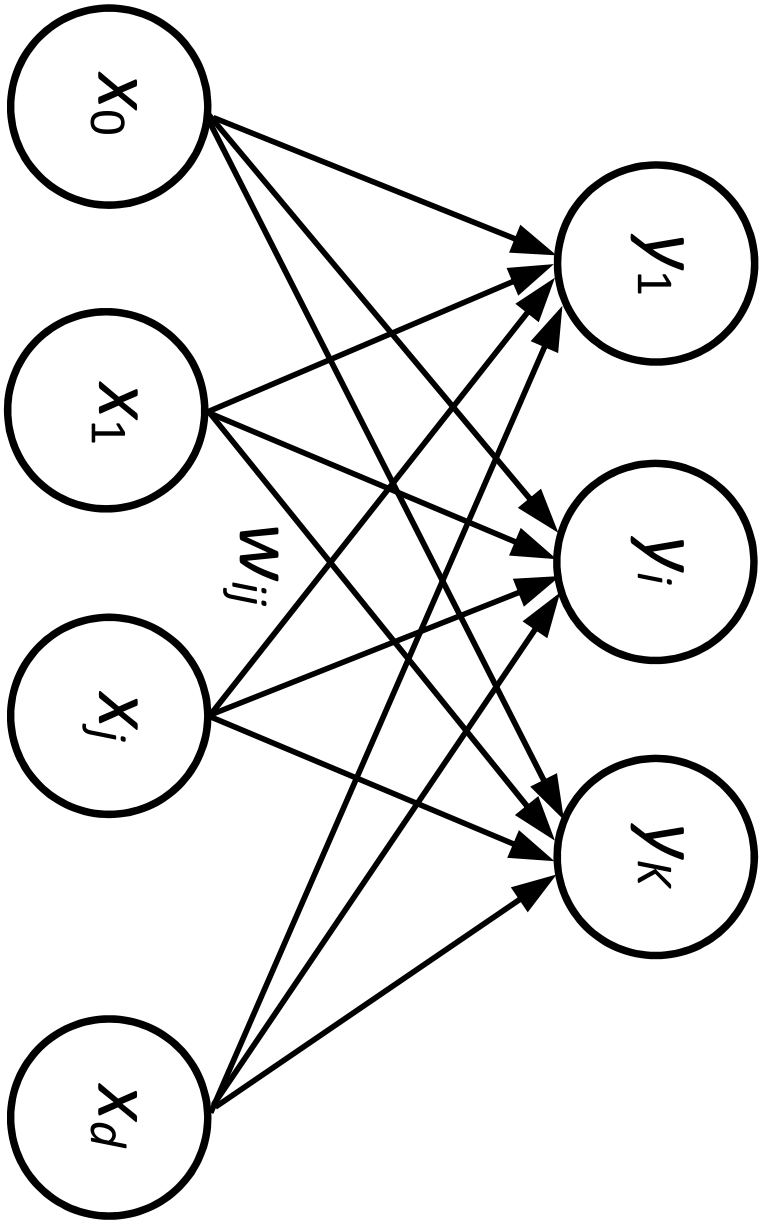
AAPT PHYSICS EDUCATION®
AMERICAN JOURNAL OF PHYSICS

PLEASE CITE THIS