Electronic Supplementary Material (ESI) for Nanoscale Advances.

Supplementary Information

Machine learning-augmented surface-enhanced spectroscopy toward next-generation molecular diagnostics

Hong Zhou[1,2,‖], *Liangge Xu*[1,2,3,‖], *Zhihao Ren*[1,2], *Jiaqi Zhu*[3]*, *Chengkuo Lee*[1,2,4]*

[1] Department of Electrical and Computer Engineering, National University of Singapore, Singapore 117583;

[2] Center for Intelligent Sensors and MEMS (CISM), National University of Singapore, Singapore 117608;

[3] National Key Laboratory of Special Environment Composite Technology, Harbin Institute of Technology, Harbin, 150001, China

[4] NUS Suzhou Research Institute (NUSRI), Suzhou 215123, China;

[‖] These authors contributed equally to this work.

* Corresponding Author: Chengkuo Lee (email: elelc@nus.edu.sg); Jiaqi Zhu (email: zhujq@hit.edu.cn);

Note S1. Principles of SERS and SEIRA

The principles of SERS and SEIRA include electromagnetic field enhancement and chemical effect. The underlying mechanisms of electromagnetic field enhancement are mainly about the interaction of molecules and plasmons excited in a SERS/SEIRA substrate. The performance metric that describes this enhancement effect is the enhancement factor G_{SERS} and G_{SEIRA}. Figure 2a in the main text shows the conventional Raman scattering $S_{\omega r}$ with incident radiation $R_{\omega 0}$. Raman spectrum provides rich information about chemical structure and identity, phase and polymorphism, intrinsic stress/strain, and contamination and impurity, but the intensity of Raman scattering is weak when detecting trace amounts of molecules, limiting its application in many scenarios. The SERS enhancement factor is calculated by

$$G_{SERS} = \frac{I_{SERS}(\omega_R)}{I_{CR}(\omega_R)} \tag{1}$$

where $I_{SERS}(\omega_R)$ and $I_{CR}(\omega_R)$ are the total Raman intensity of SERS and conventional Raman, respectively. $I_{SERS}(\omega_R)$ is determined by the induced dipole $\boldsymbol{p}_m(\omega_R, \boldsymbol{r}_m)$ at the Raman scattering frequency (ω_R) and the induced dipole of the plasmonic antennas $\boldsymbol{p}_A(\omega_R, \boldsymbol{r}_A)$. $\boldsymbol{p}_m(\omega_R, \boldsymbol{r}_m)$ depends on the electric field strength for the excitation of the target molecules and Raman polarizability derivatives, that is,[1-3]

$$\boldsymbol{p}_m(\omega_R, \boldsymbol{r}_m) = \boldsymbol{\alpha}_m^I(\omega_R, \omega_0) \bullet \boldsymbol{EL}(\omega_0, \boldsymbol{r}_m) \tag{2}$$

$$\boldsymbol{EL}(\omega_0, \boldsymbol{r}_m) = g_1(\omega_0, \boldsymbol{r}_m) \boldsymbol{E}_R(\omega_0) \tag{3}$$

where $\boldsymbol{EL}(\omega_0, \boldsymbol{r}_m)$ and $\boldsymbol{E}_R(\omega_0)$ are the local and incident electric field strength at the position $\boldsymbol{r}_m$ in which plasmonic antennas are present and absent to enhance molecular signals. The $g_1(\omega_0, \boldsymbol{r}_m)$ represents the enhancement factor of incident electric field strength. The $\boldsymbol{\alpha}_m^I(\omega_R, \omega_0)$ is the Raman polarizability derivative when the incident light frequency is ω_0 and the Raman scattering frequency is ω_R. To obtain high electric field strength for molecule detection, either increasing the incident light intensity $\boldsymbol{E}_R(\omega_0)$ or local electric field intensity $\boldsymbol{EL}(\omega_0, \boldsymbol{r}_m)$ is possible. However, high-power incident laser light can cause surface damage to the analyte, limiting its sensitivity to a certain threshold. The strategy of increasing $g_1(\omega_0, \boldsymbol{r}_m)$ by plasmonic antennas could significantly increase the local interaction power between photons and molecules near the antenna with less laser power, which is also the key advantage of SERS. For the antenna at $\boldsymbol{r}_A$, it is also locally excited by the nearby point dipolar source $\boldsymbol{p}_m(\omega_R, \boldsymbol{r}_m)$, and then its local electric field intensity at $\boldsymbol{r}_A$ is expressed as

$$\boldsymbol{EL}(\omega_R, \boldsymbol{r}_A) = C_2 \cdot \boldsymbol{p}_m(\omega_R, \boldsymbol{r}_m) \tag{4}$$

where C_2 is determined by the relative position of plasmonic antennas and molecules. Then in the induced dipole approximation, the induced dipole of the antennas $\boldsymbol{p}_A(\omega_R, \boldsymbol{r}_A)$ is calculated by

$$\boldsymbol{p}_A(\omega_R, \boldsymbol{r}_A) = \boldsymbol{\alpha}_A(\omega_R) \cdot \boldsymbol{EL}(\omega_R, \boldsymbol{r}_A) = C_2 \boldsymbol{\alpha}_A(\omega_R) \boldsymbol{p}_m(\omega_R, \boldsymbol{r}_m) \tag{5}$$

Then, the additive local source $\boldsymbol{p}(\omega_R)$ that excites signals at far-field could be expressed by

$$\boldsymbol{p}(\omega_R) = \boldsymbol{p}_m(\omega_R, \boldsymbol{r}_m) + \boldsymbol{p}_A(\omega_R, \boldsymbol{r}_A) \tag{6}$$

By substituting Equation 5 to Equation 6, $\boldsymbol{p}(\omega_R)$ can be written as

$$\boldsymbol{p}(\omega_R) = \boldsymbol{p}_m(\omega_R, \boldsymbol{r}_m)(1 + C_2 \alpha_A(\omega_R)) = \boldsymbol{p}_m(\omega_R, \boldsymbol{r}_m) \cdot g_2(\omega_R, \boldsymbol{r}_A) \tag{7}$$

