Bioinformatics, 34(7), 2018, 1215–1223
doi: 10.1093/bioinformatics/btx724
Advance Access Publication Date: 8 November 2017
Original Paper

Bioimage informatics

Deep learning for tumor classification in imaging mass spectrometry

Jens Behrmann[1,*], Christian Etmann[1,*], Tobias Boskamp[1,2], Rita Casadonte[3], Jörg Kriegsmann[3,4] and Peter Maaβ[1,2]

[1]Center for Industrial Mathematics, University of Bremen, 28359 Bremen, Germany, [2]SCiLS, 28359 Bremen, Germany, [3]Proteopath GmbH, 54296 Trier, Germany and [4]Center for Histology, Cytology and Molecular Diagnosis, 54296 Trier, Germany

*To whom correspondence should be addressed.

Associate Editor: Robert Murphy

Received on May 3, 2017; revised on October 26, 2017; editorial decision on November 6, 2017; accepted on November 7, 2017

Abstract

Motivation: Tumor classification using imaging mass spectrometry (IMS) data has a high potential for future applications in pathology. Due to the complexity and size of the data, automated feature extraction and classification steps are required to fully process the data. Since mass spectra exhibit certain structural similarities to image data, deep learning may offer a promising strategy for classification of IMS data as it has been successfully applied to image classification.

Results: Methodologically, we propose an adapted architecture based on deep convolutional networks to handle the characteristics of mass spectrometry data, as well as a strategy to interpret the learned model in the spectral domain based on a sensitivity analysis. The proposed methods are evaluated on two algorithmically challenging tumor classification tasks and compared to a baseline approach. Competitiveness of the proposed methods is shown on both tasks by studying the performance via cross-validation. Moreover, the learned models are analyzed by the proposed sensitivity analysis revealing biologically plausible effects as well as confounding factors of the considered tasks. Thus, this study may serve as a starting point for further development of deep learning approaches in IMS classification tasks.

Availability and implementation: https://gitlab.informatik.uni-bremen.de/digipath/Deep_Learning_for_Tumor_Classification_in_IMS.

Contact: jbehrmann@uni-bremen.de or christianetmann@uni-bremen.de

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Imaging mass spectrometry (IMS) has matured as a label-free technique for spatially resolved molecular analysis of small to large molecules. Given a thin tissue section, mass spectra are recorded at multiple spatial positions on the tissue yielding an image where each spot represents a mass spectrum. These spectra relate the molecular masses to their relative molecular abundances and thus offer insights into the chemical composition of a region within the tissue (see e.g. Stoeckli *et al.*, 2001). In this article, we consider matrix-assisted laser desorption/ionization imaging mass spectrometry (MALDI-IMS) (Caprioli *et al.*, 1997) for our study. However, the analysis and methods should also be applicable to other IMS modalities like secondary ion mass spectrometry (SIMS) (Benninghoven and Loebach, 1971).

In MALDI-IMS molecules of interest co-crystallize with an organic matrix compound that assists in the desorption and ionization of the molecules on irradiation with a laser beam. This sample preparation of applying a matrix in the last step is also applicable to formalin-fixed paraffin-embedded (FFPE) tissue, a common tissue storage solution in pathology. Hence, MALDI-IMS has a high potential for many pathological applications, as discussed by Aichler and Walch (2015) and Kriegsmann *et al.* (2015). One of the main advantages of MALDI-IMS is that it allows high-throughput

analysis of several tumor cores from different patients by arranging them in a single tissue microarray (TMA) (Casadonte *et al.*, 2017). Thus, within a single run of the mass spectrometer a large cohort of potentially cancerous tissue can be analyzed to extract biochemical information in a spatial manner. This biochemical information may then be used for the determination of the cancer subtypes or the identification of the origin of the primary tumor in patients with metastatic disease, where accurate typing of a tumor is crucial for successful treatment of patients. For related studies see e.g. Casadonte *et al.* (2014).

While current MALDI-IMS instruments are able to acquire molecular information at high spatial resolution ($< 20\,\mu m$ center-to-center spacing between each ablated laser spot) at short measurement times (>20 pixels/s), advanced bioinformatic tools may help to extract knowledge in a robust manner. This has been recognized as a challenging task in bioinformatics as it involves analyzing spatially distributed high-dimensional spectra (Alexandrov, 2012). Especially in tumor classification, a robust feature extraction procedure is required to integrate this workflow into a reliable routine. Even before the feature extraction, pre-processing by incorporating spatial relationships between spectra was suggested by Alexandrov and Kobarg (2011) for clustering large tissue regions into smaller subregions based on spectral similarities. We, however, focus on processing each spectrum separately as spectra in our application are measured from small tissue core regions (diameter of 0.6–1.0 mm) with little varying structure within a single core. This core represents only a small portion of the original tumor biopsy ($\sim 20 \times 20$ mm) from which it was extracted. Thus, the morphological heterogeneity of such tissue cores is strongly reduced when compared with the original tumor sample. In this study, $\geq 80\%$ of the tissue analyzed was composed of tumor cells only.