ARTICLE AS DOI: 10.1119/5.0102369



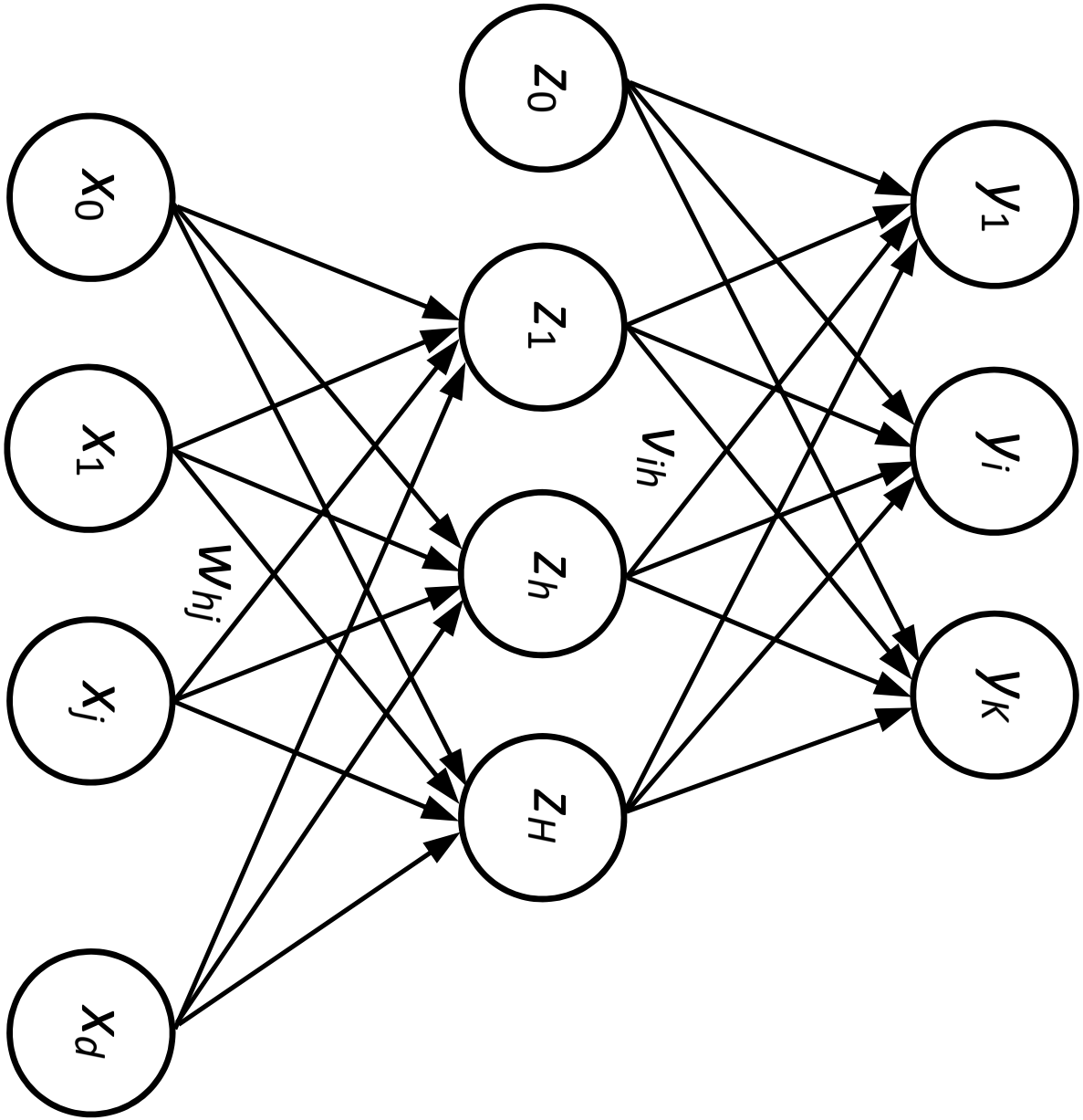
AAPT PHYSICS EDUCATION®