Then, by substituting Equations 2 and 3 to Equation 7, we can obtain

$$\boldsymbol{p}(\omega_R) = g_1(\omega_0, \boldsymbol{r}_A) \times \left[g_2(\omega_R, \boldsymbol{r}_A) \alpha_m^1(\omega_R, \omega_0) \cdot \boldsymbol{E}_R(\omega_0) \right] \tag{8}$$

The total Raman intensity $I_{SERS}(\omega_R)$ is proportional $|\boldsymbol{p}(\omega_R)|^2$, that is,

$$I_{SERS} \propto |g_1(\omega_0, \boldsymbol{r}_A)|^2 \cdot \left| g_2(\omega_R, \boldsymbol{r}_A) \boldsymbol{\alpha}_m^1(\omega_R, \omega_0) \cdot \boldsymbol{E}_R(\omega_0) \right|^2 \tag{9}$$

For the conventional Raman case, the total Raman intensity can be expressed as

$$I_{CR} \propto \left| \boldsymbol{\alpha}_m^1(\omega_R, \omega_0) \cdot \boldsymbol{E}_R(\omega_0) \right|^2 \tag{10}$$

By substituting Equations 9 and 10 to Equation 1, we can obtain the SERS enhancement factor as

$$G_{SERS} = |g_1(\omega_0, \boldsymbol{r}_m)|^2 \cdot |g_2(\omega_R, \boldsymbol{r}_A)|^2 = \left| \frac{\boldsymbol{EL}(\omega_0)}{\boldsymbol{E}_R(\omega_0)} \right|^2 \cdot \left| \frac{\boldsymbol{ES}(\omega_R)}{\boldsymbol{S}(\omega_R)} \right|^2 \tag{11}$$

As observed in Equation 11, the SERS enhancement involves two steps. First, the local field is enhanced by the localized surface plasmon polariton (LSP) of nanoparticles, that is, Local field enhancement of incident light ($|g_1(\omega_0, \boldsymbol{r}_m)|^2$). Then, the excitation and radiation efficiency of far-field Raman scattering is improved by the interaction with LSP ($|g_2(\omega_R, \boldsymbol{r}_A)|^2$).

Similarly, the local field around molecules for conventional IR spectrum can be expressed as

$$I_{IR} \propto |\mu(\omega_k) \cdot \boldsymbol{A}(\omega_k)|^2 \tag{12}$$

where $\mu(\omega_k)$ represents the electric dipole derivative concerning the kth vibrational normal modes at the IR absorption frequency (ω_k). The total IR intensity $I_{IR}(\omega_k)$ is obtained by

$$I_{SEIRA} \propto |\mu(\omega_k) \cdot \boldsymbol{EA}(\omega_k)|^2 = |g_1(\omega_k, \boldsymbol{r}_m) \mu(\omega_k) \cdot \boldsymbol{A}(\omega_k)|^2 \tag{13}$$

Therefore, the SEIRA enhancement factor is calculated as

$$G_{SERS} = \left| g_1(\omega_k, \boldsymbol{r}_m) \right|^2 = \left| \frac{\boldsymbol{EA}(\omega_k)}{\boldsymbol{A}(\omega_k)} \right|^2 \quad (14)$$

Note S2. Machine learning algorithms

2.1 Genetic algorithm (GA)

GA is an adaptive search procedure based on natural genetics and natural selection mechanism, which has been widely used for solving both constrained and unconstrained optimization problems.[4, 5] GA first generates a random population, where each individual in the population represents a possible solution and is coded as a "0" or "1", also known as a "chromosome". "1" represents the selected variable and "0" is the unselected variable. Then, the "fitness" of each individual is calculated for evaluation. After evaluation, the next population is obtained by genetic manipulation. After many iterations, the individuals in the population gradually solve the target level. The main advantage of GA is that it can find high-quality solutions in a very short computational time. Therefore, GA is often used for antenna design in SEIRA.[6]

2.2 Principal component analysis (PCA)

PCA is one of the most commonly used algorithms for SEIRA/SERS due to its function of dimension reduction. The main idea of PCA is to reduce the dimensionality of a dataset consisting of many interrelated variables, while maximizing the preservation of features in the dataset.[7-9] It can effectively reduce the pressure on signal processing caused by large amounts of data, thereby improving computational efficiency.[10] Generally, PCA is used as a preprocessing step for classification algorithms such as LDA and SVM. PCA is an unsupervised method that allows data to be inspected without pre-existing bias. The advantages of PCA include ease of computation, speeding up other ML algorithms, and offsetting problems with high-dimensional spectral data.[11] In the practical application of SEIRA/SERS, it is necessary to consider the impact of the low interpretability of principal components and the trade-off between information loss and dimensionality reduction.

2.3 Support vector machine (SVM)

SVM is a kind of supervised ML algorithm that is often used for the classification and regression analysis of two groups of data points.[12] It looks for a hyperplane separating the two

classes of data points with the largest margin.[13] SVMs can be linear and nonlinear according to the hyperplane shape.[14] SVM works by mapping training data to points in space, then maximizing the gap width between two categories, and finally predicting which category the new dataset falls on based on the gap side the new dataset falls on. The advantage of SVM is that it is efficient when the space is high-dimensional or the number of dimensions is greater than the number of data. Notably, when the number of features is much larger than the number of datasets, it is necessary to prevent overfitting when choosing the kernel function, and the regularization term is crucial. SVM is extensively used in SEIRA/SERS, such as in the detection and classification of small molecules, biomarkers, tumour cells, pathogens, and so on.[15-18]

2.4 Linear discriminant analysis (LDA)

LDA is a supervised ML algorithm that for solving more than two-class classification problems by transferring features from higher to lower dimension spaces.[19] The function of dimensionality reduction allows it to be used as a preprocessing step for some classification applications.[20] LDA is not only a dimensionality reduction tool, but also a robust classification method. Compared to SVM, LDA generally produces robust, decent, and interpretable classification results. Therefore, it is also used as a benchmarking method before the implementation of other complex methods. In SERS, LDA could be used to separate SERS spectra of multiple analytes with high precision. Furthermore, it can determine the intervals in the spectrum with the greatest contribution of spectral features, thereby assessing their contribution to the spectrum of each analyte.[21]

2.5 κ-nearest neighbor (κNN).