A common approach for the extraction of meaningful features is based on finding significant signal peaks, often referred to as peak detection. These peaks are then expected to be useful for discriminating spectra from different classes (Yang *et al.*, 2009). Some more advanced methods are designed to retrieve molecular signatures (Harn *et al.*, 2015) or characteristic patterns of the data (Boskamp *et al.*, 2016), in order to combine information from several correlated spectral features into a lower dimensional representation of the original data. After feature extraction, supervised classification methods like linear discriminant analysis (LDA) are used to classify features into tumor types (Boskamp *et al.*, 2016). For a review on machine learning methods for MALDI-IMS data see Galli *et al.* (2016).

Beside this classical approach of a separate feature extraction and classification stage, end-to-end learning where features and classification are learned in one step offers a promising alternative. The last years have seen a dramatic performance increase in several challenging tasks like image classification or speech recognition by these end-to-end learning methods. Usually, these methods are referred to as deep learning (LeCun *et al.*, 2015). Mostly, deep learning is realized by convolutional neural networks (CNNs), which compute several convolutional and non-linear transforms of the input data to retrieve high-level abstractions for the final classification stage of the network (Goodfellow *et al.*, 2016).

Due to this astonishing success in various other fields where also high-dimensional data are analyzed, we aim to discuss how to use deep learning by CNNs for tumor classification in MALDI-IMS. Our main intention is the introduction of deep learning to this field by applying it to mass spectra. Second, we derive adapted CNN architectures, as mass spectra are still quite different from RGB images, for which CNNs were originally designed. Last, we discuss ways to interpret the learned model and analyze if biologically plausible effects are visible, a crucial step for applying it to tumor diagnostics.

Deep learning has been introduced to IMS data prior to this work, but with a focus on unsupervised dimension reduction methods, see Thomas *et al.* (2016) where autoencoders were used to reduce rat brain IMS data. Moreover, Inglese *et al.* (2017) introduced a neural network based dimension reduction to find metabolic regions within tumors. However, we focus on a fully supervised deep learning approach, which is novel for large-scale tumor classification with IMS data.

In this study, we test the proposed methods on two IMS datasets, both comprised of several TMAs of a cohort of tissue cores. The first classification task (8 TMAs) is to distinguish two lung tumor subtypes, namely adenocarcinoma from squamous cell carcinoma, whereas the second task (12 TMAs) is to discriminate lung and pancreas tumors. These datasets have been used in two prior works by Kriegsmann *et al.* (2016) and Boskamp *et al.* (2016), but are used in this study to verify the potential of deep learning methods for tumor classification in IMS.

2 Materials and methods

2.1 Samples, MALDI-IMS and pre-processing

Sample acquisition, preparation, MALDI-IMS measurement and data pre-processing are described in more detail in the study of Boskamp *et al.* (2016). In short, FFPE samples were provided by the tissue bank of the National Center for Tumor Diseases (Heidelberg, Germany). Tumor status and typing of all cores were confirmed by standard histopathological examination of hematoxylin and eosin (H&E) stained slides and additional immunohistochemical (IHC) stains. Cylindrical tissue cores of all tissue samples were assembled to 12 TMA blocks, see Figure 1 (left). Tissue sample preparation for MALDI-IMS measurement was performed according to a previously published protocol (Casadonte and Caprioli, 2011), including tryptic digestion of proteins to peptides.

After the application of an MALDI matrix solution onto digested sections, MALDI-IMS data were acquired using an MALDI time-of-flight (TOF) instrument (Autoflex speed, Bruker Daltonics) in positive ion reflector mode. Spectra were measured in the mass range of 500–5000 *m/z* at 150 µm spacing between spot centers using 1600 laser shots per position. After measurement, the raw MALDI-IMS data were combined into a single dataset using the SCiLS Lab software (version 2016a, SCiLS, Bremen, Germany), followed by baseline correction using the default setting (convolution width of 20). Note that the influence of this setting is not studied here and requires further investigations. Next, the data were imported into MATLAB R2016b (MathWorks, Natick, MA) for further analysis using our MATLAB library *MSClassifyLib*. Implementation and computation of CNNs were performed using the Theano 0.8-based (Theano Team, 2016), libraries Lasagne 0.2.dev1 and nolearn 0.6 in Python. Finally, the results were loaded to the *MSClassifyLib* for further evaluation.

2.2 Deep CNNs

After pre-processing the data, each spectrum measured in a tissue spot by IMS is handled separately, see e.g. spectra in Figure 1. Henceforth, a spectrum is denoted as a data point $x \in R^d$, where d denotes the number of *m/z*-bins, for example $d = 27\,286$ in the conducted experiments. These spectra can thus be viewed as structured data points on a pre-defined grid (the *m/z*-bins). In this regard

