Received: 4 January 2021 | Revised: 19 March 2021 | Accepted: 1 April 2021

DOI: 10.1111/jmi.13012

Journal of Microscopy RMS

ORIGINAL ARTICLE

Evaluation of classification methods for identifying multiwalled carbon nanotubes collected on mixed cellulose ester filter media

Devin Smith[1,4] | **Nicole M. Neu-Baker**[2] | **Adrienne C. Eastlake**[3] | **Igor G. Zurbenko**[1] | **Sara A. Brenner**[2,5]

[1] Department of Epidemiology & Biostatistics, School of Public Health, State University of New York (SUNY) at Albany, Rensselaer, New York

[2] College of Nanoscale Science & Engineering, State University of New York (SUNY) Polytechnic Institute, Albany, New York

[3] National Institute for Occupational Safety and Health (NIOSH), Cincinnati, Ohio, USA

[4] Present address: Department of Mathematics, School of Science, Rensselaer Polytechnic Institute (RPI), 110 8th Street, Troy, NY 12180, USA

[5] Present address: United States Food and Drug Administration (FDA), Office of In Vitro Diagnostics and Radiological HealthOffice of Product Evaluation and Quality, 10903 New Hampshire Avenue, Silver Spring, MA 20993, USA

Correspondence
Nicole Neu-Baker, College of Nanoscale Science & Engineering, State University of New York (SUNY) Polytechnic Institute, 257 Fuller Road, Albany, NY 12203, USA.
Email: nneu@sunypoly.edu

Abstract

Enhanced darkfield microscopy (EDFM) and hyperspectral imaging (HSI) are being evaluated as a potential rapid screening modality to reduce the time-to-knowledge for direct visualisation and analysis of filter media used to sample nanoparticulate from work environments, as compared to the current analytical gold standard of transmission electron microscopy (TEM). Here, we compare accuracy, specificity, and sensitivity of several hyperspectral classification models and data preprocessing techniques to determine how to most effectively identify multiwalled carbon nanotubes (MWCNTs) in hyperspectral images. Several classification schemes were identified that are capable of classifying pixels as MWCNT(+) or MWCNT(−) in hyperspectral images with specificity and sensitivity over 99% on the test dataset. Functional principal component analysis (FPCA) was identified as an appropriate data preprocessing technique, testing optimally when coupled with a quadratic discriminant analysis (QDA) model with forward stepwise variable selection and with a support vector machines (SVM) model. The success of these methods suggests that EDFM-HSI may be reliably employed to assess filter media exposed to MWCNTs. Future work will evaluate the ability of EDFM-HSI to quantify MWCNTs collected on filter media using this classification algorithm framework using the best-performing model identified here – quadratic discriminant analysis with forward stepwise selection on functional principal component data – on an expanded sample set.

KEYWORDS
hyperspectral imaging, microscopy, occupational exposure assessment, predictive modelling

1 | INTRODUCTION

The need for accurate, specific, and efficient methods for occupational exposure assessment in nanotechnology research and manufacturing environments grows with the rapidly expanding market for nanotechnologies. The U.S. National Institute for Occupational Safety and Health (NIOSH) Respiratory Disease Research Program has conducted studies that indicate that engineered nanomaterial (ENM) exposures represent potentially preventable occupational health hazards[1] and advocates for responsible handling of ENMs.[2] The current analytical gold standard for filter media used to sample for airborne nanoparticulate from occupational exposure assessments

is adapted from historical methods for analysis of micron-sized asbestos: offline analysis via transmission electron microscopy (TEM),[3,4] which may be coupled with energy dispersive x-ray spectroscopy (EDS) for compositional analysis. These methods were not developed for nanoscale materials and are time- and resource-intensive, preventing rapid evaluation of potential worker exposures. These limitations will be exacerbated if industries become required to monitor for nanoparticle (NP) emissions and comply with any future occupational exposure limits (OELs) for nanomaterials.

A rapid screening method for analysis of filter media that is specific for nanomaterials is critically needed to advance nanomaterial exposure science and expedite any health and safety recommendations that may be needed. Enhanced darkfield microscopy (EDFM), coupled with hyperspectral imaging (HSI), is increasingly used for analysis of ENMs in a variety of biological and environmental samples.[5–12] HSI combines spectrophotometry and imaging to capture a spectrum (400–1000 nm) from each pixel in a hyperspectral image.[6] Since hyperspectral images contain both spatial and spectral data, they are also called datacubes. Pixel data may be displayed as a smooth spectrum of wavelength intensities plotted against the tightly sampled wavelength bands, representing the pixel's spectral signature. The spectral signature data contained in each pixel of a datacube may be used to identify pixels in an image that contain some given materials of interest by comparing the image's pixel data with pixel data collected from samples of known composition. This is known as hyperspectral classification.[13]

There are many methodologies for preprocessing hyperspectral data and classifying pixels in a datacube. It is common with HSI to test several methods on an image containing pixels whose classes are known in order to determine the accuracy, sensitivity, and specificity of a given method, as well as to compare across methods in order to identify those that perform the best. However, at microscopic scales, establishing such a 'ground truth' map against which the accuracy of a classification algorithm can be assessed may be nontrivial, especially when the media is complex, as it may be extremely difficult to tell with certainty which pixels contain the material of interest. As a result, studies utilising EDFM-HSI for hyperspectral classification of ENMs often leave statistical parameters, such as sensitivity and specificity, untreated, or informally examine specificity only, by applying the classification algorithm to a sample known to be unexposed and showing that few pixels are falsely mapped to the particle of interest.[14] An additional consequence of this obstacle to training and testing classification algorithms in the context of HSI for ENMs is that few classification algorithms have been formally tested.

The spectral angle mapper (SAM) algorithm is commonly employed for classification in the ENM context because a full 'ground truth' map is not required for training this algorithm. Instead, the SAM algorithm constructs an initial collection of spectra corresponding to pixels believed to contain a material of interest from a positive control image. The collection of spectra is then filtered against an image of a sample in which it is known that the material of interest is absent (a negative control image) in order to remove all pixels in the collection that do not correspond to the material of interest. Thus, the collection is reduced to contain only spectra from the material of interest and specificity is thereby increased. This reduced collection of spectra is then used to map new images of unknown composition.[14] This method appears to make reliable hyperspectral classification possible in the absence of a 'ground truth' map, but the absence of such a 'ground truth' map has, so far, prevented users from determining the method's classification sensitivity and from testing other classification methods.

However, filter media used to capture airborne multi-walled carbon nanotubes (MWCNTs) presents a unique opportunity to test classification algorithms on ENM spectral data in a fully rigorous fashion, replete with a large set of pixels from both MWCNT(+) and MWCNT(−) classes for which a 'ground truth' map is available. Several classification algorithms may thus be fully characterised and compared. Since the sample matrix is quite simple – a very visually homogenous filter – and the materials of interest are morphologically unique, spatially contiguous tubes, a relatively large sample of MWCNT(+) pixels may be visually selected with confidence. While the distinct morphology of MWCNTs lends itself to a visual assessment of the presence of MWCNTs captured on filters – making the use of software classification algorithms seem like overkill – using a classification algorithm can expedite analysis of a large number of samples, particularly if an estimation of structure count is needed. The work presented here will also inform protocol development for analysis of other ENMs on filters, for example, spherical or quasi-spherical ENMs that cannot be confidently identified by visual assessment alone.

From this sample set, MWCNT(−) pixels are easily sampled from images of filters unexposed to MWCNTs. These samples of pixels may then be used to train classification algorithms and determine their accuracy (the percent of all pixels correctly classified), sensitivity (the percent of MWCNT(+) pixels correctly classified), and specificity (the percent of MWCNT(−) pixels correctly classified). Accuracy, sensitivity, and specificity may then be used to compare the performance of competing classification algorithms.

This study leverages this property of MWCNTs to test several well-known data preprocessing techniques and classification models against the popular SAM algorithm and compare performance. Here, we conduct a rigorous and objective evaluation of optimal signal processing algorithms using MWCNTs as the test material. A sample of pixels of known class (MWCNT(+) and MWCNT(−)) was amassed by means of visual assessment and divided into one training and one testing sample. The training set was used to tune several classification algorithms, each of which combined a variety of data preprocessing steps with a variety of classification models. Each algorithm was then tested on the testing sample for accuracy, sensitivity, and specificity, and the algorithms' effectiveness were then compared. Our goal is for this analysis to provide an evidence base to determine which classification algorithms are most appropriate in this context, complete with fully rigorous effectiveness characterisations. In this way, this study will lay the groundwork necessary for actualising EDFM-HSI as a modality for MWCNT exposure assessment.

2 | METHODS

2.1 | Sample preparation

2.1.1 | Multiwalled carbon nanotubes (MWCNTs)

Nanocyl NC7000 MWCNTs (Nanocyl SA, Belgium) were used for this study. These MWCNTs are thin multiwall carbon nanotubes that are produced by the catalytic chemical vapour deposition (CCVD) process and have an average diameter of 9.5 nm and an average length of 1.5 μm.