κNN is a non-parametric supervised ML algorithm that uses proximity to classify or predict groupings of individual data points.[22] The algorithm is nonparametric, which means it makes no assumptions about the underlying data. The core idea of the κNN algorithm is to assume that similar things are very close. The choice of neighbor number (κ) is based on the data set. When the dataset is small, the classification of the dataset by κNN is simple and accurate. However, as the dataset grows, κNN becomes increasingly inefficient, affecting the overall model performance. Recently, κNN was used for SERS-based breast cancer detection to improve classification accuracy.[23-25]

2.6 Random forest (RF)

RF is a supervised ML algorithm consisting of decision tree algorithms for solving regression and classification problems.[26] The ensemble of decision trees builds the "forest" of the RF algorithm. "Forests" are trained by bagging or bootstrap aggregating, where bagging is an ensemble meta-algorithm that improves the accuracy of ML algorithms. Since the prediction results depend on the average of the outputs of various trees, the accuracy of the results can be improved by increasing the number of trees. The advantages of the RF algorithm include: a) more accurate than decision tree algorithms; b) providing an efficient way to deal with missing data; c) producing reasonable predictions without hyper-parameter tuning; d) being free from overfitting. Recently, RF was used to identify significant SERS signals, evaluate the correlation of predefined spectral groups, and further analyze the relationship of different SERS signals.[27-29] Therefore, RF is promising for the sophisticated analysis of complex biological samples.

2.7 Artificial neural networks (ANN)

Neural networks, whose name and structure are inspired by the human brain, process data by mimicking the way biological neurons transmit signals to each other.[30, 31] ANNs are supervised learning algorithms that can be classified into single-layer, multi-layer, and recurrent networks.[32] ANN usually consists of an input layer, one or more hidden layers, and an output layer. Each node is connected to another node through weights and thresholds. When data enters a node, the output of the node is compared with a set threshold. If the output is above the threshold, the node will be activated and send the data to the next layer of the network. If the output is below the threshold, the data is not passed to the next layer of the network. The advantages of the ANN algorithm include: a) storing information on the entire network; b) ability to work with incomplete knowledge; c) high fault tolerance, that is, damage to one or more cells does not prevent it from generating output; d) parallel processing capability to perform more than one job at the same time. Recently, ANN was utilized to identify significant SEIRA signals and discriminate accurately proteins, nucleic acids, carbohydrates, and lipids.[33-35]

2.8 Convolutional neural network (CNN)

CNN is a supervised deep learning algorithm that is good at processing data with grid patterns,

so it is widely used in the field of image computer vision.[36] CNN has three main types of layers, which are convolutional layer, pooling layer, and fully-connected layer. It uses the principles of linear algebra, especially matrix multiplication, to identify patterns in images.[37] The advantages of the CNN algorithm include: a) high accuracy in image recognition problems; b) automatically detecting the important features without any human supervision; c) weight sharing. Although the output of SERS/SEIRA is a two-dimensional spectrum, its output can form an image when its application is related to time, groups of concentrations, or multiple analytes. Therefore, CNN can be used to assist SERS and SEIRA in efficiently detecting and identifying analytes. Recently, some work reported that the SERS control spectra of normal and cancer cell metabolites were classified by the ANN algorithm, and the prediction accuracy reached 100%.[38-40]

Note S3. Overview of the mathematics behind machine learning algorithms

The application of ML algorithms in SEIRA/SERS mainly includes regression, clustering, dimensionality reduction, and classification. Here we briefly introduce the mathematical formula derivation of four typical algorithms.

3.1 Linear Regression

In linear regression, the output h(x) is linearly dependent on input x, hence we can create a hypothesis that can be resembled the equation of a straight line, that is,

$$h(x) = \theta_0 + \theta_1 x \tag{15}$$

where (θ_0) and (θ_1) are called regression coefficients. To predict values more precisely, the value of θ_0 and θ_1 are important. That is, we need to obtain the optimal value of θ_0 and θ_1 from the training set to minimize the difference between the measured result y and predicted result h(x). It can be expressed as $\sum_{i=1}^{i=m}(h_0(x^i)-y^i)^2$, where m is the number of records present in our dataset. Therefore, we need to reduce the squared error f_n of the hypothetical model, which is also called cost function $C(\theta_0, \theta_1)$,

$$C(\theta_0,\theta_1) = f_n = \frac{1}{2m}\sum_{i=1}^{i=m}(h_0(x^i)-y^i)^2 \tag{16}$$

One of the techniques that help to find optimal θ_0 and θ_1 is gradient descent. The process is: 1) pick random values of θ_0 and θ_1; 2) keep on simultaneously updating values of θ_0 and θ_1 till the convergence; 3) if the cost function does not decrease anymore, we reached our local minima. It can be expressed using the formula below,

$$temp0 = \theta_0 - \alpha\frac{\partial}{\partial\theta_0}J(\theta_0,\theta_1) \tag{17}$$

$$temp1 = \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1) \tag{18}$$

$$\theta_0 = temp0 \tag{19}$$

$$\theta_1 = temp1 \tag{20}$$

where α is the learning rate. By using the training set to train the above model, the optimal value of θ_0 and θ_1 can be obtained. The above linear regression can be extended to multiple linear regression (MLR) by adding multiple independent features x. The hypothesis will change to

$$h(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + ... + \theta_n x_n \tag{21}$$

And the cost function will be modified to

$$C(\theta_0, \theta_1, ...\theta_n) = f_n = \frac{1}{2m} \sum_{i=1}^{i=m} (h_0(x^i) - y^i)^2 \tag{22}$$

The gradient descent simultaneous update will change to

$$\theta_j = \theta_j - \alpha \frac{\partial}{\partial \theta_j} C(\theta_0, \theta_1, ...\theta_n) \tag{23}$$

where j=0,1,2…n. The demonstration in Figure 5b-v used MLR as the algorithm for dynamic biomonitoring.