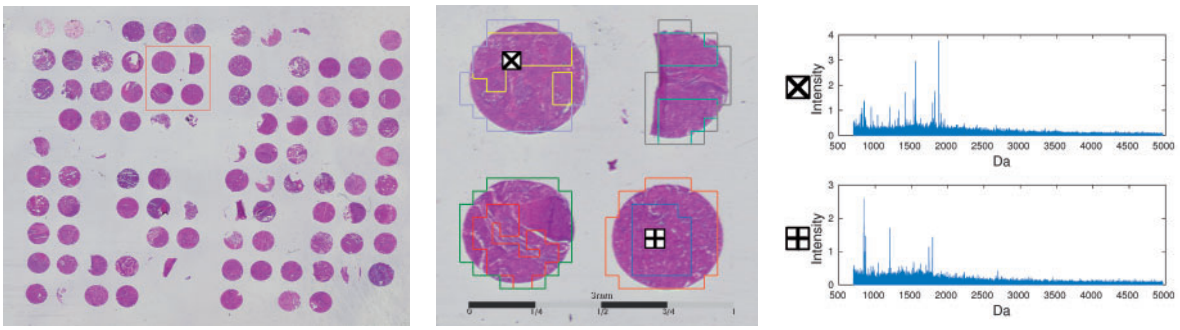

Fig. 1. Overview on structural hierarchies of the IMS data, from TMA to tissue core to a single spectrum. Left, an H&E image of a TMA is shown, which is measured in a single IMS measurement. The box (left) marks the four tissue cores shown in the middle. These tissue cores have two annotations, the outer region marks the measurement region for the laser, while the inner region marks the ROI annotated by a pathologist. Furthermore, the x-marked and +-marked squares correspond to a spot of the imaging data. Each of these spots correspond to a mass spectrum shown in the right figure

spectra are similar to images, where the grid is given by the pixels. Hence, mass spectra can be understood as one-dimensional images. However, one major difference is that the underlying grid of these spectra is not necessarily equidistant (in contrast to images). Still, it seems reasonable to assume that methods commonly applied to image classification might also be suitable for mass spectra.

Over the course of the last few years, deep CNNs have led to major breakthroughs in many computer vision applications, especially image classification (Krizhevsky *et al.*, 2012). One key observation is that the depth of the employed neural networks (i.e. the number of layers) is instrumental in achieving high accuracies. This concept is commonly known as *deep learning* and has been successfully applied to numerous other tasks like localization or speech recognition (LeCun *et al.*, 2015). In this section, a description of the involved techniques will be given. For more in-depth information, interested readers are referred to Goodfellow *et al.* (2016).

A neural network for classification is a function

$$f_\theta : R^d \to (0,1)^C \text{ with } \sum_{j=1}^{C} f_\theta(x)_j = 1, \tag{1}$$

where C is the number of classes of the considered classification problem and θ is a parameter vector, which the neural network depends on. The individual entries of the vector $f_\theta(x)$ can be regarded as the estimated probabilities of x belonging to each respective class. The class with the highest probability is in turn assigned to the spectrum x.

As the behavior of the neural network is governed by the parameter vector θ, it needs to be tuned appropriately. This is achieved by first choosing a labeled *training set* $\mathcal{T} = \{(x^{(i)}, y^{(i)})\}_{i=1,\ldots,N}$, where $y^{(i)}$ represents the correct class label of the data point $x^{(i)}$. For example, a label could be the specific tumor type of a spectrum. The class labels are encoded as C-dimensional standard unit vectors, so that e.g. $(0,1,0)^T$ represents Class 2 in a 3-class problem. Then, the average *negative log-likelihood error* over the training set combined with a regularization term yields the cost function

$$J(\theta;\mathcal{T}) = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C} y_j^{(i)} \log\left(f_\theta\left(x^{(i)}\right)_j\right) + \lambda||\theta||_2^2, \tag{2}$$

which is minimized with respect to θ. The *weight decay parameter* $\lambda > 0$ is a regularization parameter intended to prevent overfitting to the training set. The minimization problem (2) can be approximately solved by iteratively performing gradient descent steps

$$\theta \leftarrow \theta - \eta\nabla_\theta J(\theta;\mathcal{T}), \tag{3}$$

given by the backpropagation algorithm, where $\eta > 0$ is called the *learning rate*. Since the gradient has to be computed for every sample in each iteration, this approach results in a high computational cost. Instead, for every iteration over the training set, $\mathcal{T}$ is randomly partitioned into M much smaller *mini-batches* (e.g. with 128 elements). For one such mini-batch $\mathcal{B}$ we have $J(\theta;\mathcal{B}) \approx J(\theta;\mathcal{T})$, such that instead of parameter update (3), a stochastic gradient descent (SGD) update

$$\theta \leftarrow \theta - \eta\nabla_\theta J(\theta;\mathcal{B}), \tag{4}$$

can be performed. This means that for roughly the same computational cost of performing one gradient step (3), M gradient steps (4) can be performed (an *epoch*). This also means that the training of a neural network only has a complexity of $\mathcal{O}(N)$ for a fixed number of epochs. Furthermore, in practice larger N often enable a lower setting of the number of epochs. Using mini-batches $\mathcal{B}$, the memory usage can be limited, such that even arbitrarily large IMS datasets could be processed. In this paper, an adaptive modification of (4) is used, which is called *Adam* (Kingma *et al.*, 2012).

Usually, f_θ can be seen as a composition of L (generally non-linear and parametric) functions, $f_\theta = g_L \circ \ldots \circ g_1$, where the g_k are called the layers and L is called the *depth* of the *feedforward neural network*. This composition means that the output of a layer serves as the input of the next layer, hence the term *feedforward*. There are several common types of layers, four of which are defined in the following.

A *fully connected layer* is defined as $g(x) = \zeta(Wx + b)$, where the *bias vector* b has the same number of rows as W. Often, the so-called *activation function* ζ is chosen as the element-wise application of the *rectified linear unit* function $\mathrm{ReLU}(t) = \max(0,t)$, which results in sparse function values and an easy optimization (Glorot *et al.*, 2011). For the last layer g_L, ζ is chosen to be the *softmax* function defined as

$$\zeta(x)_j = \frac{\exp(x_j)}{\sum_k \exp(x_k)}, \tag{5}$$

which ensures that the output may be understood as a probability.

