RamanNet: A generalized neural network architecture for Raman Spectrum Analysis

Nabil Ibtehaz[1], Muhammad E. H. Chowdhury[2,*], Amith Khandakar[2], Susu M. Zughaier[3], Serkan Kiranyaz[2], and M. Sohel Rahman[4]

[1]Department of Computer Science, Perdue University, West Lafayette, IN 47907, United States, nibtehaz@purdue.edu (NI)
[2]Department of Electrical Engineering, Qatar University, Doha-2713, Qatar; mchowdhury@qu.edu.qa (MEHC), amitk@qu.edu.qa (AK), mkiranyaz@qu.edu.qa (SK)
[3]Department of Basic Medical Sciences, College of Medicine, QU Health, Qatar University, Doha, Qatar, PO Box 2713, szughaier@qu.edu.qa (SMZ)
[4]Department of Computer Science and Engineering, Bangladesh University of Engineering and Technology, Dhaka-1205, Bangladesh, msrahman@cse.ac.buet.bd (MSR)
*Corresponding author

Abstract

Raman spectroscopy provides a vibrational profile of the molecules and thus can be used to uniquely identify different kind of materials. This sort of fingerprinting molecules has thus led to widespread application of Raman spectrum in various fields like medical dignostics, forensics, mineralogy, bacteriology and virology etc. Despite the recent rise in Raman spectra data volume, there has not been any significant effort in developing generalized machine learning methods for Raman spectra analysis. We examine, experiment and evaluate existing methods and conjecture that neither current sequential models nor traditional machine learning models are satisfactorily sufficient to analyze Raman spectra. Both has their perks and pitfalls, therefore we attempt to mix the best of both worlds and propose a novel network architecture RamanNet. RamanNet is immune to invariance property in CNN and at the same time better than traditional machine learning models for the inclusion of sparse connectivity. Our experiments on 4 public datasets demonstrate superior performance over the much complex state-of-the-art methods and thus RamanNet has the potential to become the defacto standard in Raman spectra data analysis.

1 Introduction

Raman scattering is one the various light-matter interaction, comprising absorption and subsequent emission of light by matter [1]. Spectroscopy being the study of interaction of light or broader electromagnetic radiation with matter, thus has also focused on Raman scattering, ever since it's discovery in 1928 by Raman and Krishnan [2]. Unlike elastic scatering, e.g. Raleigh scattering, the wavelength of incident light changes in Raman scattering [3]. Following the typical norm of inelastic light scattering, when a photon excites the sample the electrons are raised to a higher virtual energy state [4]. This excitation event is usually short-lived and the molecule soon reaches

a new stable energy state, either lower (Stokes shift) or higher (anti-Stokes) [5]. Based on the difference in energy, the sample achieves a different vibrational and rotational state. Therefore, Raman spectroscopy can be used to analyze the vibrational modes of various molecules, extracting the structural fingerprint of such materials in the process [1].

Being capable of uniquely fingerprinting materials, Raman spectra has been used in a wide variety of applications, covering medical diagnoistics, forensics, mineralogy, bacteriology and virology [6]. **ADDING CITATIONS**

Conventionally Raman spectra are analyzed in terms of wavenumber $\tilde{v}$ (cm^{-1}). The standard practice is to present them with the wavenumber shift linearly increasing along the horizontal axis. On the contrary, the vertical axis ordinate is proportional to intensity [5]. Therefore, this setting is not apparently too different from traditional spectograms. However, the issue in treating Raman Spectra as typical spectrograms is that for Raman spectrum we don't have time along the horizontal axis, thus we cannot apply models that are capable of spectrum analysis, e.g., CNN. On the contrary, although the traditional machine learning methods are more capable, they suffer from curse of dimensionality as Raman spectra data is usually long. Principal Component Analysis (PCA) has been widely used for feature reduction, as 34 out of recent 52 papers used that [6], still it may not always be able to compress Raman spectra as PCA is more suitable for tabular data source, where the features are ideally correlated, but in Raman spectra the intensities at nearby Raman shifts are expected to demonstrate correlation.

With the reduction of cost and complexity related to Raman data extraction pipeline, there has been an unprecedented expansion in Raman datasets in recent years. These sudden growth of available Raman data requires suitable methods to analyze them properly. However, to the best of our knowledge, till date there has not been any machine learning methodology devised solely focused on Raman spectra data, considering the pattern and properties of this unique data source. To this end, we carefully contemplate the properties of Raman spectra data and corresponding attributes of contemporary machine learning methods. We argue why application of convolutional neural networks (CNN) may not be appropriate for invariance properties, but at the same time acknowledge that CNN is more capable than traditional machine learning methods for the sparse connection and weight sharing properties. We make an attempt to fuse the best of both worlds and propose RamanNet, a deep learning model that also follows the paradigm of sparse connection without the limitation of temporal or spatial invariance. We achieve this by using shifted densely connected layers and emulate sparse connectivity found in CNN, without the concern of invariance. Furthermore, dimensionality reduction has been involved in most Raman spectra applications, thus we employ Triplet loss in our hidden layers and make more separable and informative embeddings in lower dimensional space. Our experiments on 4 public datasets demonstrate superior performance over the much complex state-of-the-art methods and thus RamanNet has the potential to become the defacto standard in Raman spectra data analysis.

2 Motivations and High Level Considerations

From a visual perception, Raman spectrum resembles a signal-like waveform. Which has led to the application of 1D Convolutional Neural Networks to analyze Raman spectrums [7, 8, 9]. However, Raman spectrum is not a typical signal rather it's an energy distribution plot, which may prevent appropriate utilization of CNNs. In this section, we briefly discuss the properties of CNNs and present our motivations and rationale.