3.2 Dimensionality reduction

Principal component analysis (PCA) is the typical technique for dimensionality reduction. The measured dataset is set as $\boldsymbol{Z}$. The first step of PCA is the standardization of the continuous variables of the dataset as

$$\mathbf{X} = \frac{Z - \mu}{\sigma} \tag{24}$$

$$\mu = \frac{1}{N} \sum_{i=1}^{N} Z_i \tag{25}$$

$$\sigma^2 = \frac{1}{N} \sum_{i=1}^{N} (Z_i - \mu)^2 \tag{26}$$

The second step is to construct the covariance matrix by doing a simple matrix operation on the input matrix $\boldsymbol{X}$, as shown in the following formula,

$$\mathbf{A} = cov(\mathbf{X}, \mathbf{Y}) = \frac{1}{N-1} \sum_{i=1}^{N} (X_i - \mu_X)(Y_i - \mu_Y) \tag{27}$$

where X_i and Y_i are the specific training dataset from variables X and Y, and μ_x and μ_y are the means

of the variables. The next step is to compute eigenvectors and corresponding eigenvalues. In general, the eigenvector of a matrix **A** is the vector for which the following holds:

$$\mathbf{A}\vec{v} = \lambda\vec{v} \tag{28}$$

where λ is the eigenvalue and $\vec{v}$ is the eigenvector. We set $\lambda_1 > \lambda_2 > \lambda_3 > \cdots$, Then, the 1st PC v_1 is the eigenvector of the sample covariance matrix **A** associated with the largest eigenvalue. The 2nd PC v_2 is the eigenvector of the sample covariance matrix **A** associated with the second largest eigenvalue.

In the last step, the samples are transformed to the new subspace by reorienting the dataset from the original axis to the axis now represented by the principal components. It can be expressed as

$$\mathbf{D}=[v_1,v_2\ldots]\cdot\mathbf{X} \tag{29}$$

In conclusion, steps involved in PCA are

- Standardization of the continuous variables of the dataset.
- Computing the Co-variance Matrix to identify Co-relations.
- Computing the Eigen Values and Eigenvectors of the covariance Matrix to identify the Principal Components.
- Deciding on the Principal Components to be kept for further analysis based on the variation in the Components using the Scree Plot.
- Recast the data along the Principal Component's axes.

3.3 Classification

Support vector machine (SVM) is a typical classification algorithm that classifies input data into two classes by calculating the distance between data groups and maximizing the gap between them. The measured dataset is set as **X1** and **X2**, as shown in Figure 3a of the main text. Then a hyperplane in SVM is developed to separate the two classes. The main task of the SVM model is to find the best hyperplane to classify the classes. The hyperplane can be expressed by the following equation.

$$\boldsymbol{\omega}\cdot\mathbf{x}-b=0 \tag{30}$$

where $\boldsymbol{\omega}$ is a weights that determines the orientation of the hyperplane and b is the bias. The distance between hyperplane and origin is the value of the bias divided by the length of the normal vector, which can be expressed as

$$D = \frac{|b|}{\left\|\vec{\omega}\right\|} = \frac{|b|}{\sqrt{\omega_1^2 + \omega_2^2 + \ldots}} \tag{31}$$

The distance of any point x in **X1** or **X2** to the hyperplane is calculated as

$$d = \frac{\left|\boldsymbol{\omega}^T \cdot \mathbf{x} - b\right|}{\|\boldsymbol{\omega}\|} \tag{32}$$

The training task is to maximize the distance. When making predictions on training data classified as positive and negative groups, a value greater than 0 will be obtained if points are replaced from the positive group in the hyperplane equation. It can be expressed as

$$\boldsymbol{\omega} \cdot \mathbf{x} - b > 0 \tag{33}$$

Predictions from the negative group in the hyperplane equation would give negative value as

$$\boldsymbol{\omega} \cdot \mathbf{x} - b < 0 \tag{34}$$

3.4 Clustering

K-means clustering is the most common clustering algorithms. The *K*-means algorithm mainly performs two tasks: 1) determine the best value for K center points, and 2) minimize pairwise distances of data points within the same cluster. For the measured dataset $\mathbf{X} = [x_1, x_2, \ldots x_n]$, these points can be divided into k clusters $C_1, \ldots C_k$ according to the objective below

$$\min_{C_1, \ldots C_k} \sum_{r=1}^{k} \frac{1}{|C_r|} \sum_{i,j \in C_r} \left\|x_i - x_j\right\|^2 \tag{35}$$

This cost function is a weighted average of the cluster variances. Its weight is proportional to the cluster size, expressed in points $|C_r|$. To derive it, we set $\mu_r = \frac{1}{|C_r|}\sum_{i \in C_r}^{n} x_i$ be the centroid. Note that

$$\begin{aligned} \sum_{i,j \in C_r} \left\|x_i - x_j\right\|^2 &= \sum_{i,j \in C_r} (\|x_i\|^2 + \left\|x_j\right\|^2 - 2\langle x_i, x_j \rangle) \\ &= 2|C_r| \sum_{i,j \in C_r} \|x_i\|^2 - 2|C_r|^2 \|\mu_r\|^2 \end{aligned} \tag{36}$$

Besides,

$$\begin{aligned} \sum_{i \in C_r} \|x_i - \mu_r\|^2 &= \sum_{i \in C_r} (\|x_i\|^2 + \|\mu_r\|^2 - 2\langle x_i, \mu_r \rangle) \\ &= \sum_{i \in C_r} \|x_i\|^2 - |C_r|^2 \|\mu_r\|^2 \end{aligned} \tag{37}$$

Therefore, according to Equations (36)-(37), we obtain

$$\frac{1}{2}\sum_{i,\, j\in C_r}\left\|x_i - x_j\right\|^2 = |C_r|\sum_{i\in C_r}\left\|x_i - \mu_r\right\|^2 \tag{38}$$

Equations (35) can be derived as

$$\min_{\substack{C_1,\ldots C_k \\ \mu_1,\ldots \mu_k}} \sum_{j=1}^{k}\sum_{i\in C_j}\left\|x_i - \mu_j\right\|_2^2 \tag{39}$$

Then, the following steps are used to find the solution to the k-means objective (39).

- Choose k initial cluster centers $\mu_1,\ldots,\mu_k$.
- Assign each point x_i to its correct cluster C_j according to $j = \text{argmin}\left\|x_i - \mu_j\right\|_2^2$.
- Update the centers μ_j based on the new clusters.
- Repeat above two steps until convergence to some stopping criterion.