In contrast to pre-multiplying their input with a large matrix, convolutional layers (LeCun *et al.*, 1989) work by convolving their

input with several small filter kernels. A convolutional layer g is defined as

$$g(x)^{\langle j \rangle} = \zeta\left(\sum_k K^{\langle j,k \rangle} * x^{\langle k \rangle} + b^{\langle j \rangle}\right), \quad (6)$$

where $x^{\langle k \rangle}$ denotes the kth channel (or *feature map*) of x and $K^{\langle j,k \rangle}$ denotes the filter kernel between the kth feature map of the input and the jth feature map of the output. In particular, this can be used for image data, where there are three RGB channels. In contrast, mass spectra only have one channel. With convolutional layers however, the number of feature maps can be increased throughout the network, to extract many different types of features, e.g. varying peak shapes due to near-isobar chemicals and non-equidistant binning as further elaborated upon in Section 1.1.3 (see Supplementary Material).

Residual layers are a recent innovation of convolutional layers, which have seen great success for image classification (He *et al.*, 2015). These layers introduce *residual connections*, which essentially allow their input to bypass other layers. A residual layer g of depth r is defined as

$$g(x) = x + c_r(c_{r-1}(\ldots(c_1(x)))), \quad (7)$$

where the c_k are appropriate convolutional layers. In theory, any number of residual layers could be inserted into a neural network without harm, since the network can always learn $c_r \circ \ldots \circ c_1$ to be the zero mapping, turning g into the identity mapping. Because of this, they make very deep networks possible. If the number of feature maps changes in the convolutional portion or if strided convolutions are employed, the addition in (7) is not well defined any more. In this case, x is replaced by an appropriate convolutional layer with *id* as its activation function, where each $K^{\langle j,k \rangle} \in R$ and $b^{\langle k \rangle} = 0$.

While the standard convolution slides a small window over the input, where the values of the sliding window are constant over the whole domain of the input, *locally connected layers* use a different 'convolution', where the values of this sliding window may differ depending on its location. This is therefore sometimes called *unshared convolution* (Goodfellow *et al.*, 2016). Another important step of CNNs is downsampling in order to decrease the dimensionality, realized by *strided convolutions* (Goodfellow *et al.*, 2016) in this paper. This operation applies the convolutional kernel with a step size (*stride*) larger than one, resulting in subsampling by the factor of the size of those steps.

Fully connected layers, convolutional/residual layers and locally connected layers all have weight matrices or kernels as well as biases. These are the parameters that comprise θ, which are learned through the above-mentioned training algorithm.

2.3 Architectures for IMS

A main driving force in the design of deep CNNs for images is first the need to handle high-dimensional data, which is why the idea of convolutional transforms with their few parameters of the filter kernel plays a key role. Secondly, the layered architecture is motivated by extracting features from different levels of abstraction. While the first layers may be able to extract edges in images, the goal of higher layers is to extract more complex shapes like curves or even entire structures like faces of humans (LeCun *et al.*, 2015). However, the application of these concepts to spectra from IMS data poses the question on how these operations may act in this domain. As discussed in Section 2.2, IMS spectra are also high-dimensional data on a grid, the *m*/*z*-bins. Hence, CNNs can offer the same remedy for working in a high-dimensional domain by grouping neighboring *m*/*z*-bins together through convolutions. As the spectra are transformed through the network, this grouping of neighboring *m*/*z*-bins grows, resulting in lower dimensional data through subsampling these groups by pooling or strided convolution.

Altogether, the same idea may be transferable to IMS data to extract features from high-dimensional data, however it is important to discuss the underlying assumptions. The main assumption of a convolutional transform is that neighboring *m*/*z*-bins are correlated, which can be exploited by a filter kernel. This is certainly plausible when considering raw data from a TOF mass spectrometer, where a peak is spread over several *m*/*z*-bins. On the other hand, deep CNNs perform the mentioned grouping also on transformed data to extract higher level features, where its impact onto spectral data is less obvious. While peaks may be the counterpart of edges in images, mid-level features may be represented by isotope patterns or even adduct patterns of the same peptide. See for example Harn *et al.* (2015), where use of these adducts is made to extract patterns from the data. On the highest level, tryptic-digested proteins may contribute to several measured peptides, resulting in patterns across the entire mass range. See for example Boskamp *et al.* (2016), where the idea is to extract these characteristic spectral patterns. The key difference between these patterns is their position on the mass grid. While isotope patterns can be considered local as they are formed in a small, connected mass interval, protein patterns are non-local as the digested peptides may be spread irregularly over the entire measurement range.

This assumption of the composition of the spectral data led us to an adapted architecture design for IMS spectra. We estimate the number of *m*/*z*-bins of large measurable isotope patterns of peptides based on a model of an average amino acid, see Section 1.1.1 (Supplementary Material). Using this estimate, we restrict the local grouping roughly to the size of large isotope patterns, such that one variable is able to encode such a local feature. However, it should be noted that these groups (called *receptive fields*) are partially overlapping, such that each variable may encode more than one and just parts of an isotope pattern. Furthermore, the downsampling rate by *strided convolutions* are determined by an average distance between peaks. Together with the filter sizes and number of filters, these design choices are crucial, which is why we discuss these aspects and the non-equidistant binning in more detail in Section 1.1.2 (see Supplementary Material).