Convolutional Neural Networks are one of the most successful and widely used neural network architectures, particularly in computer vision [10] and signal processing [11] domains. The success of CNNs can be largely attributed to three primary properties of CNNs, namely sparse interaction, parameter sharing and equivariant representations [12]. The nature of image or signal data, i.e., spatial or temporal properties, works in perfect harmony with the properties of CNN. Unlike traditional feed forward neural networks, where all the input features are connected with all the hidden or output layers, CNNs employ sparse connectivity by leveraging kernels. Only a small portion of input is analyzed at a specific step using kernels, thus this prevents the computation from being overwhelmed with analyzing the entire input at once. Although focus is put on local information, global information is also considered by appropriate use of pooling operations along with broader field of vision in deeper networks. Furthermore, the use of kernel based computation also faciliatates parameter sharing, reducing number of paramaters and risk of overfitting simultaneously. Another vital property of CNN is the equivariance in representation. Mathematically for a convolutional operation f on an image I, $f(I(x, y)) = f((I(x - h, y - k))$, i.e., a point of interest whether it resides at location (x, y) or at a shifted location $(x - h, y - k)$, the output of the convoluation operation remains the same.

This equivarience to translation plays a pivotal role in working image or signal data. When processing signal data, this property implies that CNN generates a timeline of the emergence of different keypoints in the signal. As a result, regardless of the time of occurence, all the features in a signal is captured, unlike traditional neural networks which would have only sought for the features at the exact fixed timestamps. Furthermore, the application of global pooling operations makes sure that all the feature signatures are preserved in the final representation.

In Fig. 1, we have presented a simplified example of how CNN works with a signal as input. In Fig. 1a, we have three shifted ECG signals, since we cannot always be certain of the occurrence of a particular feature when dealing with signals, it is imperative that the model can identify the feature irrespective of the timestamp. In Fig. 1b, we can observe the featuremaps from a convoluational layer, it is apparent that the equivalent features are detected, albeit shifted in accordance with the shifted nature the signals, complying to the property $f(I(x, y)) = f(I(x - h, y - k))$. Finally, a global pooling operation summarizes the featuremaps and provides identical representations of all the three signals in Fig. 1c.
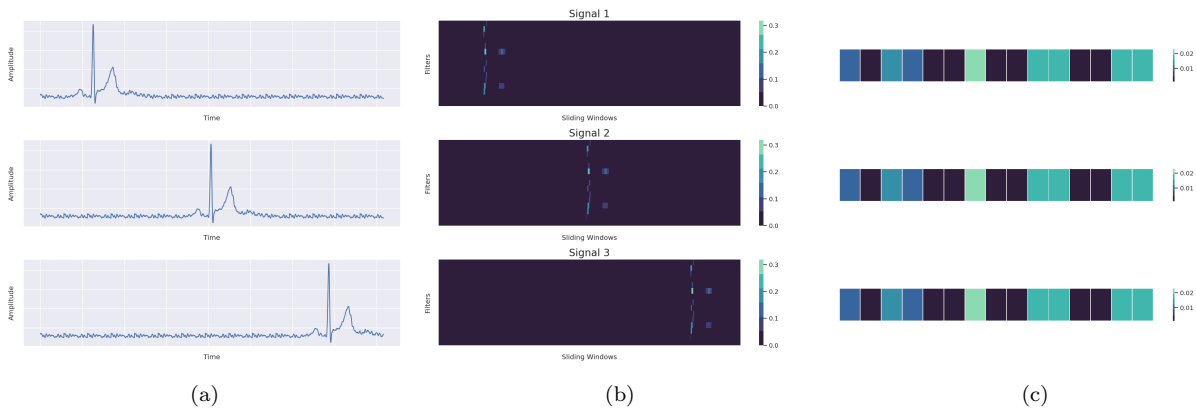

Figure 1: Analysis of CNN for a signal input.

From the example, it is obvious how applicable CNNs are for signal data. Since the pattern

of Raman spectrum resembles signal for quite a bit, it is trivial to use CNNs to analyze raman spectra. The sparse connectivity and parameter sharing truely helps in this regard, as it makes the computation simpler and less prone to overfitting. However, the equivariance to translation which proved vital for analyzing signals acts troublesome when working with Raman spectrum. The Raman spectrum is plotted as intensity vs Raman shifts. Thus the concept of equivariant translation is not applicable in this case, because similar intensity at different Raman shifts implies a completely different meaning. For example, in Fig. 2, we present a similar scenario but with Raman spectrum as input. It can be seen that, we have three completely different Raman spectrums, having similar patterns in different Raman shifts. But the CNN model, due to translational equivariance, treats them as shifted inputs and generates the same output for all of them (Fig. 2c).

Therefore, we hypothesize that the use of CNN in Raman spectrum analysis is inappropriate. This is has been addressed in prior works using CNNs by using deeper networks (broadening the field of vision thereby) along with discarding pooling layers (attempting to preserve the locations of spectral peaks) [8]. On the contrary we can use traditional neural networks or machine learning models which is capable of keeping track of the exact locations. However, they succumb to the curse of dimensionality [13], due to the long length of Raman spectrum. Although PCA has been mostly used to reduce that [6], still it feels less intuitive to model spectrum inputs using hyperplane optimization, which is more suitable for tabular data. As the spectra are apparently correlated in neighboring Raman shifts, which motivates the use of sparse connectivity through kernel-like operations in CNN instead. This conjecture is also supported by the reduced accuracy in classical models [7, 8].
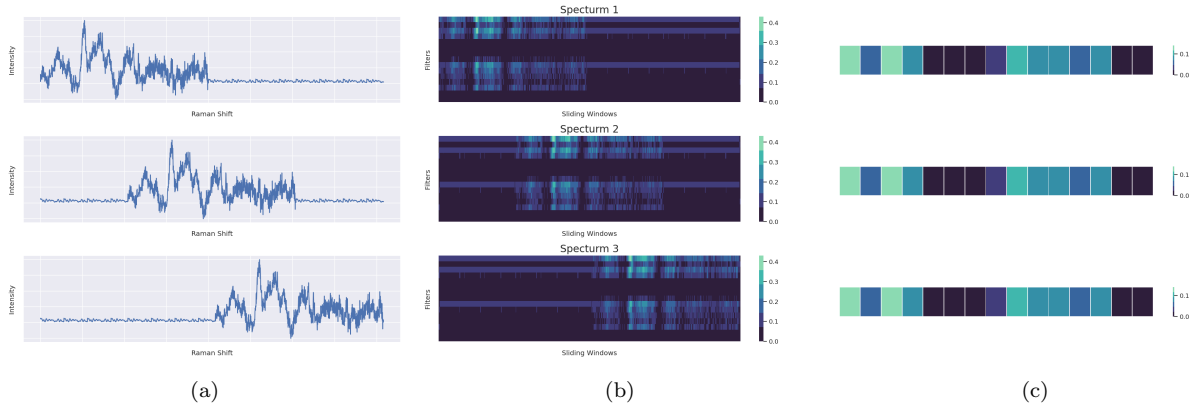

Figure 2: Analysis of CNN for a Raman spectrum input.