2.1.2 | Sample generation: MWCNTs on mixed cellulose ester (MCE) filter media

An aerosol of Nanocyl NC7000 MWCNTs was created via an aerosol generation system (NIOSH HELD, Morgantown, WV). This system is capable of sustaining an aerosol concentration of 0.5 mg/m^3. Leland pumps were used to create filter-based samples on mixed cellulose ester (MCE) filter media at a goal flow rate of 4LPM. The samples analysed and presented here were split between a training set and a testing set. For training, one filter blank and one positive control sample exposed to 17.91 mg/m^3 MWCNT [equivalent to 8 hour time-weighted average (TWA) of 3.7303 μg/m^3] was analysed. For testing, another filter blank and a positive control sample exposed to 3.07 mg/m^3 MWCNT (equivalent to 8 hour TWA of 1.279 μg/m^3) was analysed. A total of 2 filter blanks and 2 positive controls were used for this initial study. Filter blanks were prepared using filtered air only. This sample generation method approximates particle capture as would be found in field-based occupational exposure assessments of exposures to airborne nanoparticulate.

2.1.3 | Filter sample preparation for imaging

Exposed MCE filters were prepared for EDFM-his.[15] A portion of the filter (approximately 20–25% of the filter) was cut and placed on a cleanroom-cleaned glass microscopy slide (NEXTERION, SCHOTT North America, Inc., Tempe, AZ, USA). A cleanroom-cleaned glass coverslip (NEXTERION) was placed on top of the filter but was not adhered. Acetone (approximately 500 μL) was pipetted in between the slide and coverslip to saturate and clear the filter portion. The coverslip was sealed with clear nail polish. Cleanroom-cleaned glass microscopy slides and coverslips were used to minimise contamination of the filter samples.

2.2 | Enhanced darkfield microscopy and hyperspectral imaging (EDFM-HSI)

The CytoViva (CytoViva, Inc., Auburn, AL) EDFM-HSI system provides wavelength intensities for each pixel in a hyperspectral image – also called a datacube – over the spectral wavelength range 400–1000 nm, sampled with 2nm spectral resolution.[5] Optical darkfield (DF) images and hyperspectral datacubes were captured as previously described.[6–8,14,16] Briefly, DF images and hyperspectral datacubes were acquired with a CytoViva EDFM-HSI system (CytoViva, Inc.) mounted on an Olympus BX-43 microscope with an EDFM condenser. Images were obtained at 40× magnification using a DAGE optical camera system for DF images and a Pixelfly camera system for HSI that captures spectral data per pixel in the VNIR region. Ten DF images and 10 corresponding datacubes were captured per sample. Pixel resolution at 40× magnification with low spatial resolution is approximately 323nm x 323nm. Resolution was sacrificed in favour of increased area within the field of view (FOV). Images were captured randomly within the filter area for statistical purposes. MWCNTs were visible with EDFM as bright structures. DF images and hyperspectral datacubes were also collected for a filter blank. All datacubes were corrected for the spectra contributed by the microscope's light source. Following correction for the light source, all datacubes were spectrally subset from 450 to 725 nm to remove noise at <450 nm and >725 nm for use with the SAM classification algorithm. All

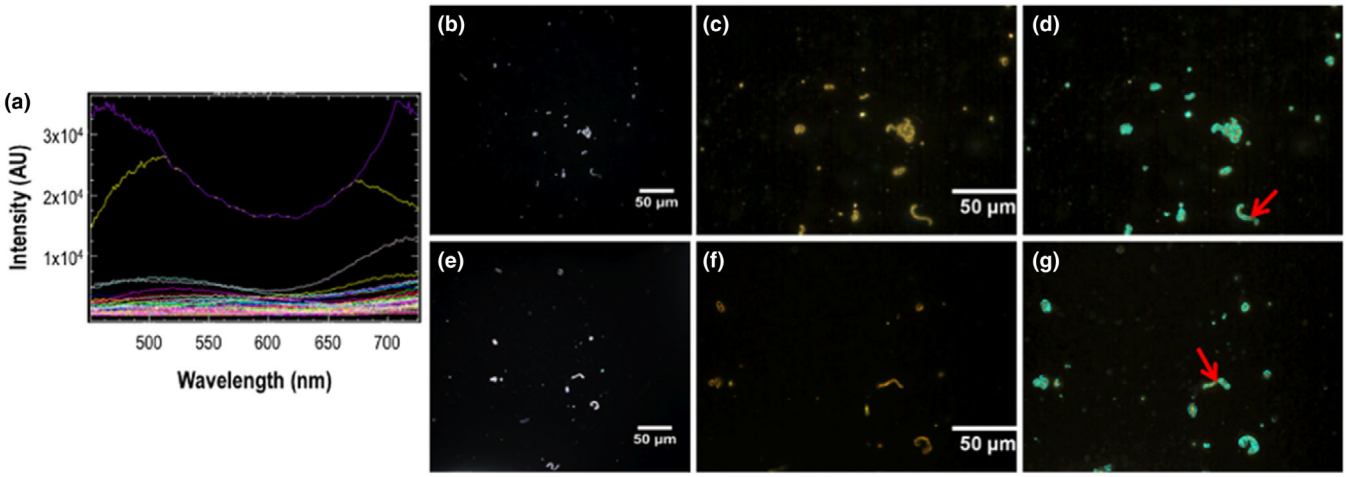

FIGURE 1 Enhanced darkfield and hyperspectral images of MWCNTs collected on MCE filter media
(a) RSL for MWCNTs collected on MCE filters. (b–d) Images of the training set positive control; (b) DF image; (c) HSI datacube; (D) mapped HSI datacube classified using the RSL in (a) with the SAM algorithm. (e–g) Images of the test set positive control; (e) DF image; (f) HSI datacube; (g) mapped HSI datacube classified using the RSL in (a) with the SAM algorithm. Pixels classified as MWCNT(+) are indicated by the aqua false coloration overlay in (d) and (g). Pixels at the centre of MWCNTs were typically unclassified (indicated by red arrows)

other classification models were used on lamp-corrected datacubes that were not spectrally subset.

2.3 | Training and testing set creation for the classification experiment

Each classification algorithm tested in this paper was trained over a set of data (the spectra corresponding to a collection of pixels) with known class identities (the training set) in order to be used to predict the class of a new pixel based on its spectrum. Additionally, in order to understand the performance of these algorithms, another set of spectral data with known class identities must be held separate from the training samples (the test set). The algorithms are used to classify this test set of data so that the predicted class labels can be compared with the known class labels in order to give a fair assessment of the models' accuracy, sensitivity, and specificity using truly new data.

2.3.1 | Training set

A training set was constructed using pixels manually selected over 10 hyperspectral datacubes taken from one MWCNT-exposed filter and 1 datacube taken from a vehicle blank filter. Known MWCNT(+) pixels were added to the training set by visually assessing these 10 hyperspectral datacubes of one MWCNT-exposed filter and selecting the pixels which were confidently assessed to correspond to MWCNTs. It should be noted that, in these 10 datacubes, not all pixels were categorised; rather, a subset of the pixels in a given MWCNT-exposed filter image were added to the MWCNT(+) pixel bank if they belonged unambiguously to the tube-like structures shown in Figures 1c and 1f. This is possible due to the characteristic morphology of MWCNTs, and how this morphology stands in bright contrast to the filter media. Known MWCNT(−) pixels were added to the training set from the image taken of the vehicle blank filter by manually selecting blocks of pixels in a manner that attempted to capture as many features in the blank filter image as possible, including the various blips (of unknown composition) appearing in the image. The training set contained 3931 MWCNT(+) pixels and 5514 MWCNT(−) pixels in total.

2.3.2 | Testing set

A testing set was then constructed from 10 hyperspectral datacubes of a different MWCNT(+) filter and 4 datacubes from a different vehicle blank filter. The test set contained 1507 MWCNT(+) pixels that were chosen from the exposed filters using the same methodology used to construct the training set. Additionally, 1250 MWCNT(−) pixels from each of the four datacubes taken of the vehicle blank filter were randomly selected for the test set. Thus, the test set contained 1507 MWCNT(+) and 5000 MWCNT(−) pixels in total.

2.4 | Statistical tests for comparison

Hypothesis tests were performed to compare the model attaining highest sensitivity on the test set to every other model tested, and to compare the model attaining the highest specificity on the test set to every other model tested. In particular, McNemar's test for paired comparison of binary data was used to test, for each suboptimal model, the hypothesis that that suboptimal model performed as well as the best performing model. For the sensitivity tests, this was carried out over test set MWCNT(+) pixels only. For the specificity tests, this was carried out over test set MWCNT(−) pixels only. p Values were adjusted to control the family error rate separately for the sensitivity tests and specificity tests, using the Holm sequentially rejective method.[17]

2.5 | Classification models

Here, we describe the classification models tested for classifying the MWCNT images. With the exception of the SAM and CNN algorithms, all models were tuned over the training set using five-fold cross-validation, such that a relevant grid of model parameters was specified and the parameterisation that maximised cross-validation classification accuracy over the training set was selected as the final model for testing. For SAM, no tuning was necessary, and for CNN, a single validation set of 450 MWCNT(+) samples and 827 MWCNT(−) samples was broken off from the training set.