After the convolutional transforms, a locally connected layer (see Section 2.2) is used to process also those local input features, which are encoded in the two neighboring variables. Furthermore, this operation enables the network to handle each local region differently due to *unshared weights*. Hence, a focus only on important peptides for the given classification task is possible, as exemplified for a squamous cell carcinoma tumor spectrum in Figure 2. Moreover, we compare the proposed architecture to a deep Residual Network (He *et al.*, 2015), the state-of-the-art design principle in image classification. Details of both architectures are listed in Supplementary Tables S2.2 and 2.3.

However, it needs to be emphasized that the adapted architecture is geared towards TOF data with a focus on peptide measurements and is only a step towards a deep network design based on domain knowledge. Especially for other measurement objectives like the extraction of metabolites or non-TOF data, designing a deep network is still to be explored. A further discussion on the transfer to other IMS domains is given in Section 1.1.4 (see Supplementary Material).

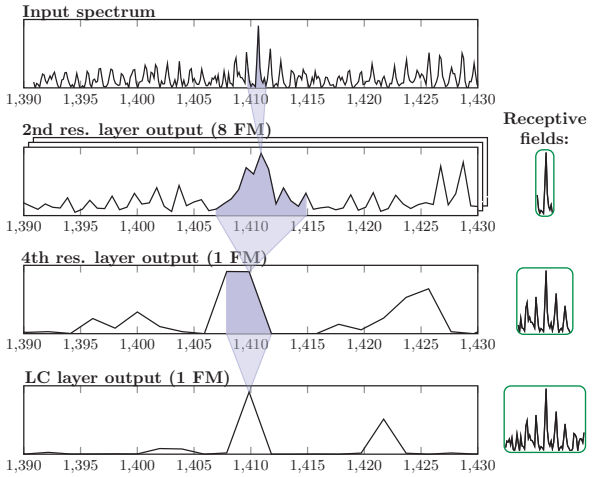

Fig. 2. Overview over the working principle of *IsotopeNet*. The first row shows a section of a recorded mass spectrum of a squamous cell carcinoma. Several residual layers of depths 2 extract interesting features of small portions of the previous layers' outputs. Due to their consecutive (and partially strided) convolutions, an increasingly large portion of the input spectrum influences each spot in the deeper layers of the neural network. This is signified by the *receptive fields* on the right-hand side, which reach the size of a whole isotopic pattern after the fourth residual layer (before the locally connected layer)

2.4 Interpretation via sensitivity analysis

Not only the accurate prediction, but also the interpretability from a biological point of view is crucial for application of automated analysis tools to tumor typing. Especially connecting the learned model to known tumor biomarkers is a step towards developing trust in an automated model. In traditional feature extraction methods like peak picking (Yang *et al.*, 2009), the *m/z*-value may be used for identification. But deep learning with an end-to-end feature extraction process through layers of a neural network does not allow this straightforward approach of analyzing features in the input domain (Zeiler and Fergus, 2013).

However, instead of analyzing the features extracted by the network, an evaluation of the relationship between predicted class probabilities and each input *m/z*-value is possible. Mathematically, this output–input relationship of the network $f_\theta : R^d \to (0,1)^C$ can be linearly approximated by the gradient $\nabla_x f_\theta(x)_j$, where $x \in R^d$ denotes a spectrum and $f_\theta(x)_j$ is the predicted probability of class j by the network. This measures how sensitive the prediction is with respect to changes in certain *m/z*-values, which has been introduced for images as the saliency map (Simonyan *et al.*, 2013). As CNNs are not only differentiable with respect to parameter θ, but also to its input x, the sensitivity can be efficiently computed via backpropagation.

To compare the sensitivity of different *m/z*-values, a normalization per dimension is necessary as the sensitivity of each dimension is dependent on the scale of the corresponding input variation. Thus, a scaling

$$\text{sens}\left(x^{(i)}\right)_{jk} = \sigma_k \cdot \left(\nabla_x f_\theta\left(x^{(i)}\right)_j\right)_k \tag{8}$$

of the sensitivity of sample $x^{(i)}$ for class j and *m/z*-value k is conducted by the standard deviation $\sigma_k = \text{std}\left(\{x_k^{(i)} | i = 1, \ldots, N\}\right)$. A motivation for this scaling is given in Chapter 2 (see Supplementary Material). Furthermore, the gradient $\nabla_x f_\theta(x)_j$ is computed per sample x, which only allows an interpretation by example. To make more general statements of the model behavior, we average (8) over the training set.