To sum up, we are faced with a dilemma between CNN and classical machine learning pipelines. On one hand, CNNs help in analyzing Raman spectra for the benefits of sparse connectivity (considering the correlation in neighboring Raman shifts) and parameter sharing (reducing the number of parameters and overfitting), but loses its applicability due to translational equivariance. On the other hand, traditional models are free from translational equivariance, but they can't comprehend the correlation in the neighboring raman shifts and suffers from the curse of dimensionality as trying to optimize too many parameters at once.

Therefore, we propose a middle ground between the two approaches, fusing the best of the both worlds. We propose to use shifted multi-layer perceptrons to analyze shifted windows of Raman spectra. This facilitates sparse connectivity as the shifted MLP layers mimic kernels and only

4

analyze a part of the input. Moreover, parameter sharing is redundant in this regard, as the kernel-like operations of a particular kernel are only performed in one part and not elsewhere. Finally, this also eliminates the issues with translation equivarience, as for differnt locations or different shifted windows we have different MLP layers. Thus,

Mathematically, a typical 1D CNN operation can be simplified as,

$$y(i) = \sigma(\sum_h x(i+h)k(h) + b) \tag{1}$$

Here, x is the 1D input, y is the output, k is a learned kernel, b is the bias term and σ is a nonlinear operation. The same kernel k is applied everywhere thus the translational equivarience is achieved.

On the contrary, our proposed modification from using a MLP,

$$y(i) = \sigma(W_{f(i)}^T x + b) \equiv \sigma(\sum_h x(i+h)k_{f(i)}(h) + b) \tag{2}$$

Here, the dot product $W^T x$ is mathematically equivalent to an 1D convolutional operation with proper relation between the weight matrix W and kernel k. In addition since we are using sliding windows, the weight matrix $W_{f(i)}$ and kernel $k_{f(i)}$ depends on the location, i.e., value of i.

3   Proposed Architecture

Hence, following our proposed modifications, we present RamanNet. As mentioned in the previous section, we mimic the convolution operation using multi-layer perceptrons or so called densely connected neural network layers. The input Raman spectra is broken into overlaping sliding windows of length w and step size dw, and each of them is passed to a different dense block with n_1 neurons. This ensures sparse connectivity and reduces risk of overfitting. Furthermore, this configuration somewhat resembles a convolution operation without translation, thus we can consider the features extracted at the neurons the same way as we would consider features learnt from kernels.

The features from all the dense blocks are concatenated together and a dropout dp_1 is applied. The concatenated features are again summarized using another dense layer with n_2 neurons. The outputs from the summarization is regularized with a dropout of dp_2.

Finally, we compute n_f features from the regulized outputs using another dense layer with n_f neurons. We name this layer Embedding Layer. Raman spectra are known to be noisy with low SNR [8], which often leads to less separation between the classes. Therefore, in order to improve this we introduce Triplet Loss [14] as an auxiliary loss. Triplet Loss is defined using eucleidian distance, f, between an anchor A, positive example P and negative example N as,
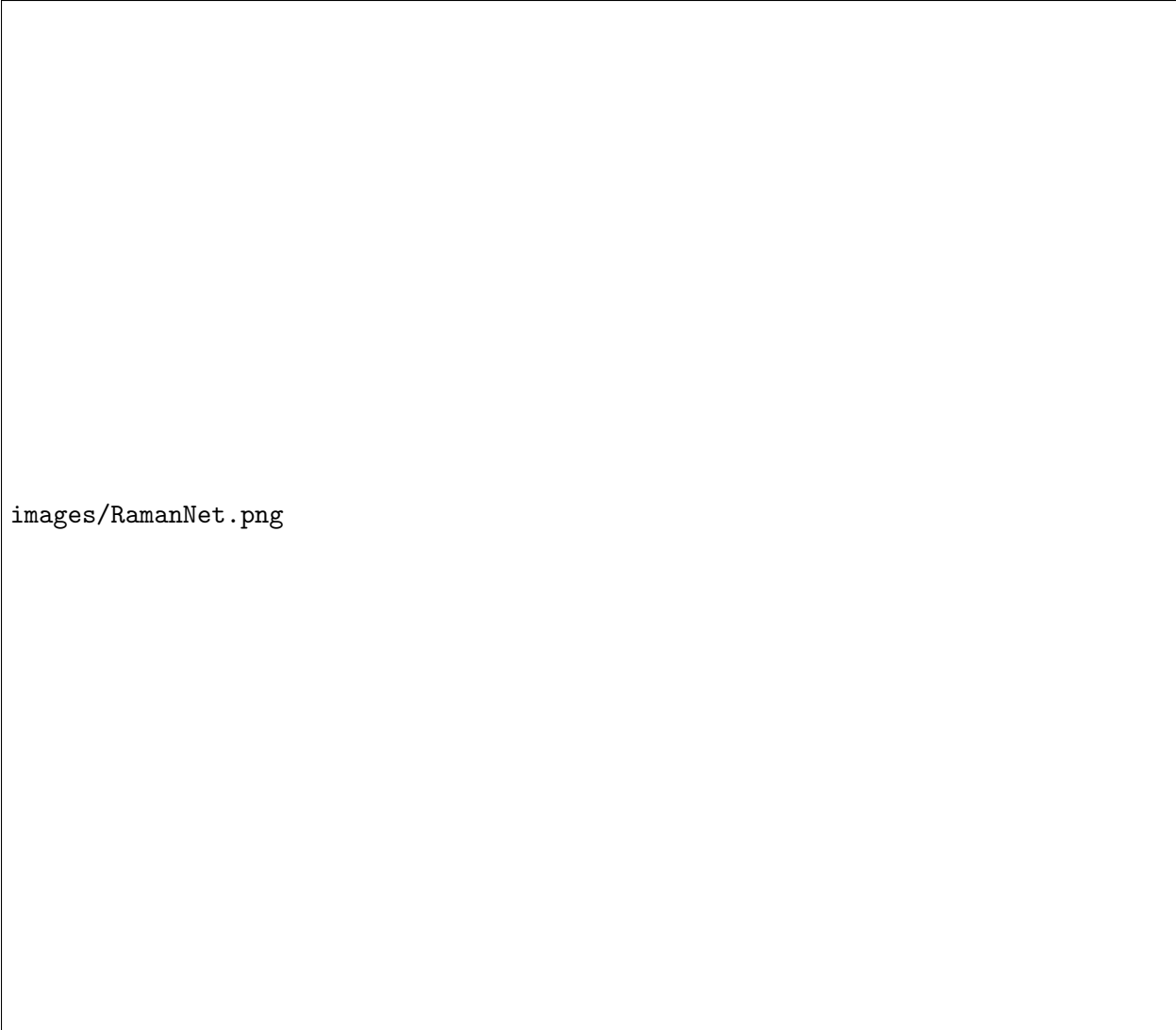
$$L(A, P, N) = max(||f(A) - f(P)||^2 - ||f(A) - f(N)||^2 + \alpha, 0) \tag{3}$$