Table S1 Summary of machine learning-enhanced SEIRA and SERS application

Types	Ref.	Substrates	Algorithms	Analytes	Function	Performance metrics
SEIRA	John-Herpin *et al.*[33]	Nanorod antennas	DNN	Proteins, nucleic acids, carbohydrates, lipids	Analyte discrimination and data-processing	4 classes of biomolecules
	Ren *et al.*[41]	Wavelength-multiplexed hook nanoantennas	PCA, SVM	methanol, ethanol, isopropanol	Analyte discrimination and classification	100% identification accuracy
	Kühner *et al.*[42]	Nanorod antennas	PCA	Glucose, Fructose	Analyte discrimination	10 g/L detection limit
	Meng *et al.*[43]	Band-stop and band-pass antennas	Frame averaging, PCA, SVM	C_2H_2, C_2H_4, C_2H_6, NH_3, O_3, SO_2)	Chemical Classifier	6 classes of chemicals
	Li *et al.*[6]	Digitized binary antennas	GA	COVID-19	Design assisting	1.66%/nm sensitivity
	Nadell *et al.*[44]	Cylindrical antennas	DNN	/	Design assisting	1.16×10^{-3} average mean squared error
	Jafar-Zanjani et al.[45]	Digitized binary antennas	Adaptive GA	/	Design assisting	±45° field-of-view
	Phan *et al.*[46]	Graphene antennas	DNN	/	Inverse Design	2 hidden layers
	Kalinin et al.[47]	Self-assembled antenna arrays	DNN	/	Design assisting	4 hidden layers
	Corcione *et al.*[48]	Nanorod antennas	ANN, Gaussian process regression	Glucose	Data-processing	0.47 g/L RMS error
	Meng *et al.*[49]	Band-stop and band-pass antennas	PCA, SVM	paracetamol, ibuprofen, aspirin, oil	Analyte discrimination	6 classes of chemicals
	Kyoung *et al.*[50]	Nanoslot antennas	k-NN	Protein A/G and IgG	Extracting complex refractive index	
SERS	William Cheung *et al.*[51]	Gold Colloid Solution	PCA, PLSR, ANNs, SVR	Sudan-1	Quantitative Analysis	10^{-3} to 10^{-4} mol L^{-1}
	Wu *et al.*[52]	Ag Nanoparticl	PCA	Carmine dye	Quantitative analysis	10^{-8} M

	es				
Ai *et al.*[53]	Silver nanoparticles	PCA	Food colorants	Qualitative and quantitative determination	10^{-8} mol/L
Weng *et al.*[54]	Gold nanorods	PLSR, SVMR, RF, PCA	Pirimiphos-Methyl	Quantitative analysis	0.25mg/L
Weng *et al.*[55]	Silver nanoparticles	RF	Fenthion	Quantitative analysis	0.05mg/L
Li *et al.*[56]	Silver nanoparticles	LS, SVM	Thiophanate-methyl and carbendazim	Analyte discrimination	R^2_P of 0.986
Dies *et al.*[57]	Silver nanoparticles	PCA, SVM	Drugs	Analyte discrimination and Quantitative analysis	100% identification accuracy and LOD100 ppb
Reza *et al.*[58]	Silver nanoparticles	PCA	Heroin and methamphetamine	Analyte discrimination	95% confidence intervals
Reshma *et al*[59]	Gold nanorods	PCA	Crystal violet and picric acid	Quantitative analysis	94.72% determination coefficients,Computation times < 10s
Thrift *et al.*[60]	gold nanoparticles	CNN	Rhodamine 6G, methylene blue	Quantitative analysis	10 fM, R^2 of 0.958
Bao *et al.*[61]	Silver nanoparticles	PAC, SVM	Flibanserin i	Analyte discrimination and Quantitative analysis	1 μg mL^{-1}, 92.3%, 91.7% and 92.0%
Li *et al.*[62]	Silver nanoparticles	LS-SVM	Thiophanate-methyl	Quantitative analysis	RPD = 6.08, R^2_P = 0.986 and RMSEP = 0.473
Cheung *et al.*[63]	Gold nanorods	PCA, PLS regression, ANNs, SVR	Food dye Sudan-1	Quantitative analysis	10^{-4} mol L^{-1}, R^2 > 0.965
Dies *et al.*[64]	Silver nanoparticles	PCA, SVM	Cocaine	Analyte discrimination and Quantitative analysis	100 ppb, 98.3% accuracy
Li *et al.*[65]	Silver nanoparticle	ANN, PLS	Ganciclovir, penciclovir, valacyclovir-hydrochloride	Quantitative analysis	1.0×10^{-6} mol L^{-1}
Uysal *et al.*[66]	/	PCA, PCR, PLS, ANNs	Butter with margarine	Quantitative analysis	R^2 of 0.968, 0.987 and 0.978
Weng *et al.*[67]	Gold nanorods	CNN, FCN, PCANet	Methyl-pyrimidine	Analyte discrimination and Quantitative	R^2 of 0.9997