2.5.1 | Spectral angle mapper (SAM) algorithm

First, a reference spectral library (RSL) was created using the particle filtering method.[6–8,14] Briefly, a preliminary spectral library was created by the particle filtering method in the ENVI 4.8 (Harris Corporation, Broomfield, CO) HSI software, which selects pixels that meet user-defined spectral intensity parameters. This process was done on a lamp-corrected and spectrally subset datacube so that the influence of the lamp spectrum and noise ≤450 nm and ≥725 nm were removed. The preliminary spectral library was then filtered against a negative control (filter blank) datacube to remove any duplicative spectra corresponding to the filter and/or any potential contamination from sample preparation. Finally, the library was filtered against itself to remove any duplicative spectra so that the final RSL consisted only of unique spectra. The final RSL was used for hyperspectral mapping using the SAM algorithm. The RSL was used to map to each lamp-corrected and spectrally subset datacube using the SAM algorithm (threshold = 0.1 radians) to create an image with a false coloration overlay that identifies the pixels with spectra that correspond to spectra in the RSL. The mapped image therefore identifies NP(+) pixels. Additionally, the SAM algorithm using the RSL was applied to two other datacubes: one that contained each MWCNT(+) pixel from the test set (n = 1507 pixels) and one that contained each MWCNT(−) pixel from the test set (n = 5000 pixels) in order to evaluate the ability of SAM to correctly classify MWCNT(+) and MWCNT(−) pixels.

2.5.2 | K nearest neighbours (KNN) with cosine angle distance

The K nearest neighbours (KNN) classification algorithm – for a given distance function, odd integer k, and training set – predicts the class of a new sample by first calculating the distance between the new sample and each element of the training set. It then classifies the new sample as the majority class membership among the k closest (in terms of the distance function) training set elements.

For this study, distance between spectra was calculated as the cosine of the angle between the data vectors that represented those spectra. For two spectral vectors v and u, this cosine angle distance may be expressed as $\cos\theta = (v \bullet u)/\|v\|\|u\|$. This distance function is chosen, in this algorithm and the SAM algorithm, because it is average intensity invariant. Since spectral intensity units are arbitrary and differences in spectral intensities between similarly shaped curves may correspond to nuisance artefacts of image acquisition, such as exposure time, focal plane or aperture, a spectrum's average intensity may add uninformative noise to the classification problem. For this reason, an average intensity invariant distance function may be more likely to quantify the important differences between spectra, such as spectral shape.

This algorithm is, in some sense, an extension of the SAM algorithm. The SAM algorithm classifies a pixel as MWCNT(+) if its cosine angle distance to any member of the RSL is within some fixed threshold, and KNN uses the same cosine angle distance function, but allows the threshold to vary by the pixel spectrum's location in spectra-space (as determined by the k nearest neighbors rule). KNN uses the training samples analogously to SAMs use of the RSL; however, KNN is trained in a fully supervised manner, while the creation of SAM's spectral library is a semisupervised process. Because KNN is a fully supervised learning procedure, the model can be more formally tuned over k values, making it a convenient and informative extension to the SAM algorithm. The parameter k was tuned over an array of possible k values ranging from 1 to 400, incrementing by 6.

2.5.3 | Logistic regression (LR) with elastic net penalty (LR net)

Logistic regression with elastic net penalty is a linear classification model that employs a penalty to stabilise model predictions and perform feature selection in the case where the model's predictors are highly correlated.[18] Where logistic regression maximises the log likelihood function loglog $L(\beta)$, the model with elastic net penalty maximises the penalised log likelihood

$$\text{loglog L}\beta + \lambda \left[(1-\alpha)\frac{1}{2}\sum_{j=1}^{n}\beta_j^2 + \alpha\sum_{j=1}^{n}|\beta_j| \right] \quad (1)$$

where α and λ are model parameters. Inside the brackets, we see a linear combination of the lasso penalty and the ridge penalty, moderated by α. The lasso penalty $\sum_{j=1}^{n}|\beta_j|$ both shrinks coefficients to increase model stability and performs feature selection by potentially tamping coefficients down to zero, while the ridge penalty $\sum_{j=1}^{n}\beta_j^2$ shrinks coefficients only. The relative performance of these two penalties is situation-dependent.[19] Hence, this model uses the α (mixing) parameter to enable the modeller to use the data to strike a balance between the ridge and lasso penalties. The λ (regularisation) parameter determines how strongly model coefficients should be penalised overall. These parameters were set via grid search ($\alpha = 0$, 0.1, 0.2, 0.4, 0.6, 0.8, 1; λ in grid including 0, 0.005 and 25 evenly spaced values from 0.01 to 0.3) in five-fold cross-validation over the training set.

2.5.4 | Quadratic discriminant analysis (QDA) using forward selection

Quadratic discriminant analysis (QDA) is a generalisation of linear discriminant analysis (LDA), a method that minimises the probability of misclassification, assuming the data in each class are multivariate normal with a common variance–covariance matrix, by finding the linear combination of predictor variables that maximises the ratio of the between-group variability and the within-group variability. Therefore, LDA finds the hyperplane between classes that minimises the probability of misclassification. QDA drops the assumption that the two classes have a common variance–covariance matrix, and as a result, yields a quadratic boundary between classes. Since the effectiveness of LDA and QDA models is deteriorated when uninformative predictors are included in the model, a forward selection procedure was incorporated. Model accuracy over 10-fold cross-validation was used as the forward selection criteria. For the multivariate PC models, selection was stopped when the 10-fold cross-validation accuracy ceased to improve or the maximum number of allowed predictors was reached. For these models, the maximum number of allowed predictors was also treated as a tuning parameter over five-fold cross-validation. Thus, the 10-fold cross-validation forward selection procedure was performed within each of the five folds for each tested number of allowed variables, an integer tuning grid from 1 to 10. For the FPCA models, the added FPC smoothing parameter increased the run-time of the fitting procedure, so the maximum number of allowed predictors was removed as a tuning parameter. Instead, selection was simply stopped when five-fold cross-validation accuracy over the training dataset increased by less than 0.001.

2.5.5 | Support vector machines (SVM)

Support vector machines (SVM) is a popular and broadly applicable model for classification. The method creates a very flexible, nonlinear decision boundary between classes by first performing a nonlinear mapping of the data into a new feature space that is higher dimensional than the Euclidean space the data occupies originally and then identifying a linear boundary between classes in this new feature space. This mapping is achieved using a kernel function.[19] Briefly, the linear boundary in the new, high dimensional feature space is chosen to maximise the distance between the boundary and the closest correctly classified data point(s), subject to a cost penalty for observations misclassified by the boundary, where cost is assigned proportional to distance from the boundary. The total allowed cost is a model parameter, C, which was tuned over a grid via cross-validation, allowing for great flexibility in striking a balance between overfitting and underfitting the data. For untransformed and average normalised data (see below), the grid was composed of C values 2^n, $n = -3, -2, \ldots, 8$. For FPCA data (see below), the grid was composed of C values 30, 60, 150, 300, 600, 1000. Additionally, the SVM models using PCA and FPCA data were tuned over the number k of FPCs retained in the model, using a grid search over $k = 10, 25, 50, 100, 150, 200, 453$. This paper employs SVM using the radial basis kernel function. This kernel function comes with another parameter, σ, which was set using an analytical estimate based on the training data.[18]

2.5.6 | Convolutional neural network (CNN)

Convolutional neural networks (CNN) are a class of deep feed-forward neural networks that utilise at least one convolutional layer. They have proven exceptionally useful on data with a natural grid structure, like image data, where

data is arranged along a two-dimensional grid of pixels; time series data, where data is arranged along the one-dimensional grid of temporal observations; and, in our case, hyperspectral data, where intensity data is arranged along the one-dimensional grid of wavelengths.[20] In a convolutional layer, output units are sparsely connected to their input units. In the one-dimensional case, output units are arranged in a series of one-dimensional grids, called filters. Each filter (grid of output units) is the result of convolving a kernel of weights across the one-dimensional grid of (potentially vector-valued) input units. The algorithm learns one kernel of weights for each filter. This means that this kind of feed-forward layer economises on parameters by sharing them across all of the output units within a filter. It also implies that the one-dimensional grid of output units within a given filter represent the same learned feature observed like a sliding window across the one-dimensional grid of input units. Mathematical details are expounded upon elsewhere.[20,21] In recent years, CNNs have been employed successfully in the hyperspectral classification context.[22,23] In this paper, we bring the model architecture designed by Mei et al. for remote sensing problems to our hyperspectral microscopy problem. The architecture is shown in Figure 2 below. The CNN incorporates a batch-normalisation layer between the convolutional layer and its activation function, which helps prevent overfitting in the absence of pooling layers, and which has been shown to enable higher learning rates during optimisation.[24] In the dropout layer, the dropout rate was set to 0.1. ReLu activations were used throughout hidden layers, and the output layer uses the conventional sigmoid activation function. The binary cross-entropy loss function was minimised using stochastic gradient descent with shuffled batches of 35 samples over 200 epochs with a learning rate of 0.1. Let n_1 denote the number of filters and n_2 denote the kernel widths for each filter in the convolutional layer; let n_3 denote the number of units in the dense layer. Then we tuned the model's parameter triplet (n_1, n_2, n_3) over the grid $\{15, 20, 25\} \times \{150, 200, 250\} \times \{80, 100, 120\}$; we selected the parameters which maximised validation set accuracy. All computations were carried out using the Keras package for R Studio.[25] CNN is an end-to-end classification algorithm which automates a supervised feature design and selection procedures; it is for this reason that, in the work that follows, CNN is not coupled with any of the data preprocessing methods discussed in the next section.