Here, α is a margin term to ensure that the model learns non-trivial information.

Using triplet loss we obtain a well separated embedding space. The embeddings are finally used to predict the classes of input using Softmax activation function and a dropout of dp_3.

Other than the output layer, all the layers uses LeakyRELU activation function [15] and are batch normalized [16]. In our experiments we used the hyperparameters as $w = 50$, $dw = 25$, $n_1 = 25$, $n_2 = 512$, $n_f = 256$, $dp_1 = 0.5$, $dp_2 = 0.4$, $dp_3 = 0.25$.

A simplified diagram of the RamanNet has been presented in Fig. 3.

5

Figure 3: RamanNet Architecture

4 Datasets

One particular limitation when working with deep learning for Raman spectrum analysis is the lack of sufficient public benchmark datasets [6]. Although there has been several of recent works utilizing deep learning models for Raman spectrum analysis, the datasets are mostly proprietary or private [17, 18]. From an elaborate study of the recent works, investigating Raman Spectrum analysis, we selected 4 publicly available datasets for our analysis and benchmarking.

4.1 Covid Dataset

We used the publicly available data [19] from the recent work [20], which acts as a pilot study of primary screening of COVID-19 by Raman spectroscopy. This dataset contains a total of 177 serum samples collected from 63 COVID-19 patients, 59 suspected ones and 55 healthy people

(i.e., control group). The COVID-19 group was recruited at the Chengdu Public Health Clinical Medical Center, and it includes 58 symptomatic and 5 asymptomatic patients. The suspected group demonstrated 'flu' like symptoms but were tested negative but at least to RT-PCR tests. For all the subjects, 1 hour repose of blood sampling the serum was extracted by centrifuging at 3000 rpm for 10 minutes, and were stored at 4 ℃. Later, a single-mode diode laser with 785 nm wavelength and 100 mW power was used for Raman excitation. The laser power on the sample was measured around 70 mW and the spectra were recorded in the range of 600-1800 cm^{-1}.

4.2 Melanoma Dataset

Erzina et al. [9] used Surface Enhanced Raman Spectroscopy to detect Melanoma. Healthy residual skin and skin melanoma metastasis were collected. From cell line and primary culture 12 categories of samples were considered, each having 8-9 samples. The SERS spectra were measured using ProRaman-L spectrometer at 785 nm excitation wavelength and 33 mW power. AuMs, functionalized by ADT-NH2, ADT-COOH or ADT-$(COOH)_2$ was used to measure the SERS spectra, i.e., 3 different spectra were obtained for each sample. Finally, the background was removed using smooting algorithms and the recorded spectra were normalized to 0-1 intensity. The range of the recorded spectra is 100-4278 cm^{-1}.

4.3 Mineral Dataset

Mineral substance identification is another popular application of Raman spectroscopy. Among the various mineral databases, we selected the RRUFF database [21]. The RRUFF project aims at curating the most comprehensive set of high quality spectral data from well characterized minerals, comprising Raman spectra, X-ray diffraction and information from chemistry. The database contains Raman spectra of various minerals at different configurations, i.e. varied wavelengths and orientations, accompanied by various kinds of processing. Furthermore, the Raman spectra computed from the different materials are hardly consistent, e.g., the ranges of Raman shifts are also different. In order to alleviate such irregularities, we have only considered spectra computed at 432 nm, which is the majority. Moreover, we have only collected the raw spectra, i.e., which were not processed any way. Since the recorded spectra have different ranges of Raman shift, we have cropped the region that is common in all the spectra and used cubic spline to interpolate the locations if necessary.The resultant spectra cover the range 280 - 4237 cm^{-1} and min-max normalization was performed. Finally, we take the mineral classes with at least 10 samples, which reduces the database to twenty mineral classes and use them to evaluate the proposed model.