				analysis	
Yan *et al.*[68]	Gold nanoparticles	XGBR	Escherichia coli	Quantitative analysis	four orders of magnitude lower than that of visual limits
Villa *et al.*[69]	Au coated printing paper	MCR-ALS	Uric acid	Quantitative analysis	0.11 mmol L^{-1}, R^2 of 0.989
Alstrom *et al.*[70]	Silicon nanopillars	NMF	17β-Estradiol	Analyte discrimination	30 SNR
Luo *et al.*[71]	/	Vis-CAD	Polycyclic aromatic hydrocarbons	Analyte discrimination	99% identification accuracy
Yang *et al.*[72]	Silver nanoparticle	RamanNet	Endotoxin	Analyte discrimination	100% identification accuracy
Banaei *et al.*[73]	gold nanoparticles	CT	Extracellular vesicles	Analyte discrimination	95% sensitivity and 96% specificity
Cha *et al.*[74]	Quantum dot	barcode-based ML	TentaGel etc.	Analyte discrimination	100% identification accuracy
Fang *et al.*[75]	Silver film	Residual network	Tumor cells	Analyte discrimination	100% identification accuracy
Erzina *et al.*[38]	Gold nanoparticles	CNN	Tumor cells	Analyte discrimination	100% identification accuracy
Tang *et al.*[76]	Silver nanoparticle	CNN	Staphylococcus	Analyte discrimination	ACC 98.21%, AUC 99.93%
Thrift *et al.*[77]	Au nanosphere	SVM, CNN greater than 90%	Pathogen	Analyte discrimination	Greater than 90% identification accuracy
Barucci *et al.*[78]	Silver Nanowires	PCA/KM, t-SNE/KM	Proteins	Analyte discrimination	Greater than 90% identification accuracy
Thrift *et al.*[79]	Gold nanosphere	Variational autoencoder	Bacteria	Analyte discrimination and Quantitative analysis	0.1μg/ Ml, 99% identification accuracy
Rahman *et al.*[80]	Bacterial cellulose nanocrystals	SVM	Bacteria	Analyte discrimination	87.7% identification accuracy
Nguyen *et al.*[81]	Silver nanorods	Reservoir computing model	DNA	Analyte discrimination	99.5% identification accuracy
Shi *et al.*[82]	Silver nanoparticle	DNN	DNA	Analyte discrimination	90.28% identification accuracy
Shin *et al.*[83]	Gold nanoparticle	PCA	Exosomes	Analyte discrimination	95% identification accuracy

Karunakaran *et al.*[84]	Gold nanoparticles	SVM	Cervical squamous cell carcinoma	Analyte discrimination	94% identification accuracy
Kazemzadeh *et al.*[85]	Gold nanoparticles	PCA	Extracellular vesicles	Analyte discrimination	100 times more sensitive than ELISA
Koster *et al.*[86]	Gold nanoparticles	PCA-QDA	Extracellular vesicles	Analyte discrimination	98.3% identification accuracy
Park *et al.*[87]	Gold nanoparticles	PCA	Exosome	Analyte discrimination	95.3% sensitivity and 97.3% specificity
Lim *et al.*[88]	Gold nanoparticles	PCA	Influenza viruses	Analyte discrimination	95% identification accuracy
Ferreira *et al.*[89]	Silver nanoparticle	PCA	Breast cancer exosomes	Analyte discrimination	10^{-11} M, 95% identification accuracy

Reference

1. R. Rojas and F. Claro, *The Journal of Chemical Physics*, 1993, **98**, 998-1006.
2. J. F. Li, Y. J. Zhang, S. Y. Ding, R. Panneerselvam and Z. Q. Tian, *Chem. Rev.*, 2017, **117**, 5002-5069.
3. H. L. Wang, E. M. You, R. Panneerselvam, S. Y. Ding and Z. Q. Tian, *Light-Science & Applications*, 2021, **10**, 161.
4. S. Katoch, S. S. Chauhan and V. Kumar, *Multimed Tools Appl*, 2021, **80**, 8091-8126.
5. A. Hiassat, A. Diabat and I. Rahwan, *J. Manuf. Syst.*, 2017, **42**, 93-103.
6. D. Li, H. Zhou, X. Hui, X. He and X. Mu, *Anal. Chem.*, 2021, **93**, 9437-9444.
7. F. Wen, Z. X. Zhang, T. Y. He and C. K. Lee, *Nat. Commun.*, 2021, **12**, 5378.
8. S. J. Wetzel, *Phys Rev E*, 2017, **96**, 022140.
9. Y. Aït-Sahalia and D. Xiu, *J. Am. Stat. Assoc.*, 2018, **114**, 287-303.
10. J. Zhu, Z. Ren and C. Lee, *ACS Nano*, 2021, **15**, 894-903.
11. H. Abdi and L. J. Williams, *Wiley Interdiscip. Rev. Comput. Stat.*, 2010, **2**, 433-459.
12. H. Zhou, D. Li, X. He, X. Hui, H. Guo, C. Hu, X. Mu and Z. L. Wang, *Adv. Sci.*, 2021, **8**, e2101020.
13. B. Richhariya and M. Tanveer, *Expert Syst. Appl.*, 2018, **106**, 169-182.
14. A. Zendehboudi, M. A. Baseer and R. Saidur, *J. Cleaner Prod.*, 2018, **199**, 272-285.
15. J. Yan, F. Shi, M. Zhao, Z. Wang, Y. Yang and S. Chen, *IEEE Sens. J.*, 2019, **19**, 9624-9633.
16. Y. X. Leong, Y. H. Lee, C. S. L. Koh, G. C. Phan-Quang, X. Han, I. Y. Phang and X. Y. Ling, *Nano Lett.*, 2021, **21**, 2642-2649.
17. S. Kang, I. Kim and P. J. Vikesland, *Anal. Chem.*, 2021, **93**, 9319-9328.
18. Y. Hong, Y. Li, L. Huang, W. He, S. Wang, C. Wang, G. Zhou, Y. Chen, X. Zhou, Y. Huang, W. Huang, T. Gong and Z. Zhou, *J. Biophotonics*, 2020, **13**, e201960176.
19. F. Zhu, J. Gao, J. Yang and N. Ye, *Pattern Recogn.*, 2022, **123**.
20. P. Buzzini, J. Curran and C. Polston, *Forensic Chem.*, 2021, **24**.
21. I. Boginskaya, R. Safiullin, V. Tikhomirova, O. Kryukova, N. Nechaeva, N. Bulaeva, E. Golukhova, I. Ryzhikov, O. Kost, K. Afanasev and I. Kurochkin, *Biomedicines*, 2022, **10**.
22. X. Xu, L. Zhao, Q. Xue, J. Fan, Q. Hu, C. Tang, H. Shi, B. Hu and J. Tian, *Anal. Chem.*, 2019, **91**, 7973-7979.
23. Q. Li, W. Li, J. Zhang and Z. Xu, *Analyst*, 2018, **143**, 2807-2811.
24. P. Reokrungruang, I. Chatnuntawech, T. Dharakul and S. Bamrungsap, *Sens. Actuators, B*, 2019, **285**, 462-469.
25. S. Weng, W. Zhu, R. Dong, L. Zheng and F. Wang, *Sensors (Basel)*, 2019, **19**.
26. J. L. Speiser, M. E. Miller, J. Tooze and E. Ip, *Expert Syst Appl*, 2019, **134**, 93-101.
27. S. Seifert, *Sci Rep*, 2020, **10**, 5436.