AMERICAN JOURNAL OF PHYSICS

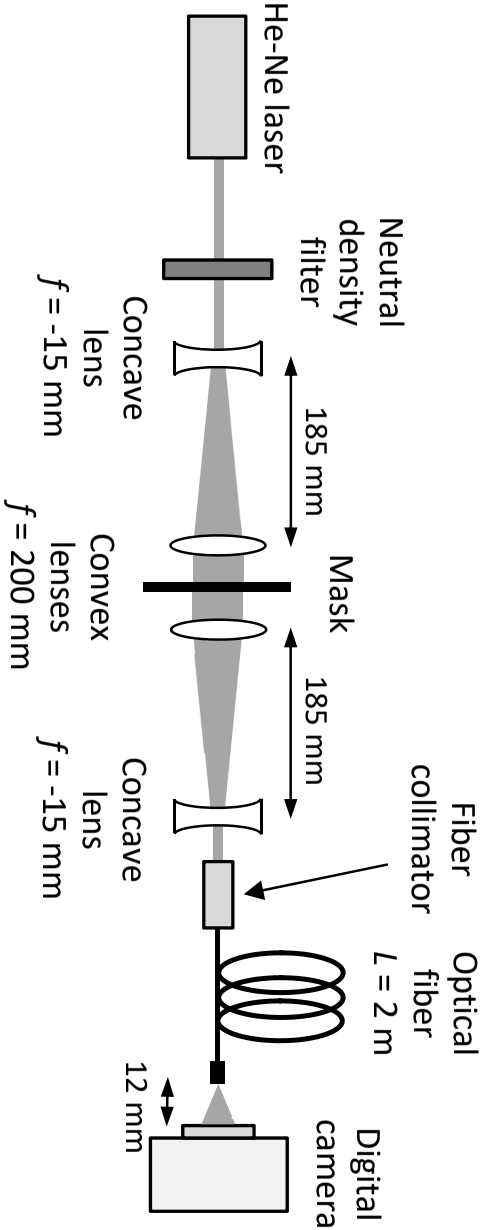
PLEASE CITE THIS ARTICLE AS DOI: 10.1119/5.0102369