2.6 | Data preprocessing methods

Several different methods for preprocessing the spectral data were tested in order to explore how the unique properties of spectral data might best be represented or transformed for classification. However, before each of these techniques was employed, the spectral reflectance intensities from each image were normalised to fall between 0 and 1. Each of the methods discussed below were combined with some or all of the classification models above, as was judged appropriate based on a given model's sensitivities to highly correlated data. Table 1 shows the model/preprocessing method combinations tested.

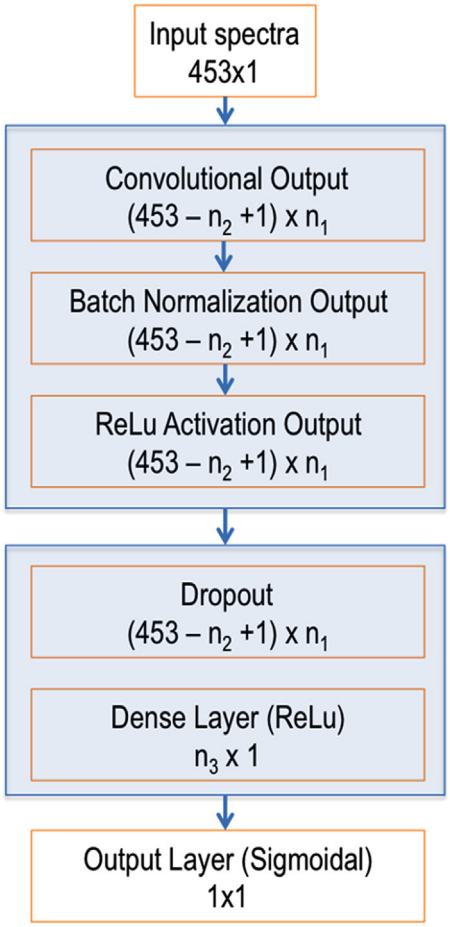

FIGURE 2 Convolutional neural network architecture

2.6.1 | Average normalisation

Figure 3 and the discussion in the KNN methods section illustrates that a spectrum's shape may be more important to solving the hyperspectral classification problem than its average intensity. Hence, it has been proposed that spectra be normalised by their average intensity prior to employing a classification modelling technique, in order to force the model to utilise the spectra's shape information and to downplay the role of overall intensity in classification.[13,26] The normalisation is constructed as follows: for a given spectrum $\upsilon = [\upsilon_1, \ldots, \upsilon_n]$, where υ_i is the spectrum's intensity at wavelength band i, the spectrum's average intensity

TABLE 1 Classification algorithm results on the test dataset

	Normalisation	MWCNT(+) pixels misclassified	MWCNT(−) pixels pixels misclassified	Sensitivity	Specificity	Accuracy
SAM	Untransformed	1386	0	0.080*	1.000	0.7867
LR net	Untransformed	23	26	0.985*	0.995**	0.9925
	Avg. norm.	1	2492	0.999	0.502**	0.6169
	PCA	46	2	0.969*	1.000	0.9926
	PCA – avg. normalised data	4	2013	0.997	0.597**	0.6900
	FPCA – B-spline smooth	31	3	0.979*	0.999	0.9948
	FPCA – PC smooth	32	4	0.979*	0.999	0.9945
SVM	Untransformed	41	10	0.973*	0.998**	0.9922
	Avg. norm.	2	769	0.999	0.846**	0.8815
	PCA	12	154	0.992*	0.969**	0.9745
	PCA – avg. normalised data	0	4	1.000	0.999	0.9994
	FPCA – B-spline smooth	5	7	0.997	0.999	0.9982
	FPCA – PC smooth	2	6	0.999	0.999	0.9988
QDA step	PC	19	2	0.987*	1.000	0.9968
	PCA – avg. normalised data	0	27	1.000	0.995**	0.9959
	FPCA – B-spline smooth	0	9	1.000	0.998**	0.9986
	FPCA – PC smooth	0	6	1.000	0.999	0.9991
KNN	Untransformed ($K = 349$)	1507	0	0.000*	1.000	0.7684
	Untransformed, post hoc ($K = 375$)	3	315	0.998	0.9370**	0.9511
CNN	Untransformed	47	8	0.969*	0.998**	0.9915

*Statistically significantly worse sensitivity than the highest sensitivity models tested, at the 5% level. The highest sensitivity models all had 100% sensitivity.
**Statistically significantly worse specificity than the highest specificity models tested, at the 5% level. The highest specificity models all had 100% specificity.

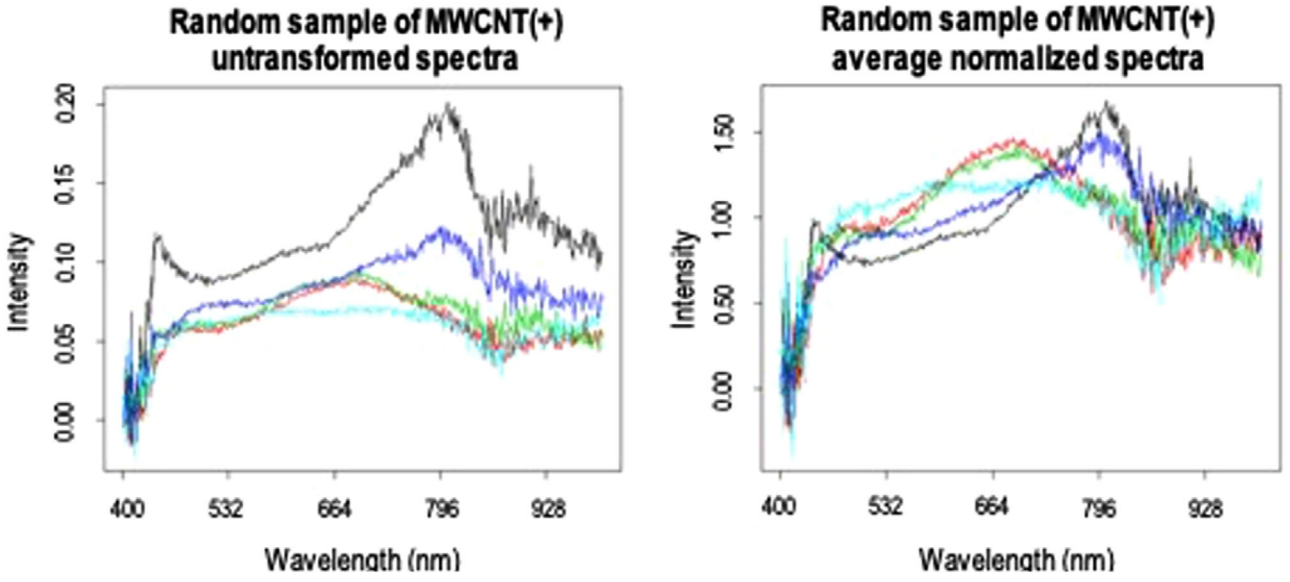

FIGURE 3 Random sample of MWCNT(+) spectra before and after average normalisation
A random sample of MWCNT(+) spectra before and after average normalisation; spectral shape characteristics are emphasised as spectra are constrained to have unit average intensity

is defined equal to $\frac{1}{n}\sum_{i=1}^{n} v_i \equiv \mu$. The average normalised spectrum is then defined equal to $\frac{1}{\mu}v$. This normalisation is carried out separately for each spectrum.