28. T. Moisoiu, S. D. Iancu, D. Burghelea, M. P. Dragomir, G. Iacob, A. Stefancu, R. G. Cozan, O. Antal, Z. Balint, V. Muntean, R. I. Badea, E. Licarete, N. Leopold and F. I. Elec, *Biomedicines*, 2022, **10**.
29. W. Hu, S. Ye, Y. Zhang, T. Li, G. Zhang, Y. Luo, S. Mukamel and J. Jiang, *J. Phys. Chem. Lett.*, 2019, **10**, 6026-6031.
30. F. Lussier, V. Thibault, B. Charron, G. Q. Wallace and J.-F. Masson, *TrAC, Trends Anal. Chem.*, 2020, **124**.
31. L. C. Chu, S. Park, S. Kawamoto, A. L. Yuille, R. H. Hruban and E. K. Fishman, *J. Comput. Assist. Tomogr.*, 2021, **45**, 343-351.
32. K. E. Schackart, III and J.-Y. Yoon, *Sensors*, 2021, **21**.
33. A. John-Herpin, D. Kavungal, L. von Mucke and H. Altug, *Adv. Mater.*, 2021, **33**, e2006054.
34. R. Yan, T. Wang, X. Jiang, X. Huang, L. Wang, X. Yue, H. Wang and Y. Wang, *Nanotechnology*, 2021, **32**.
35. E. Vahidzadeh and K. Shankar, *Nanomaterials*, 2021, **11**.
36. J. Gao, Q. Jiang, B. Zhou and D. Chen, *Math. Biosci. Eng.*, 2019, **16**, 6536-6561.
37. S. M. Anwar, M. Majid, A. Qayyum, M. Awais, M. Alnowami and M. K. Khan, *J. Med. Syst.*, 2018, **42**, 226.
38. M. Erzina, A. Trelin, O. Guselnikova, B. Dvorankova, K. Strnadova, A. Perminova, P. Ulbrich, D. Mares, V. Jerabek, R. Elashnikov, V. Svorcik and O. Lyutakov, *Sensors and Actuators B-Chemical*, 2020, **308**.
39. Q. Fu, Y. Zhang, P. Wang, J. Pi, X. Qiu, Z. Guo, Y. Huang, Y. Zhao, S. Li and J. Xu, *Anal. Bioanal. Chem.*, 2021, **413**, 7401-7410.
40. J. Zhu, A. S. Sharma, J. Xu, Y. Xu, T. Jiao, Q. Ouyang, H. Li and Q. Chen, *Spectrochimica Acta Part a-Molecular and Biomolecular Spectroscopy*, 2021, **246**, 118994.
41. Z. Ren, Z. Zhang, J. Wei, B. Dong and C. Lee, *Nat. Commun.*, 2022, **13**, 3859.
42. L. Kuhner, R. Semenyshyn, M. Hentschel, F. Neubrech, C. Tarin and H. Giessen, *ACS Sens*, 2019, **4**, 1973-1979.
43. J. Meng, J. J. Cadusch and K. B. Crozier, *ACS Photonics*, 2021, **8**, 648-657.
44. C. C. Nadell, B. Huang, J. M. Malof and W. J. Padilla, *Opt. Express*, 2019, **27**, 27523-27535.
45. S. Jafar-Zanjani, S. Inampudi and H. Mosallaei, *Sci. Rep.*, 2018, **8**, 11040.
46. A. D. Phan, C. V. Nguyen, P. T. Linh, T. V. Huynh, V. D. Lam, A.-T. Le and K. Wakabayashi, *Crystals*, 2020, **10**.
47. S. V. Kalinin, K. M. Roccapriore, S. H. Cho, D. J. Milliron, R. Vasudevan, M. Ziatdinov and J. A. Hachtel, *Adv. Opt. Mater.*, 2021, DOI: 10.1002/adom.202001808.
48. E. Corcione, D. Pfezer, M. Hentschel, H. Giessen and C. Tarin, *Sensors*, 2022, **22**.