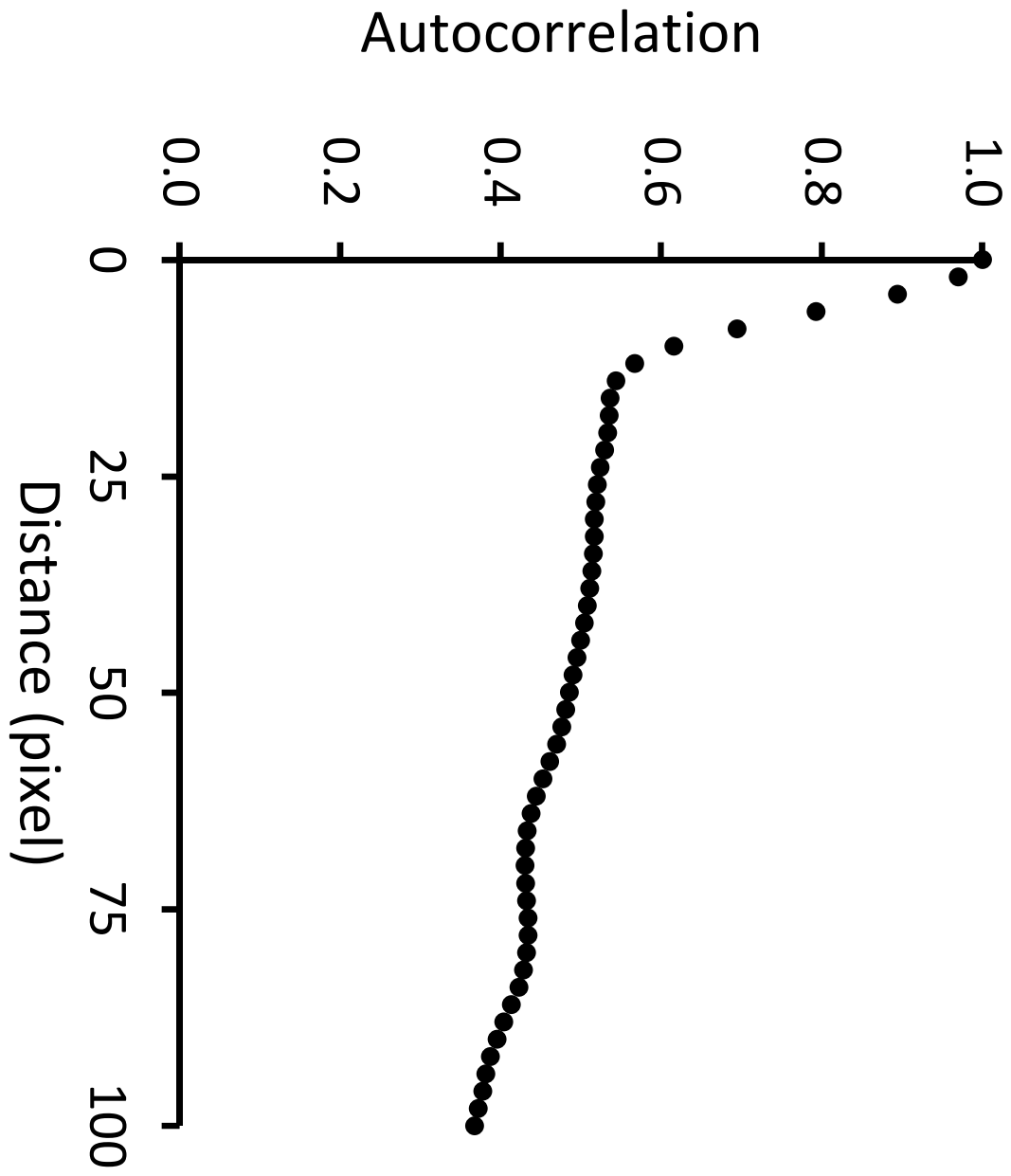


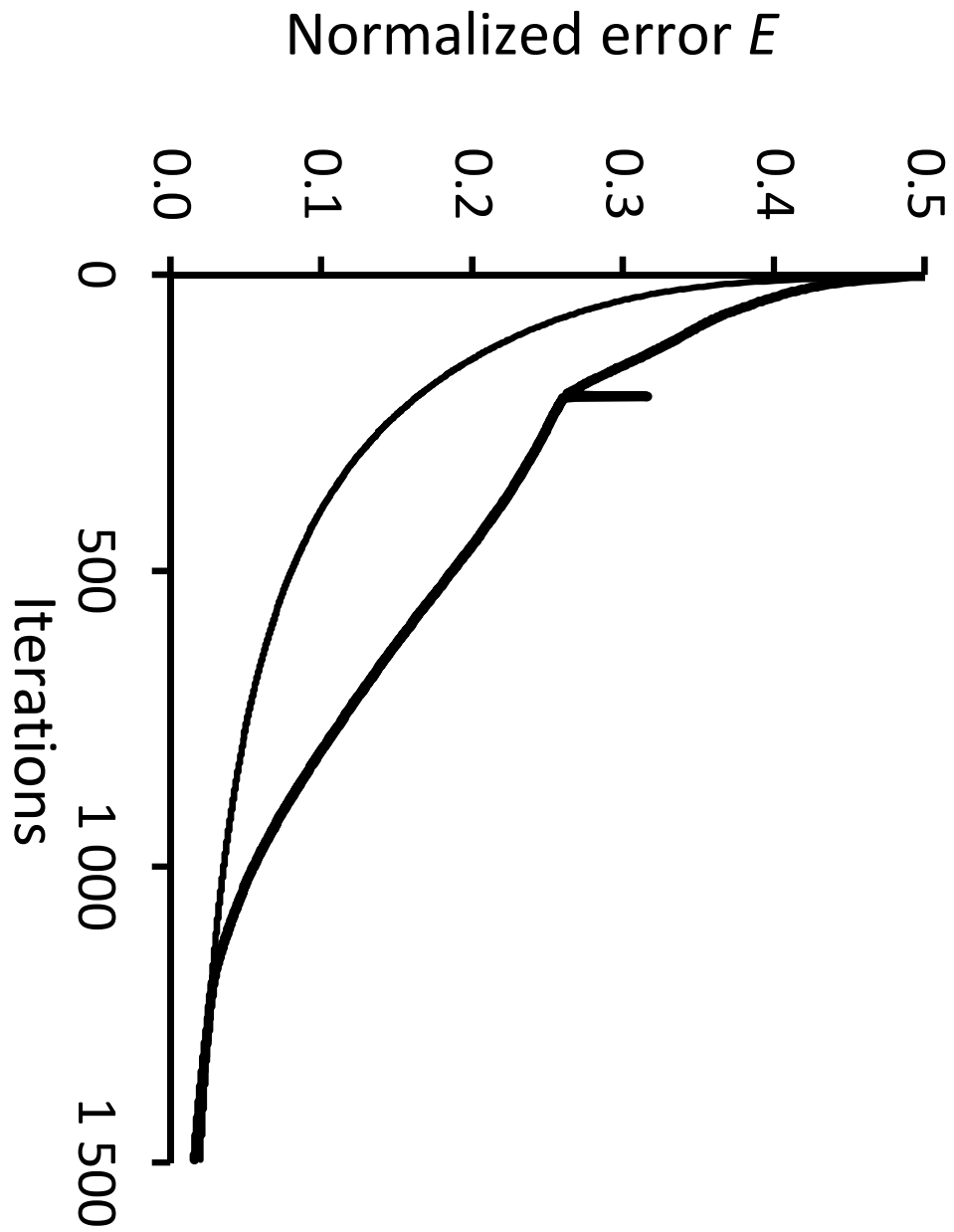
AAPT PHYSICS EDUCATION®