2.6.2 | Principal component analysis (PCA)

As is evident from the curve-like structure of a pixel's spectrum, spectral wavelength bands are very highly correlated. Highly correlated predictors pose a problem for many popular classification algorithms; for example, the QDA classification model is likely to become overfit and make unstable predictions when there is a large number of highly correlated predictors. Principal component analysis (PCA) takes linear combinations of the original predictors to create a new, reduced set of uncorrelated features that represent a large portion of the variation in the sample. Principal components were derived from the training data only; the test set principal component scores are derived from these training set principal components. Additional detail regarding PCA is provided in Supplemental Materials.

2.6.3 | Functional principal component analysis (FPCA)

Functional data analysis (FDA) provides a framework for analysing data that are discretely observed realisations of a continuous stochastic process. Functional principal component analysis (FPCA) is used here to create smoothed approximations to the continuous spectra underlying the HSI data and to specify the approximations in terms of an orthonormal basis for a finite dimensional subspace of L^2 functions. The L^2 inner product between approximated spectra is then conveniently equivalent to the dot product of their vectors of coefficients in the basis. In this way, the method provides both a smooth approximation to the spectra and translates relations between spectra from L^2 to Euclidean space, where most classification algorithms are designed to operate. Goodfellow et al. provide a comprehensive introduction to these topics, which is summarised briefly in **Supplemental Materials** in the context of FPCA for HSI data. Smoothness is imposed on the data through a smoothing parameter λ. The spectra may be smoothed before the FPCA; we refer to this method of preprocessing as B-spline-smoothed FPCA. Alternatively, the spectra may be smoothed directly through the FPCA; we refer to this method of preprocessing as FPC-smoothed FPCA. The smoothing parameter was set via grid search over 10-fold cross-validation on the training set. For B-spline-smoothed FPCA, the search grid set $\lambda = 1.778279 \times 10^n$, $n = 0, 1, \ldots, 5$. For FPC-smoothed FPCA, the grid set $\lambda = 0.1, 1, 10, 100, 316.2, 1000, 1778.3$.

3 | RESULTS

3.1 | Imaging via EDFM-HSI

Filter samples were imaged by EDFM and HSI. MWCNTs were apparent as bright fibre-like structures against a dark background (Figure 1). MWCNT(+) pixels are easily identifiable by visual assessment alone, given the distinct morphology of these structures. MWCNTs were not observed on filter blanks (negative controls; data not shown).

3.2 | Comparison of classification methods

Several classification model/preprocessing method combinations were trained over the training dataset and then tested on the testing dataset, which was composed of 1507 MWCNT(+) pixels and 5000 MWCNT(−) pixels. The accuracy, sensitivity and specificity results for each algorithm tested are displayed in Table 1. The highest accuracy algorithm was SVM with PCA on average normalised spectra, which misclassified only 4 of the 6507 test pixels, attaining perfect sensitivity and 99.9% specificity. QDA forward selection performed over FPC-smoothed FPC scores, which misclassified only six pixels, attaining 99.9% accuracy, perfect sensitivity and 99.9% specificity. This SVM model and the QDA models performed on average normalised PC scores, B-spline-smoothed FPC scores and PC-smoothed FPC scores each had perfect sensitivity. The sensitivity of these algorithms was statistically compared with each of the others using McNemar's test with a Holm p-value correction to control the family error rate. The SAM model, KNN model, QDA model performed on untransformed PC scores and the LR net model performed on average normalised multivariate PC data each had perfect specificity. The specificity of these algorithms was compared with each of the others using McNemar's test with a Holm p-value correction to control the family error rate. All comparisons were made at the 5% level. Statistical test results are displayed in Table 1.

The SAM algorithm had 78.7% accuracy, correctly classifying all MWCNT(−) pixels in the test set, but only correctly classified 8% of the MWCNT(+) pixels. Based on a visual assessment of the SAM classification overlay (Figure 1), it is apparent that many MWCNT(+) pixels were not identified as such. Typically, the pixels at the centre of MWCNT fibres were classified as MWCNT(−). The SAM algorithm

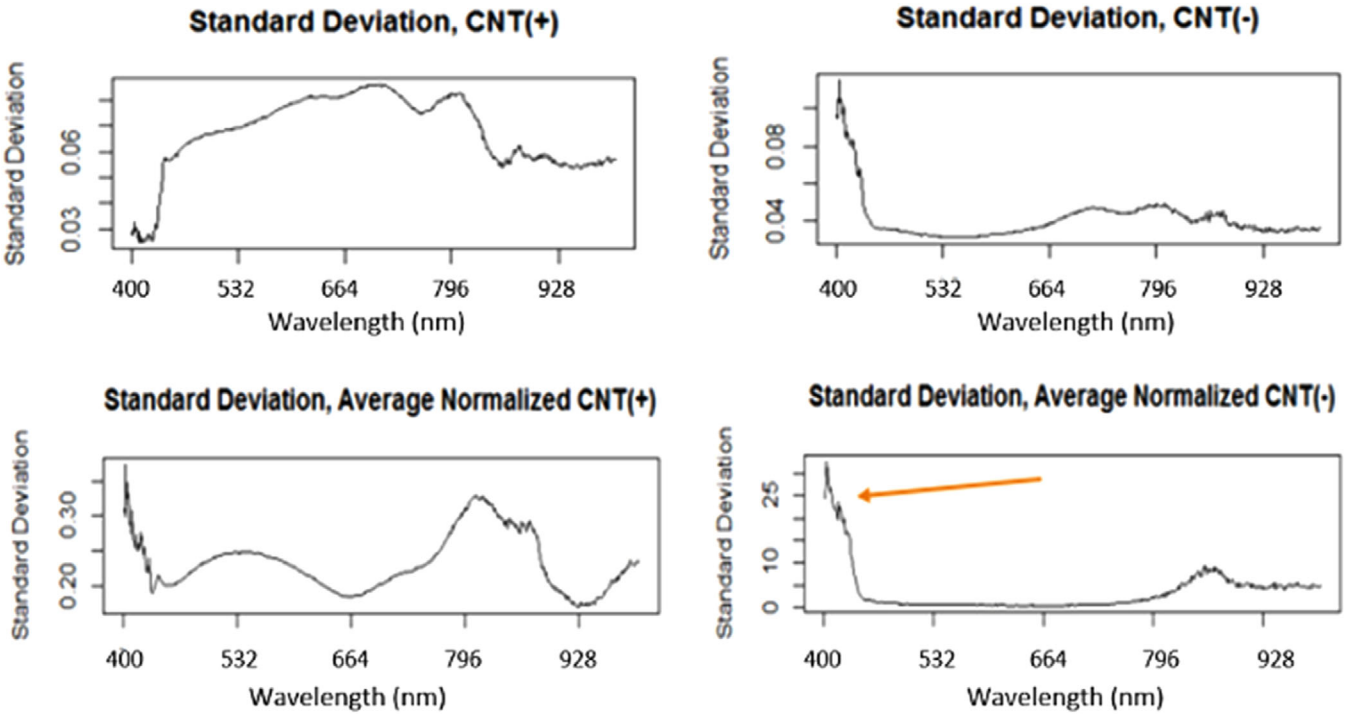

FIGURE 4 Plot of within-class standard deviations by spectral band number, untransformed and average normalised training set pixels
Within-class standard deviation of intensity by band for training pixels before and after average normalisation

had statistically significantly lower sensitivity than the perfect sensitivity algorithms, and had the second lowest sensitivity overall.

The average normalisation transformation appears to successfully emphasise spectral shape and downplay differences in spectral average intensity when applied to MWCNT(+) pixels. A simple random sample of five spectra corresponding to MWCNT(+) pixels is shown in the left-hand panel of Figure 3. Two of the spectra, the black and royal blue plotted curves, have markedly similar shapes, but differ in average intensity: roughly speaking, they appear to be constant multiples of one another. The results of the average normalisation performed on the same random sample of spectra are shown in the right panel of Figure 3. All of the spectra are cinched together and the black and royal blue spectra in particular are nearly overlapping. However, this transformation also dramatically inflated the standard deviations of wavelength bands in MWCNT(−) pixels. Figure 4 shows how the standard deviations in the first 50 spectral bands of MWCNT(−) pixels is greatly amplified by the average normalisation. This standard deviation amplification does not occur to the same extent in the same bands of MWCNT(+) pixels. Many of the spectra corresponding to MWCNT (−) pixels correspond to microscope slide background and so are dominated by noise, with no apparent signal and very small average intensity. Thus, the average normalisation divides noisy intensity spikes by a small number, effectively amplifying the signal's noise. On the other hand, since MWCNT(+) pixels usually have larger spectral average intensities, noise is not as amplified. In this way, the average normalisation puts the standard deviations of the two pixel classes on essentially different scales. The SVM model using PCA on average normalised data was the best-performing model tested; however, the rest of the classification algorithms using average normalised data all had statistically significantly worse specificity than the perfect specificity algorithms, and among QDA and LR net models, the algorithms using average normalised data had the lowest specificities. Thus, we see that the SVM model works more effectively with the amplified noise in average-normalised MWCNT(−) pixels than the LR net and QDA models tested. Algorithms using average normalised data did not have statistically significantly worse than perfect sensitivity, and within each model type, algorithms using average normalised data have the highest sensitivities.