4.4 Bacteria Dataset

We collected the Bacteria-ID dataset from [8], where the potential of Raman spectrosocpy in label-free bacteria detection has been investigated. This dataset consists of 30 bacterial and yeast isolates, including multiple isolates of Gram-negative and Gram-positive bacteria. The dataset is organized into a reference training dataset, reference fine-tuning set and test set. The fine-tuning dataset is used to account for the changes in measurement caused by optical system efficiency degradation. The training dataset contains 2000 spectra for each of the 30 isolates, whereas the fine-tuning and test set contains 100 spectra for each isolate. The isolates were cultured on blood agar plates sealed with Parafilm and stored at 4 $°C$. The Raman spectra was generated using Horiba LabRAM HR

Evolution Raman microscope, with 633 nm illumination at 13.17 mW along with a 300 l/mm grating. The spectra was computed at 1.2 cm^{-1} dispersion to simultaneously maximize signal strength and minimize background signal. The recorded spectra was normalized to $0 - 1$ intensity, covering the spectral range between 381.98-1792.4 cm^{-1}.

5   Experimental Setup

The experiments have been conducted in a server computer with Intel Xeon @2.2GHz CPU, 24 GB RAM, and NVIDIA TESLA P100 (16 GB) GPU. We implemented the RamanNet architecture using Tensorflow [22]. The codes are available in the following github repository.

<div align="center">

`https://github.com/nibtehaz/RamanNet`

</div>

In the following subsections, we briefly explain the experimental protocols and evaluation procedures adopted for the different tasks. We have tried to mimic the corresponding baseline papers to ensure a fair comparison. As mentioned earlier, we were only compelled to follow a different evaluation scheme for the Mineral dataset, thus we reproduced that baseline model's output following our exact protocol.

5.1   Covid Dataset

We followed the identical evaluation method as presented in [20]. In this work, the authors conducted a "blind" validation. They randomly divided the whole dataset into 70% for training and 30% as hold-out test set. In order to further assess the independence of the data over model performance, this process was repeated 50 times and the average values of the metrics were recorded. We followed the same protocol to evaluate RamanNet. In order to avoid overfitting, we used 10% data from the training set as validation data, but the hold-out test data was left completely independent from the training process and was only used for evaluation. The models were trained for 1000 epochs.

5.2   Melanoma Dataset

In the original work Erzina et al. [9], performed 75%:25% train-validation split. Therefore, we attempted to follow a similar splitting criterion. Since, the splitting information was not provided, we opted to perform a 4 fold cross-validation instead, as it would also split the data in the same ratio. The authors demonstrated 100% accuracy using 3 different spectra (AuMs functionalized by ADT-NH2, ADT-COOH or ADT-$(COOH)_2$ respectively) together. However, to make the task difficult, we experimented with using only one type of spectra as input. In order to achieve that, we reproduced the model as described in [9] and evaluated the model with one particular spectra as input at a time.

5.3   Mineral Dataset

Jichao et al. [7] also used the RRUFF database for mineral classification task. However, they employed the leave one out cross validation scheme, which is computationally too expensive. In order to reduce the computational requirements, we thus opted for a 5 fold cross-validation instead.

In order to compare RamanNet with the model presented by Jichao et al. [7], we implemented their proposed model and used it in our analysis.

5.4   Bacteria Dataset

In order to assess RamanNet on the Bacteria-ID dataset, we followed the same training and evaluation procedure as presented in [8]. Similar to their approach, we first pre-trained the model using the reference training dataset, through a 5 fold cross-validation scheme. The five models obtained in this process was then fine-tuned on the fine-tuning dateset, which was spilt into 90% training and 10% validation. The model with the highest accuracy on this validation set was considered and was evaluated on the independent test dataset. The models were trained for 100 and 250 epochs respectively on the reference training and fine-tuning set.

6   Results

6.1   RamanNet Consistently Outperforms Existing Models

6.1.1   Covid Dataset

In [20], a SVM model was developed to distinguish the different categories, namely healthy, suspected and COVID-19 patients. Instead of working with the entire spectrum, wave points with significant differnces in ANOVA test were selected. Thus, a statistically sound feature reduction was performed and the reduced featureset was used as the input to the SVM model. On the contrary, RamanNet takes the entire spectrum as input and adaptively finds the significant region therein.

The average performance over 50 random trials for the different tasks are presented in Table 1, here we present the accuracy, sensitivity and specificity values. Among the 3 tasks, differentiating between suspected and healthy subjects is the most challenging one, as evident from the inferior performance of SVM (all metrics $\leq 70\%$). RamanNet, on the other hand, performed comparatively better in this task. Notably, RamanNet improved accuracy and specificity by 13% and 21% respectively. RamanNet also achieved improved sensitivity score.

On the other two tasks the SVM model performed comparatively better. However, our proposed RamanNet consistently outperforms SVM in all these two tasks as well. Most promisingly, RamanNet improves sensitivity greatly in both the two tasks. For this problem sensitivity is crucial as we need to correctly detect the Covid-19 patients. This improvement in sensitivity did not come at any cost of specificity, rather the specificity has also been improved compared to the SVM model.

6.1.2   Melanoma Dataset

As described previously, we perform a 4 fold cross-validation on the melanoma dataset. Following the original evaluation as performed by Erzina et al. [9], we consider all the three different types of spectra simultaneously as input. In addition, we perform a difficult version of the problem by taking only one type of spectra as input at a time. The results are presented in Table 2.

From the results, it is evident that although the CNN model manages to achieve perfect 100% accuracy (in 2 folds out of 4) when given 3 spectra as input, the accuracy falls when a single spectra is given as input. This drop in performance can be explained by the loss of information, when working with a single spectra. For different functionalizations of AuMs, the sample is observed

Table 1: Results on Covid Dataset

Covid vs Suspected			
Method	Accuracy	Sensitivity	Specificity
SVM	87 ± 5	89 ± 8	86 ± 9
RamanNet	93 ± 3	97 ± 4	90 ± 6
Covid vs Healthy			
Method	Accuracy	Sensitivity	Specificity
SVM	91 ± 4	89 ± 7	93 ± 6
RamanNet	95	95 ± 4	96 ± 3
Suspected vs Healthy			
Method	Accuracy	Sensitivity	Specificity
SVM	69 ± 5	70 ± 9	66 ± 9
RamanNet	82 ± 6	77 ± 15	87 ± 11

Table 2: Results on Melanoma Dataset

	$-NH_2$		$-(COOH)_2$		$-COOH$		All		#Parameters	
Fold	RamanNet	CNN	RamanNet	CNN	RamanNet	CNN	RamanNet	CNN	RamanNet	CNN
1	**100**	97.42	**100**	100	**99.35**	94.19	100	100	1.3 M	25.7M
2	**99.35**	96.13	**99.35**	98.71	**98.71**	96.12	100	98.71		
3	**100**	95.45	**100**	98.05	**99.35**	87.01	100	100		
4	**100**	97.40	**100**	98.70	**96.75**	96.10	100	99.35		

from a different point of view and different information is obtained. Thus, when working with reduced number of spectra, insightful information are likely get lost and that negatively affects the performance. Although for the $-(COOH)_2$ as input the accuracy stays above 98%, it falls below 97% for the other two input spectra.