AMERICAN JOURNAL OF PHYSICS

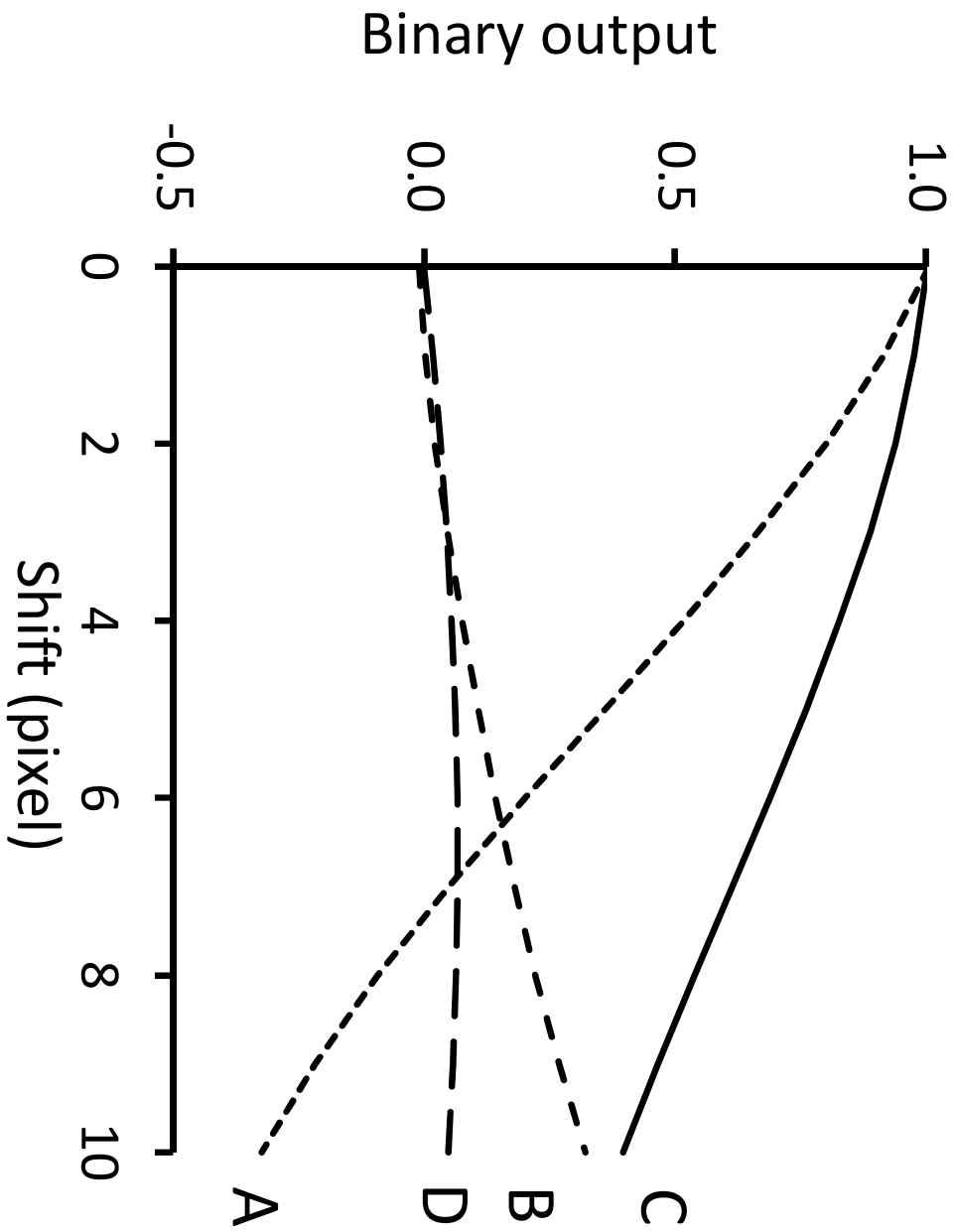
PLEASE CITE THIS ARTICLE AS DOI: 10.1119/5.0102369