All models using multivariate PC scores had at least 97% sensitivity, and all but the LR net model using multivariate PC scores constructed from average normalised data had over 99% specificity. When paired with a forward stepwise variable selection procedure under the QDA model, multivariate PCA was really effective at reducing model dimensionality: the stepwise variable selection procedure for the QDA multivariate PC model performed over untransformed data kept just 3 variables, while 10

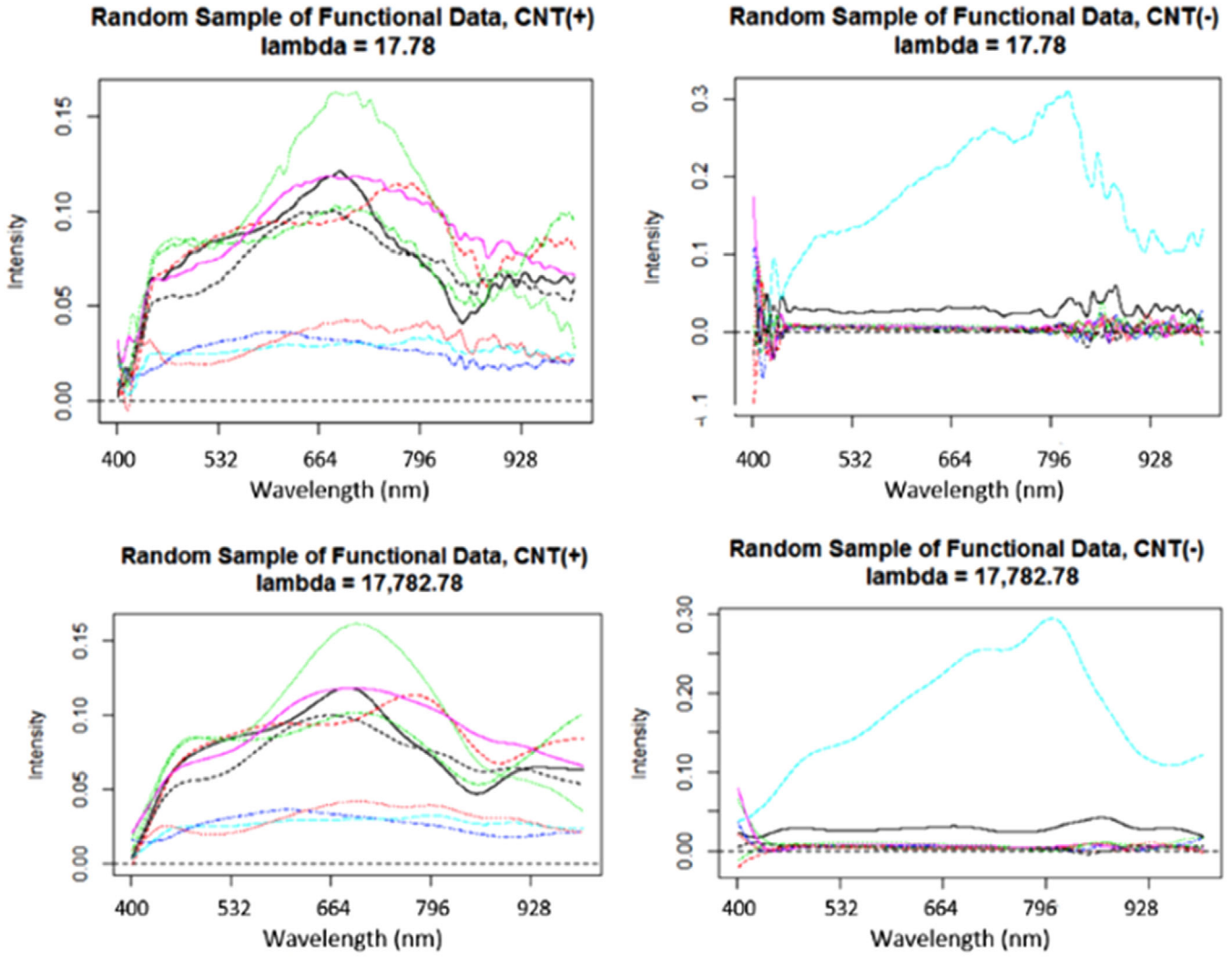

FIGURE 5 Spectra after B-spline fitting
Top row: $\lambda = 17.78$; bottom row: $\lambda = 17782.78$

variables were retained when the same algorithm was performed over average normalised data.

Functional PCA was uniformly effective, with classification accuracy exceeding 99% in every instance in which it was tested. The QDA forward selection models using FPC-smoothed FPC scores data and B-spline-smoothed FPC scores data accounted for two of the four algorithms with perfect sensitivity, and both SVM algorithms based on FPC scores did not have statistically significantly worse sensitivity. Additionally, the SVM and QDA forward selection algorithms using FPC-smoothed FPC scores data and both LR algorithms using FPCA did not have statistically significantly worse specificity. The SVM and QDA stepwise algorithms using FPC-smoothed FPC scores and the SVM algorithm using B-spline-smoothed FPC scores were three of only four algorithms which did not have statistically significantly worse than perfect sensitivity or specificity. Interestingly, the CNN method on untransformed data, which uses the data to design and select its own features in a supervised manner, performed worse than the FPCA methods, and had statistically significantly worse than perfect sensitivity and specificity.

In tuning algorithms that used FPCA, different models selected different optimal λ values: for example, the optimal B-spline-smoothed FPCA λ value was 1.78 for the LR net model, 17,782 for the QDA model, and 17.78 for the SVM model. Figure 5 shows a random sample of MWCNT(+) and MWCNT(−) spectra plotted as their B-spline-fitted continuous functions using the penalised LS criterion (1) with λ set to 17.78 and 17,782. Figure 6 shows, for these two values of λ, the mean and standard deviations of these residuals by band across all training sample spectra. From Figure 5, it is clear that a higher value of λ imposes more smoothness on the fitted curve. For example, in the relatively high intensity teal curve in the MWCNT(−) sample, when λ is set equal to 17.78, small spikes are visible in the band range from 310 to 380, but are completely smoothed

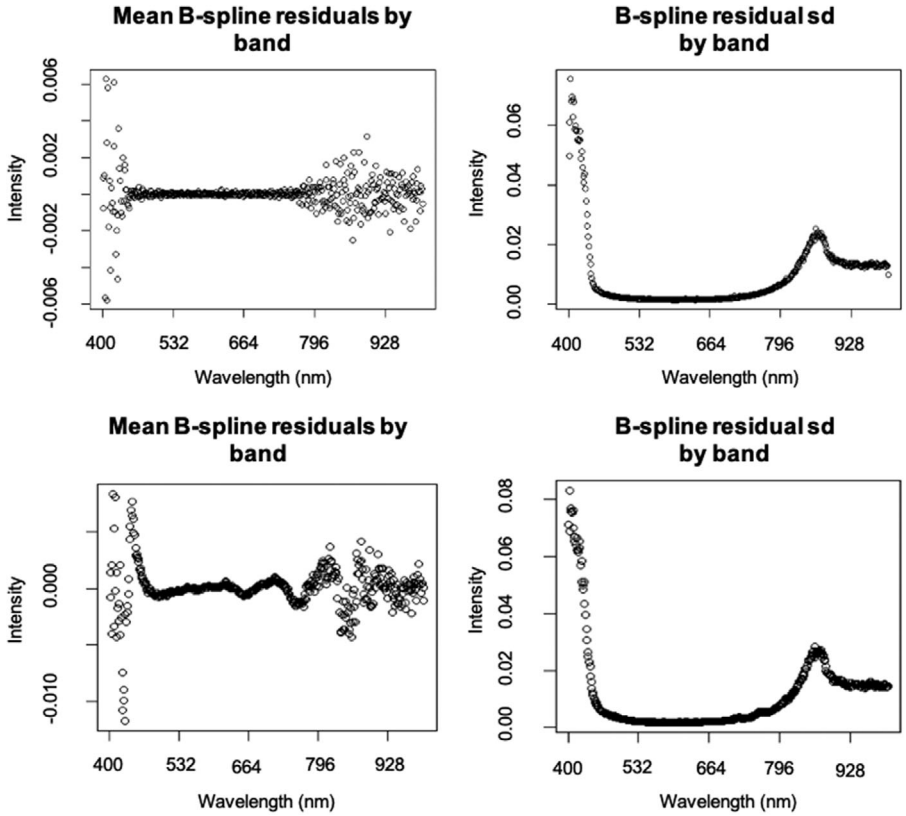

FIGURE 6 B-spline fitting, mean and standard deviation (*SD*) sample residuals
Top row: $\lambda = 17.78$; bottom row: $\lambda = 17782.78$