3 Results

3.1 Datasets and evaluation

In this study, we test the proposed CNNs on two challenging real-world datasets consisting of 12 MALDI-IMS measurements of a large cohort of tumor tissue cores. In this comparison we use the same setting as the previous study by Boskamp *et al.* (2016), to establish a solid comparison of the proposed methods to other common approaches. Supplementary Table S1 shows the details of each TMA, where the selection of cores was done to include only cores with a significant portion of tumor tissue and to obtain an approximately balanced number of spectra per patient across all TMAs (Boskamp *et al.*, 2016). From this dataset we derive two different classification tasks: tumor subtyping of adenocarcinoma versus squamous cell carcinoma (called task ADSQ) and primary tumor typing of lung versus pancreas tumor (called task LP). Note that there are several tissue cores collected per patient (one or two in lung TMAs, three on average in pancreas TMAs). Furthermore, in the lung dataset there are also annotated subregions called Regions of Interest (ROI), see Figure 1. These regions were marked by a pathologist as relevant subregions within the tissue core for subtyping the tumor. To perform classification only on those subregions, only those spectra within each ROI are used for task ADSQ, resulting in a reduced number of spectra of 4672. Note that previous studies using whole tissue cores yielded inferior results. On the other hand, for task LP the entire tissue core was used, which also include spots with non-tumor cells, resulting in a total of 27 475 spectra.

For evaluation of performance we used randomized 4-fold cross-validation on TMA level, see Supplementary Table S2. The predicted labels on the test set are obtained by taking the class with the highest predicted probability, see Equation (1). Then these labels are compared to the ground truth for each spectrum (spot level evaluation). For evaluation of core performance, the predicted class is assigned to each core by the majority of predicted labels within the core (core level evaluation). As a single performance measure we used the balanced accuracy $\text{balAcc} = \frac{1}{2}\left(\frac{TP}{P} + \frac{TN}{N}\right)$, where *TP/P* denotes *true positive/positive* ($j = 1$) and *TN/N* denotes *true negative/negative* ($j = 2$). This measure is, unlike the accuracy, not biased by the relative class proportions in the data. Within cross-validation, the median balanced accuracy of the four cross-validation runs is used.

As a baseline method we use a feature extraction based on discriminative *m/z*-values to set the proposed deep learning methods in contrast to straightforward approaches. This method aims at identifying individual *m/z*-values by computing the Mann–Whitney–Wilcoxon statistic for each *m/z*-value separately (*ROC* method). After computing this statistic, we perform a selection of discriminative *m/z*-values by taking those K features with the highest test statistic in a range from $K = 5$ to $K = 100$. Subsequent to feature extraction by discriminative *m/z*-values, an LDA classifier is used, a standard algorithm for creating classification models (Hastie *et al.*, 2001). It should be noted that this method was used in the study of Boskamp *et al.* (2016) as a baseline as well.

3.2 Model comparison

To train a deep CNN, several parameters have to be set appropriately. First of all, the architecture (size of each layer, filter kernel size, etc.) has to be specified. In this comparison we used a deep

Residual Network as a state-of-the-art approach in image classification and a specialized architecture for IMS (named *IsotopeNet*), as discussed in Section 2.3. The second parameter set specifies the training of the neural networks. Both the training parameters and the design of the architectures can exert a substantial influence on the classification performance. Hence, we further discuss the design of the architectures in more depth in Section 1.1 (see Supplementary Material). Moreover, an overview of training parameters settings, as well as a discussion on the influence and a motivation of each setting, is given in Section 1.2 (see Supplementary Material). Apart from the described settings, each layer of both architectures was further normalized via batch normalization (Ioffe and Szegedy, 2015).

Prior to feature extraction, all spectra were normalized by the total ion count measure (Deininger *et al.*, 2011). Figure 3 (left) reports the results on both tasks ADSQ and LP, where the method *ROC/LDA* refers to a feature extraction based on discriminative *m/z*-values followed by an LDA, see Section 2.3. For this baseline method we report the worst and the best performance over the number of features K from $K = 5$ to $K = 100$ to get an impression of the variance. For task ADSQ *ROC/LDA* reaches a balanced accuracy of 78.7% on spot level and 82.7% by aggregation on cores, while the performance for task LP is about 5% higher.

Figure 3 (left) further shows how the *ResidualNet* compares to the domain adapted architecture *IsotopeNet*. Due to the stochasticity of the training process through SGD [Equation (4)], random initialization and regularization by dropout (Srirastava *et al.*, 2014), both methods were run four times using the same parameter setting. From those four runs the median balanced accuracy is reported to get a robust idea of the average performance. Furthermore, the interquartile range is stated below the median to estimate the variance induced by the mentioned stochasticity. Overall, the domain adapted architecture *IsotopeNet* performs better than both *ResidualNet* and *ROC/LDA*, for example with a spot level balanced accuracy of 84.5% for task ADSQ.

Although the previous discussion considered the variance of several runs over the entire dataset, Figure 3 (right) visualizes the variance over the cross-validation folds on spot level. For this box plot, the balanced accuracy of the four identical runs was computed for each fold. For *ROC/LDA*, however, only the best model over the number of features was selected. As visible from the median line, *IsotopeNet* outperforms the other methods on both tasks. Furthermore, the variance is lower but still rather large and outliers