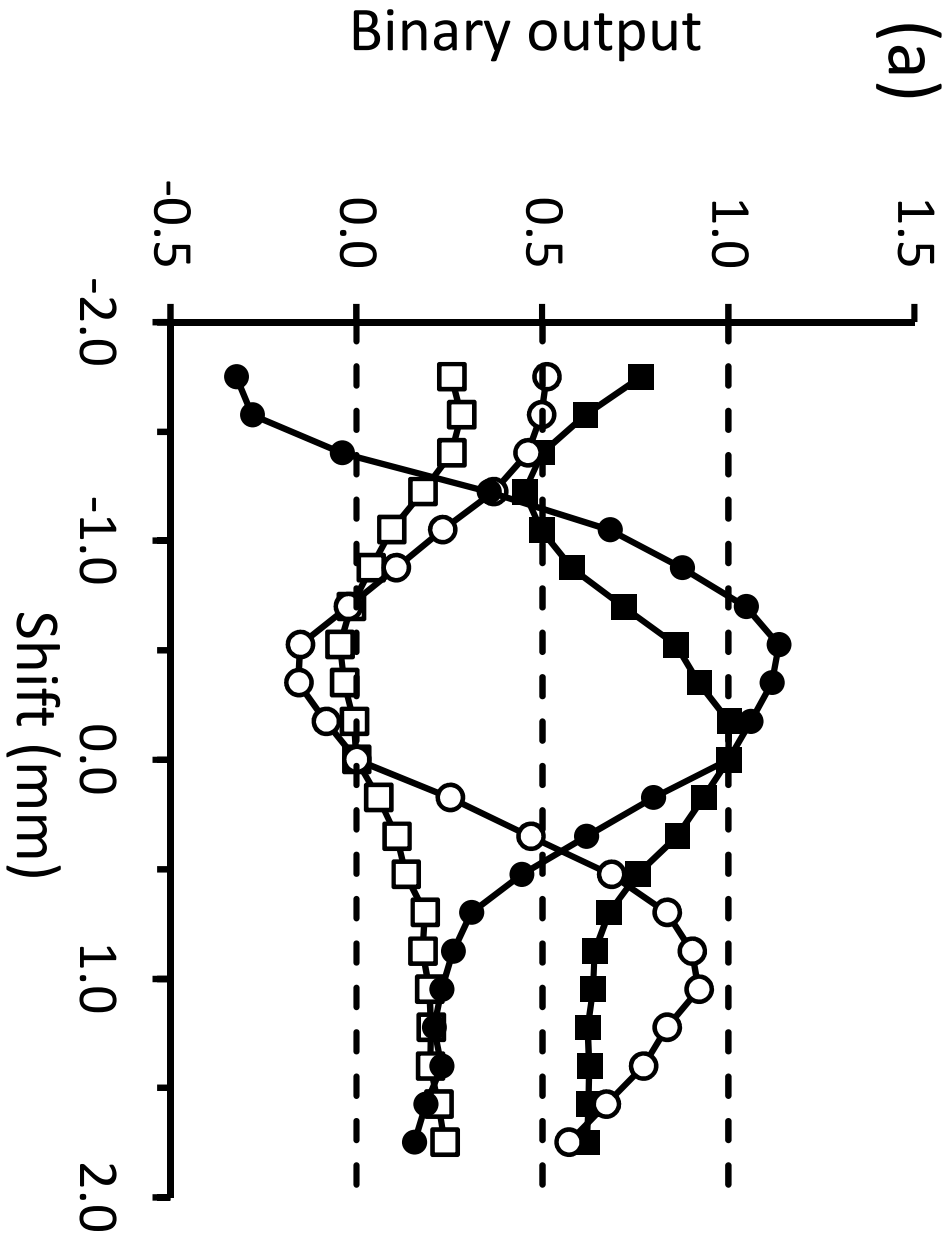


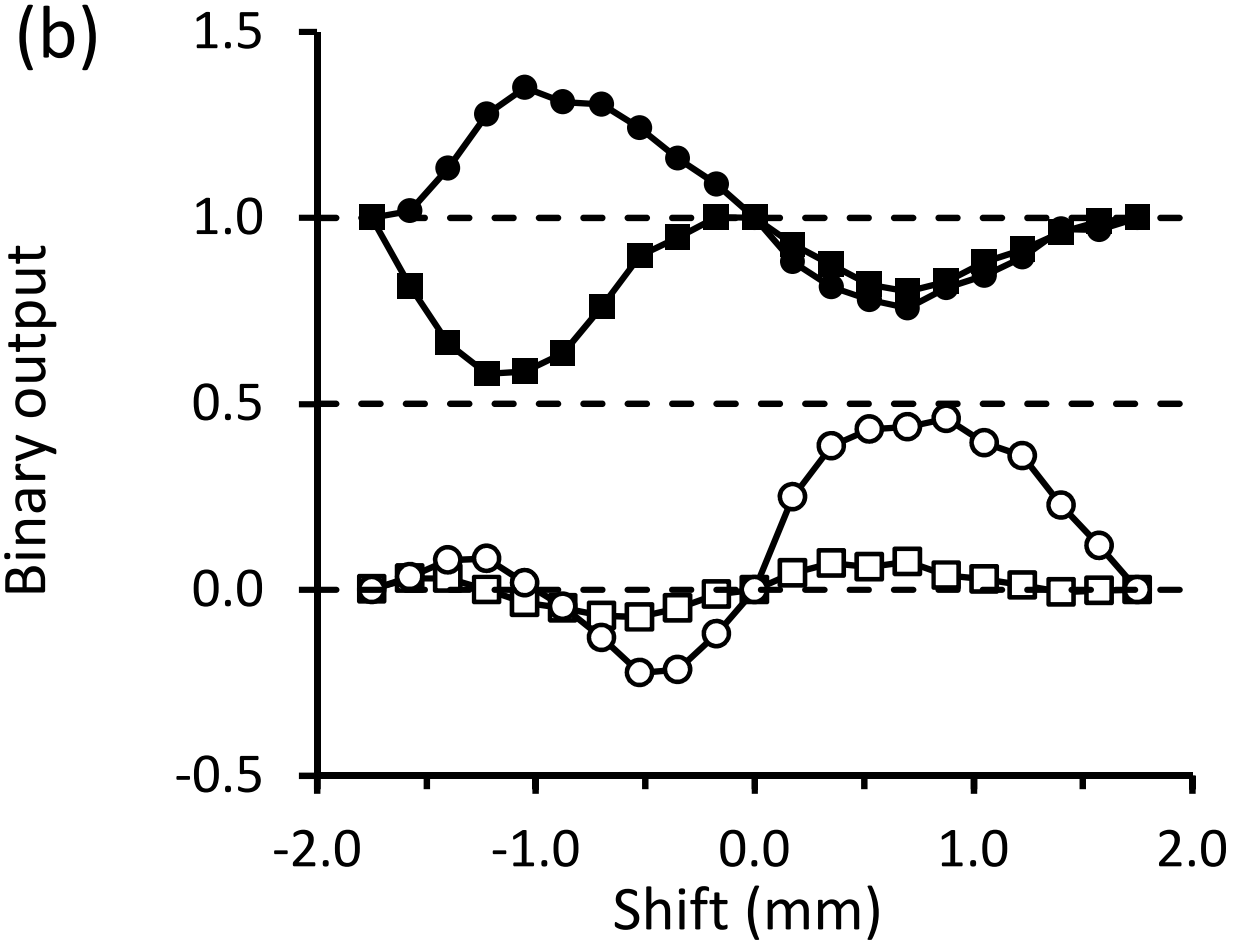