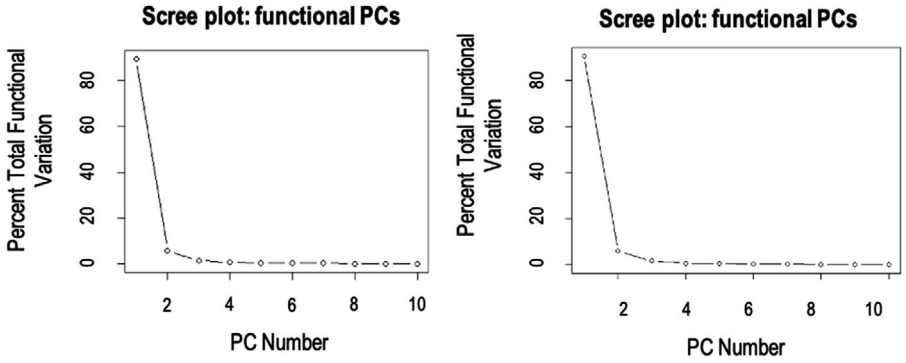

FIGURE 7 FPC scree plot: B-spline smoothed
Left: $\lambda = 17.78$; right: $\lambda = 17782.78$

over when $\lambda = 17782.78$. Clearly, the added smoothness comes with a cost in terms of bias, as is evident in Figure 6, where the top left plot shows the residual means by band when B-splines are fitted with $\lambda = 17.78$ and the bottom left plot shows the residual means by band when $\lambda = 17782.78$. We see that the residual means oscillate in a random pattern around 0 when λ is set to 17.78, indicating that signal is not being smoothed over; however, when $\lambda = 17782.78$, the residual mean pattern has distinct curvature across bands, indicating that some signal is being smoothed over. This suggests the fitted curves are slightly biased. It is interesting that, in tuning the FPCA-based algorithms through cross-validation, some models sacrifice bias for smoothness, and others sacrifice smoothness for bias; this indicates that it is important to tune FPCA-based algorithms over lambda, in order to let the data influence the optimal bias-smoothness tradeoff. Figure 7 shows a scree plot for B-spline-smoothed FPCs performed over curves fit using the fitting criterion specified in Equation (2) with $\lambda = 17.78$ (left) and $\lambda = 17782.78$ (right). For $\lambda = 17.78$, over 99% of the sample's total variation is accounted for by the first 8 FPCs and for $\lambda = 17782.78$, over 99% of the sample's total variation is accounted for in just the first 5 FPCs.

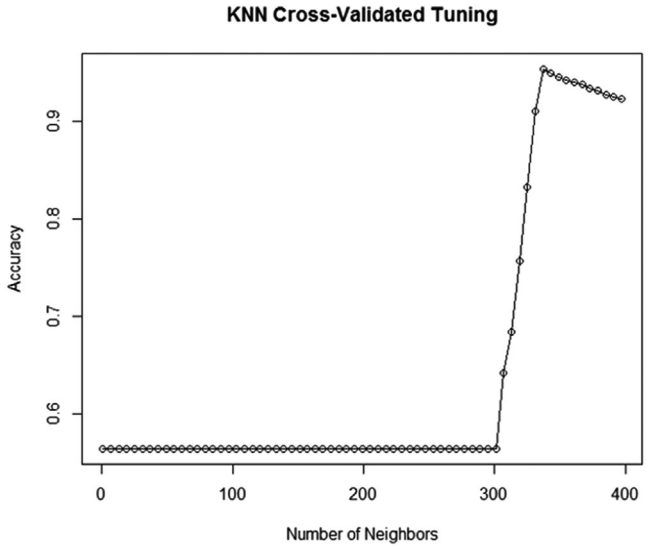

FIGURE 8 KNN model tuning on training set
KNN model five-fold cross-validation accuracy in training; accuracy increases sharply to its maximum and declines slowly afterwards

The KNN model's performance over a grid of k values in five-fold cross-validation on the training set is displayed in Figure 8; peak performance occurred at $k = 337$. The final model selected for testing set $k = 349$, since the plot makes clear that the performance cost of undershooting the truly optimal k value is higher than the cost of overshooting such a k value. Despite setting $k = 349$, 12 above the training set optimal k value, the truly optimal value of k was still undershot, and modelling accuracy was detrimentally impacted as a result: this model classified every pixel in the testing set as MWCNT(−) and therefore had 0% sensitivity and 100% specificity. In post hoc investigations, classification results were much improved when k was set to 375. Thus, we see that the tuning parameters for the KNN algorithm are pretty unstable.

4 | DISCUSSION

Overall, the best performing classification algorithms were the SVM and QDA algorithms that used PC-smoothed FPCA score data, the SVM model using B-spline-smoothed FPCA scores, and the SVM model using multivariate-PCA scores derived from average-normalised data; these were the only algorithms that did not have statistically significantly worse than perfect specificity and sensitivity. The B-spline-smoothed FPCA-based QDA model did not perform statistically significantly worse in terms of sensitivity and had only marginally worse specificity. This suggests that FPCA is an effective method for preprocessing hyperspectral microscopy data for classification, outperforming even the supervised feature design and selection implicit in the CNN algorithm, although it is possible that with more training data, CNN would have performed better. This provides evidence in practice that FPC scores effectively carry the angle and distance relationships between spectra from the linear space $L^2[w_1, w_2]$ of continuous pixel spectra to the Euclidean space where most classification models were designed to operate. Outside of these five algorithms, larger sacrifices of either sensitivity or specificity must be made. Importantly, these algorithms, and several others, performed much better than the SAM algorithm, which had poor sensitivity.

The KNN results indicate why SAM is underperforming in this context: most of the time, the k-nearest neighborhood of a MWCNT(+) pixel is filled with mostly MWCNT(−) pixels, until k exceeds roughly 4% of the training sample. Thus, if the SAM cosine angle distance threshold is raised enough to increase sensitivity, you are guaranteed a wave of false positives. Additionally, the QDA stepwise model proved to be a consistently good performer over the various data preprocessing methods tested. Also, the results suggest that the average normalisation preprocessing method is only advisable in the context of a flexibly nonlinear classification model like SVM, as this preprocessing step had deleterious effects on sensitivity when paired with models which construct a linear (LR net) or quadratic (QDA) classification boundary.

PCA and FPCA were also effective in reducing the dimensionality of the problem when paired with the supervised feature selection technique forward selection, as implemented with the QDA model. The B-spline-smoothed FPC and FPC-smoothed FPC QDA models used just five and four variables, respectively, and the multivariate PC QDA models on untransformed and average normalised data kept 3 and 10 variables, respectively. However, it should be noted that a *supervised* variable selection procedure was essential to achieving reduced data dimensionality without sacrificing classification accuracy. For example, QDA forward selection on B-spline-smoothed FPCs picked the 139th FPC as fourth most important to the classification problem. If one selected FPCs in an unsupervised manner, like using a scree plot 99% total sample variation cutoff, one would retain the first eight FPCs, and then leave out the important 139th FPC. Supervised feature selection techniques are thus strongly preferred in this context, as it seems the information important for classification may be contained in any PC. This makes the QDA model advantageous: a QDA model trains more quickly than the more complicated SVM model, making costly supervised variable selection techniques like forward-selection more feasible.

This work characterises the accuracy, sensitivity and specificity of an ensemble of schemes for EDFM-HSI pixel classification. We identify a few such schemes that

discriminate between MWCNTs and background filter media in filter-captured air samples with high accuracy. Once trained, these classification schemes, which are composed of data preprocessing and classification algorithms, can be implemented automatically on large batches of hyperspectral datacubes, without any attention from the end user. Thus, if these methods were coupled with a system that could automatically acquire a spatially random sample of images from a filter sample, then users could automate the time-consuming imaging and analysis processes associated with screening filter-captured air samples for MWCNTs. Given the high accuracy of the underlying classification schemes, this autoscreening process could be done with high sensitivity and specificity under a suitable image sampling design. However, it should be noted that the data presented here is based on idealised filter-captured air samples, which by design were minimally exposed to particles other than MWCNTs. This is an important first step for such a systematic, rigorous comparison of classification schemes, even though real-world filter-captured air samples are likely loaded with numerous particle species, including many that may not be of experimental interest. The classification schemes presented here are as yet unproven on these noisy, real-world samples. In any case, in future work, these schemes could be trained and tested as they are here, replacing the MWCNT(–) pixel samples from filter media blanks with MWCNT(–) pixel samples from real-world filter samples acquired in labs where MWCNTs are definitely absent.

From a practical standpoint, using a classification model that produces false negatives due to its low sensitivity would be worse than using a classification model that produces false positives due to its low specificity. In keeping with the precautionary principle, it would be preferable to overestimate potential worker exposure to ENMs – particularly in the absence of material-specific exposure limits – than underestimating potential worker exposure. The sensitivity and specificity of a classification model and the potential for misclassification therefore have real-world implications for worker health.