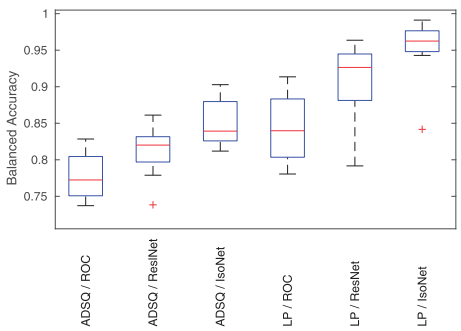

Fig. 3. Left: Comparison table of 4-fold cross-validation on both tasks. For ROC/LDA the worst and best results over 5–100 features are reported in the table. For ResidualNet and IsotopeNet, the table shows the median obtained from four runs with identical parameter settings, together with the interquartile range to estimate the spread. The core level results are obtained by taking the majority of the predicted label. Right: Boxplot of the balanced accuracy from each method over the four cross-validation folds, reported on spot level for both tasks

(+) occur for both methods. Hence, the impact of the choice of the splitting between training and test may have an influence, which is why a conclusion based on small performance differences may be too early. Moreover, Figure 3 shows that task LP seems to be easier for all methods. This is expected, as the task to differentiate primary tumor is most likely easier and more spectra were available, which is especially crucial for deep learning.

In addition to the test set performance discussed previously, the training set performance can be used to judge overfitting of methods. For example, *IsotopeNet* consistently reached a training balanced accuracy of about 95% on task ADSQ, whereas *ResidualNet* had a balanced accuracy of >98%. This effect may be explained by the number of parameters shown in Table 1, as *ResidualNet* is by far the larger model. Furthermore, the runtime per epoch is reported in this table, which further underlines the effectiveness of *IsotopeNet*. The reported tests were conducted on a powerful graphics card GeForce GTX TITAN X (Nvidia) and computations were compiled to CUDA via the Python framework Theano 0.8 (Theano Team, 2016). A further extensive parameter search for both networks may improve the results, especially for *ResidualNet*, but it is out of the scope of this study.

3.3 Interpretation of models

As mentioned in Section 2.4, competitive performance is only the first step towards the acceptance of an automated model for tumor typing. Interpretation from a biological point of view is crucial to uncover the strengths and weaknesses of a model. The common approach is to look for discriminative *m/z*-values of the feature extraction process, which is straightforward for the baseline in this paper, previously called *ROC/LDA*. After finding these *m/z*-values, an identification process by MS/MS technology has to be conducted. See for example Kriegsmann *et al.* (2016), where an identification analysis of the differentially expressed peptide ions was directly conducted by MS/MS on digested tissue for task ADSQ. However, finding the most significant *m/z*-values for deep learning models is more involved. Thus, for interpretation we rely on the described sensitivity analysis, see Section 2.4.

The goal of this section is to analyze the proposed *IsotopeNet* for both tasks. This is done by considering the best network out of four consecutive runs with the same setting. Then, the model from the best cross-validation fold is taken into consideration. After choosing the model, the class under examination is chosen (AD for task ADSQ, Lung for task LP). Finally, the sensitivity $sens\left(x^{(i)}\right)_j$ from Equation (8), where j denotes the chosen class, is computed for all spectra in the training data ($i = 1, \ldots, N$) and the mean over the samples i is taken.

In Figure 4 (left), the sensitivity for task ADSQ from the best performing *IsotopeNet* is shown. This sensitivity has the same dimension as a raw spectrum, which makes interpretation in the input domain feasible. In contrast to spectra, negative values occur in the sensitivity as well. The sign of a value indicates the slope direction, which means that positive values indicate a positive slope in the direction of higher probabilities for AD. On the other hand, negative values indicate a negative slope for AD, which in turn means that an increase of intensities with negative slope will result in higher probabilities for the other class, SQ. Hence, both the sign and the height of each peak in the sensitivity map in Figure 4 are important.

Most apparent in this sensitivity is the concentration in the area of 1000–2000 *m/z*, which is to be expected as most peptides are measured in this range. As the zoom in Figure 4 (middle) shows, high peaks are found at 1406.6 Da and 1410.7 Da, together with a

Table 1. Comparison table of both architectures showing the number of trainable parameter and the runtime per epoch (iteration of SGD over the entire training set)

Method	Runtime (ADSQ) per epoch (s)	Runtime (LP) per epoch (s)	Number of parameters
ResidualNet	44.55	109.44	2 132 130
IsotopeNet	14.16	35.34	13 935

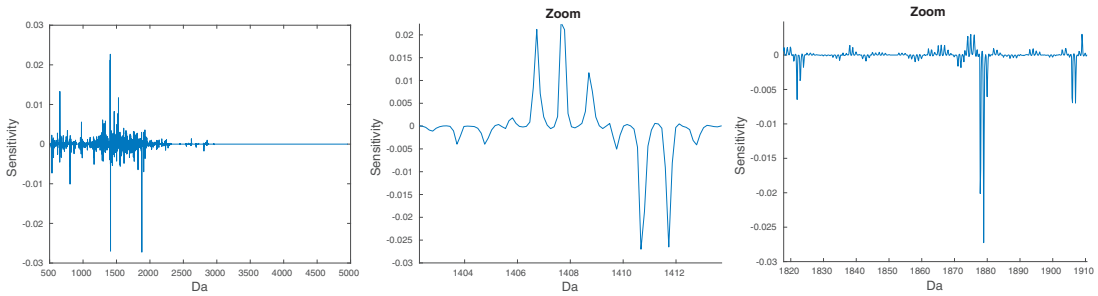

Fig. 4. Left: Sensitivity of *IsotopeNet* on task ADSQ, where the sensitivity (see Section 2.4) was computed for the predicted probability of class AD. Middle: Zoom in with high peaks at 1406.6 and 1410.7 Da. Right: Zoom in with high peaks at 1821.8, 1877.8 and 1905.9 Da