RamanNet on the other hand seems to consistently outperform the CNN model for both when all the 3 spectra are considered together or separately. RamanNet not only consistently achieved 100% accuracy with all the 3 spectra as input, but also with $-NH_2$ and $-(COOH)_2$ individual spectra as input. Only for $-COOH$ spectra as input, the performance was not up to the mark, but still the performance was superior to the CNN. All these improvements become more significant when we compare the number of the parameters of the two models. The CNN model consists of 25.7 M parameters whereas RamanNet has only 1.3 M parameters ($\sim 5\%$). Thus, RamanNet is not only more accurate, it is also computationally efficient at the same time.

6.1.3 Mineral Dataset

As described in the previous sections, the mineral dataset consists of 20 mineral classes and in order to compare with the model proposed by [7], we implement their model and perform a five fold cross-validation. The results are presented in Table 3.

Here, we have presented the Top 1, Top 5 and Top 10 accuracy metrics, which are popularly used for multiclass classification problems. It is evident that RamanNet consistently outperforms the previous state of the art CNN model. For Top 1 Accuracy, the improvement is more prominent, nevertheless RamanNet performs better in regards to the other metrics as well. Another point worth

Table 3: Results on Mineral Dataset

	Top 1 Accuracy		Top 5 Accuracy		Top 10 Accuracy		#Parameters	
	RamanNet	CNN	RamanNet	CNN	RamanNet	CNN	RamanNet	CNN
Fold 1	**87.5**	81.25	**95.83**	87.5	**100**	93.75		
Fold 2	**93.75**	85.42	97.92	97.92	**100**	97.92		
Fold 3	**89.58**	79.17	**93.75**	89.58	**93.75**	91.67	1.3M	6.6M
Fold 4	**97.92**	85.42	**100**	91.67	**100**	93.75		
Fold 5	**85.12**	63.83	**93.62**	78.72	**95.74**	85.10		

mentioning is that, even in this case the CNN model is heavier comparatively (6.6 M parameters vs. 1.3 M parameters for RamanNet).

6.1.4 Bacteria Dataset

In the original work [8], the authors used a 25 layer deep residual convolutional neural network to classify the bacteria isolate raman spectrum to one of the 30 classes. The proposed model achieved an average accuracy of 82.2%, and the resulting confusion matrix is presented in Fig. 4a. The authors also experimented with common and popular classifiers like Logistic Regression (LR) and Support Vector Machine (SVM) as baselines, but those models could only reacg 75.7% and 74.9% accuracy respectively.

RamanNet on the other hand manages to achieve an average accuracy of 85.5% on this dataset, despite only having 3 hidden layers. As shown in the confusion matrix (Fig. 4b), the number of misclassifications has reduced. Although erroneous predictions still exist, most Gram-positive and Gram-negative bacteria have been misclassified as Gram-positive and Gram-negative bacteria respectively, and the errors are also mostly confined within same genus, as analyzed in [8]. Although this 3.3% improvement may seem minor, this is achieve using a much shallower network (3 layers vs 25 layers). Furthermore, the improvements become more apparent when we compare the individual classifications next to each other. As presented in Fig. 4c, RamanNet achieves either equal or better accuracy for 25 out of 30 bacteria isolate classes. For other than two isolate classes, ResNet is only better with a small margin. On the contrary, the cases where RamanNet performed better it surpassed ResNet with higher margin (e.g., MSSA 3, P. mirabilis, K. aerogenes etc.).

6.2 Model Interpretation

Interpretability has been one of the focus in deep learning research in recent years [23]. Deep learning models are competent function approximator and given sufficient amount of data, they are capable of modelling almost any complex functions. With this potential, also comes the concern of what the model is actually learning from the data. The model can learn actual significant information and perform prediction accordingly, or it can merely learn from the noises and get confused from various confounding factors instead. Therefore, it is imperative to investigate the model interpretability and examine what the model is learning from.

For convolutional neural networks, we can use various methods like saliency maps [24] or score-CAM [25] methods to study model interpretation. However, for multi-layer perceptrons, it is non-trivial to do so. The various visualization methods have been designed based on CNNs and they cannot be directly translated to MLPs.

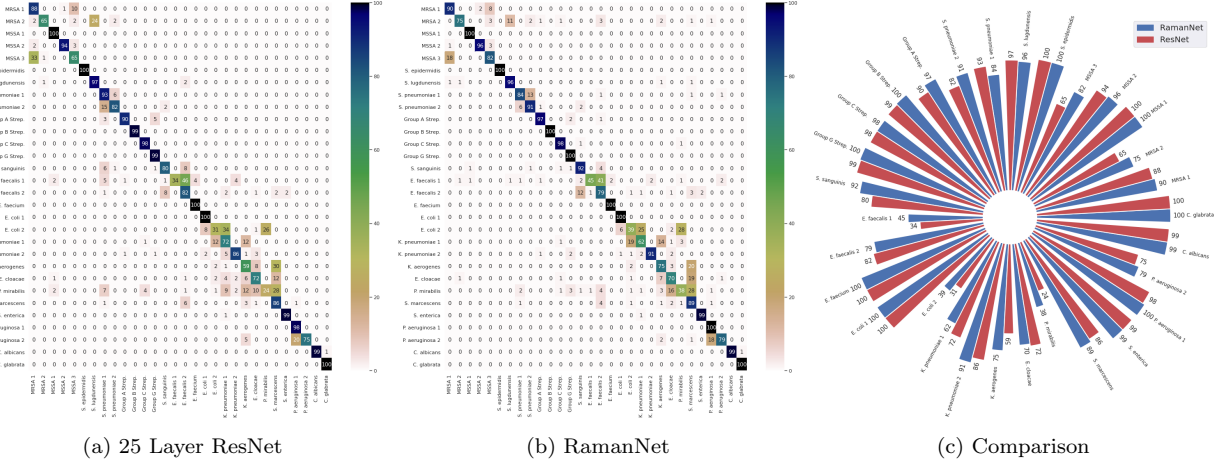(a) 25 Layer ResNet (b) RamanNet (c) Comparison