5 | CONCLUSIONS

Here, we provide objective evaluation of signal processing and classification algorithms for EDFM-HSI using MWCNTs as a test material. The methodology presented in this study for training and testing models for hyperspectral classification for HSI analysis of MWCNTs collected on filter media was successful in characterising the accuracy, sensitivity, and specificity of a suite of classification models. The present study has identified several models capable of classifying pixels as MWCNT(+) or MWCNT(–) in images of filter-captured air samples generated in the lab with high specificity and sensitivity. The SVM and QDA FPCA models performed equivalently well, and would make great options in this modelling context. The data reduction that was obtained with all four of the tested QDA + forward selection models might make these algorithms preferable; the untransformed multivariate PCA and the FPC-smoothed FPC QDA models, for example, used just three and four PCs, respectively, so that one can nearly visualise the entire classification problem. Additionally, the FPCA preprocessing method proved to be a useful tool for preprocessing hyperspectral data. In general, the success of these models suggests that EDFM-HSI may be reliably employed to assess filter-captured air samples taken from occupational environments for the presence of MWCNTs. Future work will further test the methods presented here on mixed-material filter samples used to capture airborne nanomaterials from real-world work environments.

ACKNOWLEDGEMENTS

This research effort was financially supported by the U.S. Centers for Disease Control-National Institute for Occupational Safety and Health (CDC-NIOSH). N.M. Neu-Baker received funding from NIOSH through Interagency Personnel Assignments (14IPA1418072 & 18IPA1816710-M03) and a contract (No. 200-2017-M-94751). The authors thank CytoViva, Inc. for technical support and guidance. Additionally, the authors thank Nathan Drew, MS (NIOSH); Robin Dunkin, MS (NIOSH); Stephen Bertke, PhD (NIOSH); Dhimiter Bello, ScD (University of Massachusetts – Lowell) and Waldyn Martinez, PhD (Miami University of Ohio) for review of this manuscript.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

DISCLAIMER

The findings and conclusions in this report are those of the authors and do not necessarily represent the views of the National Institute for Occupational Safety and Health (NIOSH).

ORCID

Nicole M. Neu-Baker https://orcid.org/0000-0002-6820-8954

REFERENCES

1. CDC-NIOSH. (2007). Progress toward safe nanotechnology in the workplace. DHHS (NIOSH) Publication No. 2007–123. Cincinnati, OH, USA
2. CDC-NIOSH. (2013). Current strategies for engineering controls in nanomaterial production and downstream handling

processes. DHHS (NIOSH) Publication No. 2014–102. Cincinnati, OH, USA
3. CDC-NIOSH. (1994). Asbestos by TEM. Method 7402. *NIOSH Manual of Analytical Methods* (NMAM), 4th ed., DHHS (NIOSH) Publication 94–113, Cassinelli, M.E. & O'Connor, P.F., Eds.
4. ASTM D5755-03. (2003). Standard test method for microvacuum sampling and indirect analysis of dust by transmission electron microscopy for asbestos structure number surface loading. ASTM International, West Conshohocken, PA, https://www.astm.org.
5. Roth, G. A., Tahiliani, S., Neu-Baker, N. M., & Brenner, S. A. (2015). Hyperspectral microscopy as an analytical tool for nanomaterials. *Wiley Interdisciplinary Reviews: Nanomedicine and Nanobiotechnology*, *7*, 565–579.
6. Sosa Peña, M. P., Gottipati, A., Tahiliani, S., Neu-Baker, N. M., Frame, M. D., Fridman, A. J., & Brenner, S. A. (2016). Hyperspectral imaging of nanoparticles in biological samples: Simultaneous visualization and elemental identification. *Microscopy Research and Technique*, *79*, 349–358.
7. Idelchik, M. P. S., Neu-Baker, N. M., Chandrasekaran, A., Friedman, A. J., Frame, M. D., & Brenner, S. A. (2016). Relative quantitation of metal oxide nanoparticles in a cutaneous exposure model using enhanced darkfield microscopy and hyperspectral mapping. *NanoImpact*, *3–4*, 12–21.
8. Dillon, J. C. K., Bezerra, L., Sosa Pena, M. P., Neu-Baker, N. M., & Brenner, S. A. (2017). Hyperspectral data influenced by sample matrix: The importance of building relevant reference spectral libraries to map materials of interest. *Microscopy Research and Technique*, *80*, 462–470.
9. Badireddy, A. R., Budarz, J. F., Chellam, S., & Wiesner, M. R. (2012). Bacteriophage inactivation by UV-A illuminated fullerenes: Role of nanoparticle-virus association and biological targets. *Environmental Science & Technology*, *46*, 5963–5970.
10. Badireddy, A. R., Wiesner, M. R., & Liu, J. (2012). Detection, characterization, and abundance of engineered nanoparticles in complex waters by hyperspectral imagery with enhanced darkfield microscopy. *Environmental Science & Technology*, *46*, 10081–10088
11. Anderson, D. S., Patchin, E. S., Silva, R. M., Uyeminami, D. L., Sharmah, A., Guo, T., … Van Winkle, L. S. (2015). Influence of particle size on persistence and clearance of aerosolized silver nanoparticles in the rat lung. *Toxicological Sciences*, *144*, 366–381.
12. Mortimer, M., Gogos, A., Bartolomé, N., Kahru, A., Bucheli, T. D., & Slaveykova, V. I. (2014). Potential of hyperspectral imaging microscopy for semi-quantitative analysis of nanoparticle uptake by protozoa. *Environmental Science & Technology*, *48*, 8760–8767.
13. Chang, C. -I. (2013). *Hyperspectral data processing: Algorithm design and analysis*. Hoboken, NJ, USA: John Wiley & Sons.
14. Roth, G. A., Sosa Peña, M. P., Neu-Baker, N. M., Tahiliani, S., & Brenner, S. A. (2015). Identification of metal oxide nanoparticles in histological samples by enhanced darkfield microscopy and hyperspectral mapping. *Journal of Visualized Experiments*, *106*, e53317.
15. Neu-Baker, N. M., Eastlake, A. C., & Brenner, S. A. (2019). Sample preparation method for visualization of nanoparticulate captured on mixed cellulose ester filter media by enhanced darkfield microscopy and hyperspectral imaging. *Microscopy Research and Technique*, *82*, 878–883.
16. Guttenberg, M., Bezerra, L., Neu-Baker, N. M., Idelchik, M. P. S., Elder, A., Oberdörster, G., & Brenner, S. A. (2016). Biodistribution of inhaled metal oxide nanoparticles mimicking occupational exposure: A preliminary investigation using enhanced darkfield microscopy. *Journal of Biophotonics*, *9*, 987–993
17. Holm, S. (1979). A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, *6*(2), 65–70.
18. Kuhn, M., & Johnson, K. (2013). *Applied predictive modeling*. New York: Springer-Verlag.
19. Izenman, A. J. (2008). *Modern multivariate statistical techniques: Regression, classification, and manifold learning*. New York: Springer.
20. Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press, Cambridge, MA, USA: MIT Press.
21. Hu, W., Huang, Y., Wei, L., Zhang, F., & Li, H. (2015). Deep convolutional neural networks for hyperspectral image classification. *Journal of Sensors*, *2015*, Article ID 258619.
22. Slavkovikj, V., Verstockt, S., De Neve, W., Van Hoecke, S., & Van de Walle, R. (2015). Hyperspectral image classification with convolutional neural networks. *Proceedings of the 23rd ACM International Conference on Multimedia*, Association for Computing Machinery, New York, NY, USA, pp. 1159–1162.
23. Mei, S., Ji, J., Bi, Q., Hou, J., Du, Q., & Li, W. (2016). Integrating spectral and spatial information into deep convolutional neural networks for hyperspectral classification. *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, Beijing, China, pp. 5067–5070.
24. Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning*, Proceedings of Machine Learning Research. pp. 448–456.
25. Falbel, D., Allaire, J. J., Chollet, F., Tang, Y., Van Der Bijl, W., Studer, M., & Keydana, S. (2019). Keras: R interface to "Keras." R package 2.2.5.0.
26. Vermeulen, P., Pierna, J., Van Egmond, H., Dardenne, P., & Baeten, V. (2012). Online detection and quantification of ergot bodies in cereals using near infrared hyperspectral imaging. *Food Additives & Contaminants: Part A*, *29*, 232–240.

SUPPORTING INFORMATION

Additional supporting information may be found online in the Supporting Information section at the end of the article.

How to cite this article: Smith D, Neu-Baker NM, Eastlake AC, Zurbenko IG, Brenner SA. Evaluation of classification methods for identifying multiwalled carbon nanotubes collected on mixed cellulose ester filter media. *Journal of Microscopy*. 2021;*283*:102–116. https://doi.org/10.1111/jmi.13012