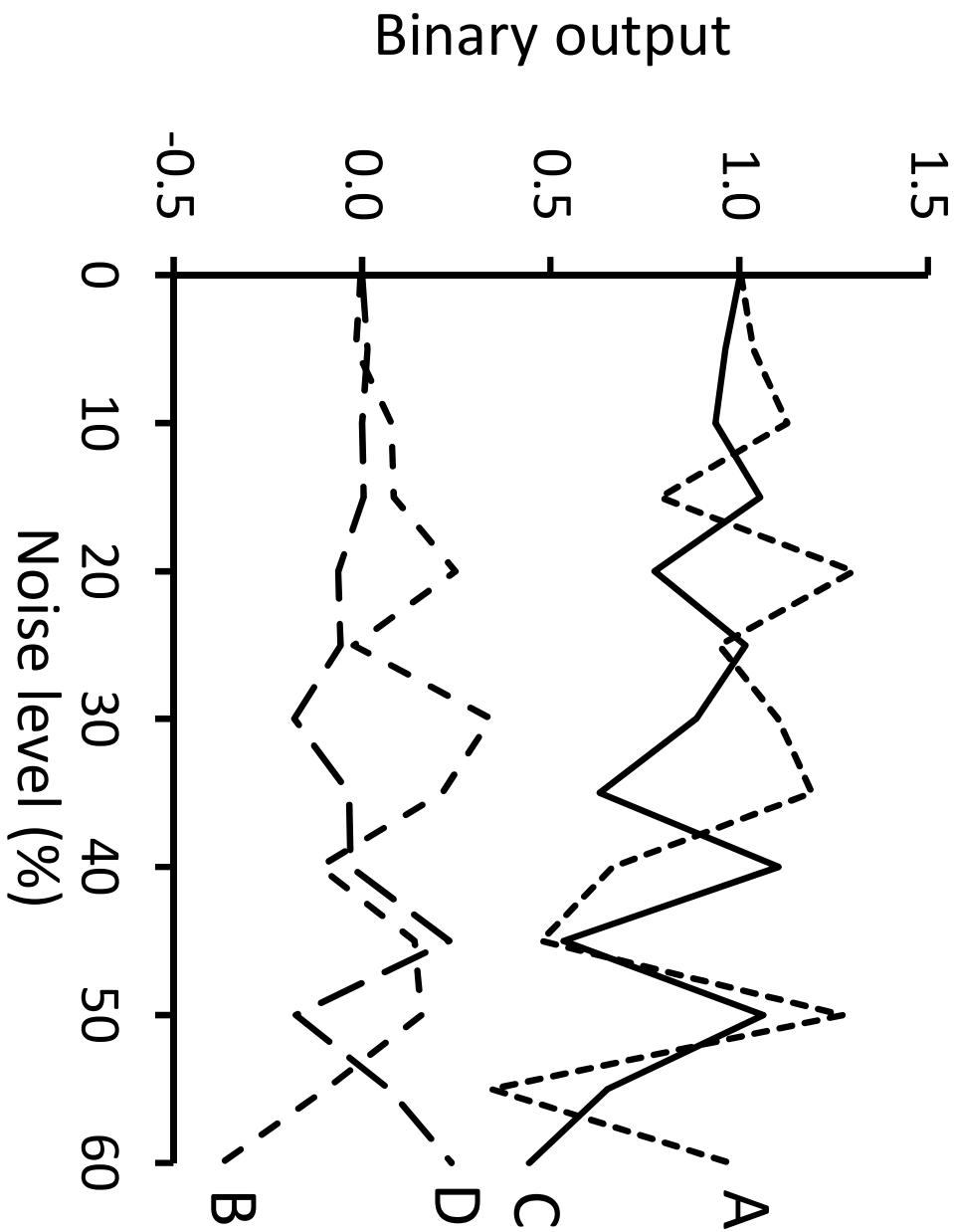


AMERICAN JOURNAL OF PHYSICS

PLEASE CITE THIS ARTICLE AS DOI: 10.1119/5.0102369







AAPT PHYSICS EDUCATION®

AMERICAN JOURNAL OF PHYSICS

PLEASE CITE THIS ARTICLE AS DOI: 10.1119/5.0102369