Figure 4

SHAP (SHapley Additive exPlanations) [26] is a game theoretic approach to explain the output of machine learning models. The biggest advantage of using SHAP is that it is model agnostic, thus it can be used to analyze any machine learning model. We extracted the features extracted from different moving windows of RamanNet and trained the identical top layer of RamanNet using Scikit-Learn MLP implementation, for compatibility reasons of the SHAP implementation [27]. We then computed the SHAP scores of all the -NH$_2$ samples in the melanoma dataset, the results are presented as a violin plot in Fig. 5.
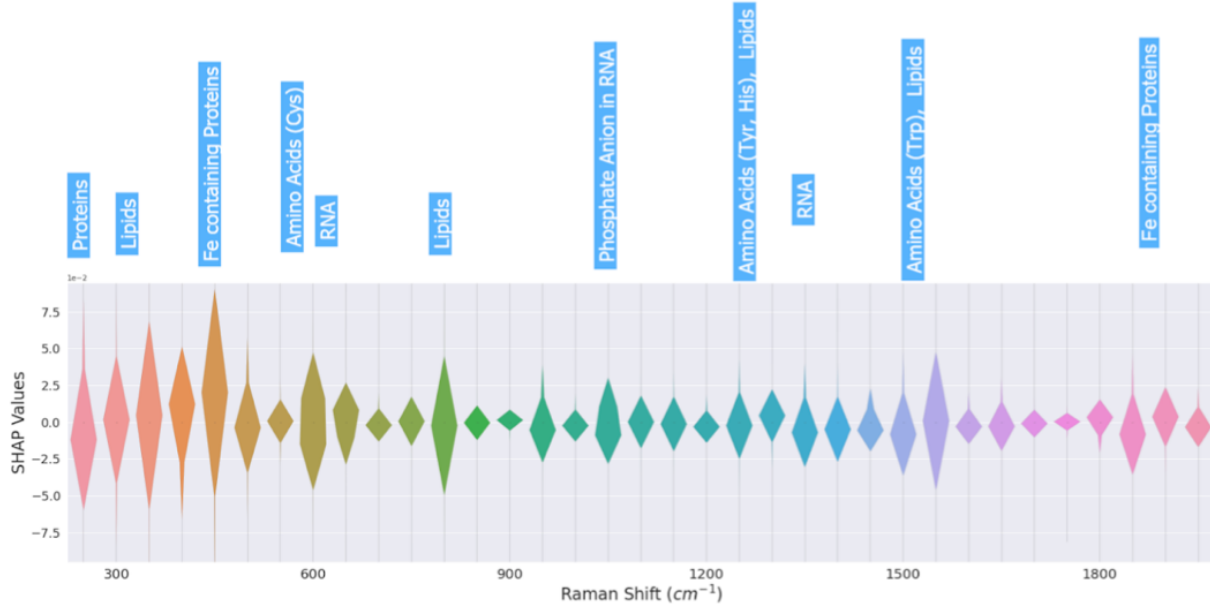

Figure 5: SHAP values for RamanNet

From the violin plot it is evident that certain regions of the spectrum contributes most to the prediction. Moreover, those regions correspond to actual significant properties of the sample. For example, the region in 400-500 cm^{-1} corresponds to Fe containing proteins and contributes most

to the prediction. After this region the lipids (300-400 cm^{-1}) contributes the most, and it goes on. The unlabelled regions in the figure correspond to less attributed regions, and it is apparent that the model was also aware enough to put less focus on them. All these findings are consistent with the one presented in [9]. Therefore, we can expect that RamanNet is learning significant information from the data and ignoring the noises instead of falling into conundrum with confounding factors.

7   Conclusion

Raman spectroscopy has slowly started to gain more attention with the advances in SERS technolgy. The gradual decrease of cost and complexity in computing raman spectrum, is paving the way to large scale raman spectrum data collection for diverse tasks. Therefore, we need suitable machine learning methods to analyze these large scale Raman spectrum data. However, there has not yet been any model developed for the sole purpose of Raman spectroscopy analysis, motivated and designed based on the properties of Raman spectrum. Recently, existing methods like CNN or SVM has shown success in Raman spectrum analysis, but in this work we presented reasonings that such methods may not always be suitable for Raman spectra analysis.

In this work, we present RamanNet, a generalized neural network architecture for Raman spectrum analysis. We take intuitions from the nature of data and design our model accordingly. We propose modifications to the convolutional network behaviors and emulate such operations using multi-layer perceptrons. This adjustment brings the best out of both worlds and it is reflected in evaluating the model on 4 public datasets. Not only, RamanNet outperforms all the state-of-the-art approaches in Raman spectroscopy analysis, it achieves it adopting less complexity. Moreover, RamanNet generates embeddings from spectrum which is much better than what is obtained from PCA, the defacto standard in Raman spectrum analysis. Furthermore, we examined the interpretibility of RamanNet and it was reveled that the model is capable of focusing on the actual important information.

The future direction of this research can be manifold. Firstly, we wish to evaluate RamanNet on more large-scale datasets, as they become public. Secondly, we have not performed any preprocessing or background removal of the spectra, we wish to investigate this further with more dedicated study to infer the denoising capabilities of RamanNet. Last but not least, we also wish to experiment with multiple particle data to assess if RamanNet is capable of identifying, segmenting and extracting the signatures of the different particles.