49. J. Meng, L. Weston, S. Balendhran, D. Wen, J. J. Cadusch, R. Rajasekharan Unnithan and K. B. Crozier, *Laser Photonics Rev.*, 2022, **16**.
50. J. Kyoung, H. E. Kang and S. W. Hwang, *ACS Photonics*, 2017, **4**, 783-789.
51. William Cheung, Iqbal T. Shadi, Yun Xu and a. R. Goodacre, *J Phys Chem C*, 2010.
52. Y. X. Wu, P. Liang, Q. M. Dong, Y. Bai, Z. Yu, J. Huang, Y. Zhong, Y. C. Dai, D. Ni, H. B. Shu and C. U. Pittman, Jr., *Food Chem*, 2017, **237**, 974-980.
53. Y. J. Ai, P. Liang, Y. X. Wu, Q. M. Dong, J. B. Li, Y. Bai, B. J. Xu, Z. Yu and D. Ni, *Food Chem*, 2018, **241**, 427-433.
54. S. Weng, S. Yu, R. Dong, J. Zhao and D. Liang, *Molecules*, 2019, **24**.
55. S. Weng, M. Qiu, R. Dong, F. Wang, L. Huang, D. Zhang and J. Zhao, *Spectrochim Acta A Mol Biomol Spectrosc*, 2018, **200**, 20-25.
56. B. Frank, P. Kahl, D. Podbiel, G. Spektor, M. Orenstein, L. Fu, T. Weiss, M. Horn-von Hoegen, T. J. Davis, F.-J. M. zu Heringdorf and H. Giessen, *Sci. Adv.*, 2017, **3**, e1700721.
57. H. Dies, J. Raveendran, C. Escobedo and A. Docoslis, *Sensors and Actuators B: Chemical*, 2018, **257**, 382-388.
58. R. Salemmilani, B. D. Piorek, R. Y. Mirsafavi, A. W. Fountain, 3rd, M. Moskovits and C. D. Meinhart, *Anal Chem*, 2018, **90**, 7930-7936.
59. R. Beeram, D. Banerjee, L. M. Narlagiri and V. R. Soma, *Anal. Methods*, 2022, DOI: 10.1039/d2ay00408a.
60. W. J. Thrift and R. Ragan, *Anal. Chem.*, 2019, **91**, 13337-13342.
61. Q. W. Bao, H. Zhao, S. Han, C. Zhang and W. Hasi, *Analytical Methods*, 2020, **12**, 3025-3031.
62. J. L. Li, D. W. Sun, H. B. Pu and D. S. Jayas, *Food Chemistry*, 2017, **218**, 543-552.
63. W. Cheung, I. T. Shadi, Y. Xu and R. Goodacre, *J Phys Chem C*, 2010, **114**, 7285-7290.
64. H. Dies, J. Raveendran, C. Escobedo and A. Docoslis, *Sensor Actuat B-Chem*, 2018, **257**, 382-388.
65. D. G. Li, Q. Y. Zhang, B. G. Deng, Y. Chen and L. M. Ye, *Appl Surf Sci*, 2021, **539**.
66. R. S. Uysal, I. H. Boyaci, H. E. Genis and U. Tamer, *Food Chemistry*, 2013, **141**, 4397-4403.
67. S. Z. Weng, H. C. Yuan, X. Y. Zhang, P. Li, L. Zheng, J. L. Zhao and L. S. Huang, *Analyst*, 2020, **145**, 4827-4835.
68. S. S. Yan, C. Liu, S. Q. Fang, J. F. Ma, J. X. Qiu, D. P. Xu, L. Li, J. P. Yu, D. X. Li and Q. Liu, *Analytical and Bioanalytical Chemistry*, 2020, **412**, 7881-7890.
69. J. E. L. Villa and R. J. Poppi, *Analyst*, 2016, **141**, 1966-1972.
70. *2014 IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, 2014.
71. S. H. Luo, W. L. Wang, Z. F. Zhou, Y. Xie, B. Ren, G. K. Liu and Z. Q. Tian, *Analytical Chemistry*, 2022, **94**, 10151-10158.

72. Y. J. Yang, B. B. Xu, J. Haverstick, N. Ibtehaz, A. Muszynski, X. Y. Chen, M. E. H. Chowdhury, S. M. Zughaier and Y. P. Zhao, *Nanoscale*, DOI: 10.1039/d2nr01277d.
73. N. Banaei, J. Moshfegh and B. Kim, *J. Raman Spectrosc.*, 2021, **52**, 1810-1819.
74. M. G. Cha, W. K. Son, Y.-S. Choi, H.-M. Kim, E. Hahm, B.-H. Jun and D. H. Jeong, *Appl. Surf. Sci.*, 2021, **558**.
75. X. Fang, Q. Zeng, X. Yan, Z. Zhao, N. Chen, Q. Deng, M. Zhu, Y. Zhang and S. Li, *J. Appl. Phys.*, 2021, **129**.
76. J.-W. Tang, Q.-H. Liu, X.-C. Yin, Y.-C. Pan, P.-B. Wen, X. Liu, X.-X. Kang, B. Gu, Z.-B. Zhu and L. Wang, *Front. Microbiol.*, 2021, **12**, 696921.
77. W. J. Thrift, A. Cabuslay, A. B. Laird, S. Ranjbar, A. I. Hochbaum and R. Ragan, *ACS Sens.*, 2019, **4**, 2311-2319.
78. A. Barucci, C. D'Andrea, E. Farnesi, M. Banchelli, C. Amicucci, M. de Angelis, B. Hwang and P. Matteini, *Analyst*, 2021, **146**, 674-682.
79. W. J. Thrift, S. Ronaghi, M. Samad, H. Wei, D. G. Nguyen, A. S. Cabuslay, C. E. Groome, P. J. Santiago, P. Baldi, A. I. Hochbaum and R. Ragan, *ACS Nano*, 2020, **14**, 15336-15348.
80. A. Rahman, S. Kang, W. Wang, Q. S. Huang, I. Kim and P. J. Vikesland, *ACS Appl. Nano Mater.*, 2022, **5**, 259-268.
81. P. H. L. Nguyen, S. Rubin, P. Sarangi, P. Pal and Y. Fainman, *Appl. Phys. Lett.*, 2022, **120**.
82. H. Shi, H. Wang, X. Meng, R. Chen, Y. Zhang, Y. Su and Y. He, *Anal. Chem.*, 2018, **90**, 14216-14221.
83. H. Shin, S. Oh, S. Hong, M. Kang, D. Kang, Y.-g. Ji, B. H. Choi, K.-W. Kang, H. Jeong, Y. Park, S. Hong, H. K. Kim and Y. Choi, *ACS Nano*, 2020, **14**, 5435-5444.
84. V. Karunakaran, V. N. Saritha, M. M. Joseph, J. B. Nair, G. Saranya, K. G. Raghu, K. Sujathan, K. S. Kumar and K. K. Maiti, *Nanomedicine-Nanotechnology Biology and Medicine*, 2020, **29**, 102276.
85. M. Kazemzadeh, C. L. Hisey, A. Artuyants, C. Blenkiron, L. W. Chamley, K. Zargar-Shoshtari, W. L. Xu and N. G. R. Broderick, *Biomedical Optics Express*, 2021, **12**, 3965-3981.
86. H. J. Koster, T. Rojalin, A. Powell, D. Pham, R. R. Mizenko, A. C. Birkeland and R. P. Carney, *Nanoscale*, 2021, **13**, 14760-14776.
87. J. Park, M. Hwang, B. Choi, H. Jeong, J. H. Jung, H. K. Kim, S. Hong, J. H. Park and Y. Choi, *Anal Chem*, 2017, **89**, 6695-6701.
88. J. Y. Lim, J. S. Nam, H. Shin, J. Park, H. I. Song, M. Kang, K. I. Lim and Y. Choi, *Anal Chem*, 2019, **91**, 5677-5684.
89. N. Ferreira, A. Marques, H. Aguas, H. Bandarenka, R. Martins, C. Bodo, B. Costa-Silva and E. Fortunato, *ACS Sens*, 2019, **4**, 2073-2083.