References

[1] Derek J Gardiner. Introduction to raman scattering. In *Practical Raman Spectroscopy*, pages 1–12. Springer, 1989.

[2] Chandrasekhara Venkata Raman and Kariamanikkam Srinivasa Krishnan. A new type of secondary radiation. *Nature*, 121(3048):501–502, 1928.

[3] Jeroen Cornel, Christian Lindenberg, Jochen Schöll, and Marco Mazzotti. Raman spectroscopy. *Industrial Crystallization Process Monitoring and Control*, pages 93–103, 2012.

[4] Gordon G Hammes. *Spectroscopy for the biological sciences*. John Wiley & Sons, 2005.

[5] Robin R Jones, David C Hooper, Liwu Zhang, Daniel Wolverson, and Ventsislav K Valev. Raman techniques: fundamentals and frontiers. *Nanoscale research letters*, 14(1):1–34, 2019.

[6] Félix Lussier, Vincent Thibault, Benjamin Charron, Gregory Q Wallace, and Jean-Francois Masson. Deep learning and artificial intelligence methods for raman and surface-enhanced raman scattering. *TrAC Trends in Analytical Chemistry*, 124:115796, 2020.

[7] Jinchao Liu, Margarita Osadchy, Lorna Ashton, Michael Foster, Christopher J Solomon, and Stuart J Gibson. Deep convolutional neural networks for raman spectrum recognition: a unified solution. *Analyst*, 142(21):4067–4074, 2017.

[8] Chi-Sing Ho, Neal Jean, Catherine A Hogan, Lena Blackmon, Stefanie S Jeffrey, Mark Holodniy, Niaz Banaei, Amr AE Saleh, Stefano Ermon, and Jennifer Dionne. Rapid identification of pathogenic bacteria using raman spectroscopy and deep learning. *Nature communications*, 10(1):1–8, 2019.

[9] M Erzina, A Trelin, O Guselnikova, B Dvorankova, K Strnadova, A Perminova, P Ulbrich, D Mares, V Jerabek, R Elashnikov, et al. Precise cancer detection via the combination of functionalized sers surfaces and convolutional neural network with independent inputs. *Sensors and Actuators B: Chemical*, 308:127660, 2020.

[10] Hyeon-Joong Yoo. Deep convolution neural networks in computer vision: a review. *IEIE Transactions on Smart Processing and Computing*, 4(1):35–43, 2015.

[11] Serkan Kiranyaz, Turker Ince, Osama Abdeljaber, Onur Avci, and Moncef Gabbouj. 1-d convolutional neural networks for signal processing applications. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8360–8364. IEEE, 2019.

[12] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

[13] Michel Verleysen and Damien François. The curse of dimensionality in data mining and time series prediction. In *International work-conference on artificial neural networks*, pages 758–770. Springer, 2005.

[14] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015.

[15] Andrew L Maas, Awni Y Hannun, Andrew Y Ng, et al. Rectifier nonlinearities improve neural network acoustic models. In *Proc. icml*, volume 30, page 3. Citeseer, 2013.

[16] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015.

[17] Hao Yan, Mingxin Yu, Jiabin Xia, Lianqing Zhu, Tao Zhang, and Zhihui Zhu. Tongue squamous cell carcinoma discrimination with raman spectroscopy and convolutional neural networks. *Vibrational Spectroscopy*, 103:102938, 2019.

[18] William John Thrift, Antony Cabuslay, Andrew Benjamin Laird, Saba Ranjbar, Allon I Hochbaum, and Regina Ragan. Surface-enhanced raman scattering-based odor compass: Locating multiple chemical sources and pathogens. *ACS sensors*, 4(9):2311–2319, 2019.

[19] Data and code on serum raman spectroscopy as an efficient primary screening of coronavirus disease in 2019 (covid-19). `https://springernature.figshare.com/articles/dataset/Data_and_code_on_serum_Raman_spectroscopy_as_an_efficient_primary_screening_of_coronavirus_disease_in_2019_COVID-19_/12159924/1`, (Last accessed on July 2021).

[20] Gang Yin, Lintao Li, Shun Lu, Yu Yin, Yuanzhang Su, Yilan Zeng, Mei Luo, Maohua Ma, Hongyan Zhou, Lucia Orlandini, et al. An efficient primary screening of covid-19 by serum raman spectroscopy. *Journal of Raman Spectroscopy*, 52(5):949–958, 2021.

[21] Barbara Lafuente, Robert T Downs, Hexiong Yang, and Nate Stone. 1. the power of databases: The rruff project. In *Highlights in mineralogical crystallography*, pages 1–30. De Gruyter (O), 2015.

[22] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. Tensorflow: A system for large-scale machine learning. In *12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16)*, pages 265–283, 2016.

[23] Quanshi Zhang and Song-Chun Zhu. Visual interpretability for deep learning: a survey. *arXiv preprint arXiv:1802.00614*, 2018.

[24] Nabil Ibtehaz, Muhammad EH Chowdhury, Amith Khandakar, Serkan Kiranyaz, M Sohel Rahman, Anas Tahir, Yazan Qiblawey, and Tawsifur Rahman. Edith: Ecg biometrics aided by deep learning for reliable individual authentication. *arXiv preprint arXiv:2102.08026*, 2021.

[25] Tawsifur Rahman, Amith Khandakar, Muhammad Abdul Kadir, Khandaker Rejaul Islam, Khandakar F Islam, Rashid Mazhar, Tahir Hamid, Mohammad Tariqul Islam, Saad Kashem, Zaid Bin Mahbub, et al. Reliable tuberculosis detection using chest x-ray with deep learning, segmentation and visualization. *IEEE Access*, 8:191586–191601, 2020.

[26] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 4765–4774. Curran Associates, Inc., 2017.

[27] Slundberg/shap: A game theoretic approach to explain the output of any machine learning model. `https://github.com/slundberg/shap`. Accessed: November 7, 2021.