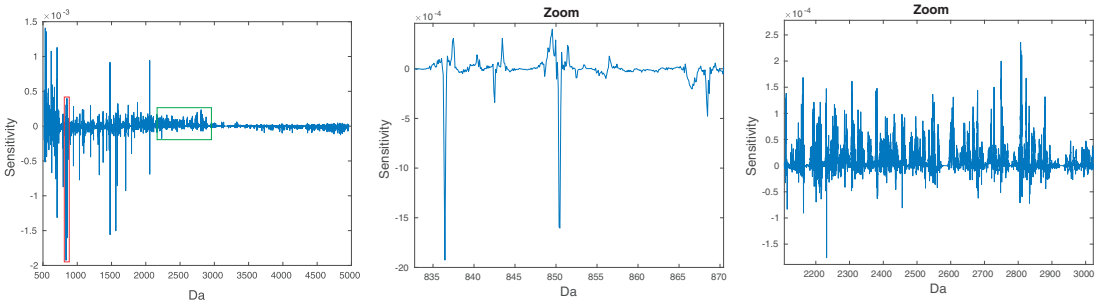

Fig. 5. Left: Sensitivity of *IsotopeNet* on task LP, where the sensitivity (see Section 2.4) was computed for the predicted probability of class Lung. Middle: Zoom in with high peaks at 836.5, 852.4 and 868.5 Da. Right: Zoom in showing high oscillations in the range of 2100–2900 Da

small isotope pattern. The positive peak 1406.6 Da acts as a marker for AD, while 1410.7 Da has a negative value and thus marks SQ. Most importantly, both peaks have been identified by Kriegsmann *et al.* (2016) as a peptide of cytokeratin-7 (CK7, 1406.6 Da) and cytokeratin-5 (CK5, 1410.7 Da). Additionally, the zoom in Figure 4 (right) shows a pattern at 1821.8 Da, a more expressed pattern at 1877.8 Da and a less expressed pattern at 1905.9 Da. Again, these were identified by Kriegsmann *et al.* (2016) as peptides of cytokeratin-15 (CK15, 1821.8 Da and at 1877.8 Da) and heat shock protein beta-1 (HSP27, 1905.9 Da). Out of these four markers, CK5 and CK7 are already well-known IHC markers, whereas CK15 and HSP27 are two new potential markers (Kriegsmann *et al.*, 2016). Hence, through analyzing the output–input relationship of the deep CNN by the sensitivity analysis, strong characteristics of the model could be attributed to known markers.

However, the sensitivity of the best model for task LP in Figure 5 appears different at first glance. Compared to the sensitivity for ADSQ, the mass range <1000 Da and high mass range >2000 Da show more activity. The zoom in Figure 5 (middle) shows peaks at 836.5, 852.4 and 868.5 Da, which were observed as discriminative *m*/*z*-values by Boskamp *et al.* (2016). However, the local structure does not show isotope patterns as for the task ADSQ. Furthermore, the zoom at the *m*/*z*-range from 2100 to 2900 Da (Figure 5, right) shows high oscillations almost for the entire interval. Hence, both zooms uncover a substantially different behavior compared to the sensitivities of ADSQ, which suggests flaws in the model. An explanation for these effects may be artifacts induced by the measurement. As reported in Supplementary Table S2.1, the lung and pancreas tissues are spread over separate TMAs and thus a discrimination of samples by classification based on measurement differences and artifacts is also able to classify lung and pancreas tumor. Hence, this is a major confounding factor in the study design, which may also explain the observed higher performances for task LP reported in Figure 3. To conclude, the sensitivity analysis uncovered a discrimination based on correlations between measurement artifacts and class labels, which allows to judge the model and classification task from a different perspective.

4 Discussion

We present a new approach for tumor classification in IMS data based on deep neural networks. Tests were conducted on algorithmically challenging real-world IMS tumor datasets, where the reported results showed the competitiveness of deep learning. However, the main goal of this paper is to establish a starting ground for further research on advanced end-to-end learning methods in the field of IMS. A drawback of training CNNs is the requirement of a powerful graphical processing unit (GPU), yet large

sample sizes can be processed efficiently as training is done on batches, see Equation (4). Hence, even large imaging data with a high number of spectra are processable without further considerations. On the contrary, deep learning is indeed known to improve its performance in big data applications (LeCun *et al.*, 2015). This may also become more relevant for future applications as modern MALDI-IMS instruments like the rapifleX MALDI Tissuetyper (Bruker Daltonics GmbH) provide higher spatial resolution and thereby more spectra per tissue.

Beside the tests on challenging tumor classification tasks, we introduced an adapted architecture to the characteristics of mass spectra. This proposed model was then compared to a standard deep learning approach and displayed superior performance. Moreover, we introduced an analysis tool based on the sensitivity of the output–input relationship, which allows interpretation in the input domain. This analysis revealed biological connections to known biomarkers for the discrimination of adenocarcinoma and squamous cell carcinoma. On the other hand, this interpretation approach also revealed model artifacts that disturbed the discrimination of lung and pancreas primary tumors. Due to a confounding induced by separate measurements, discrimination can be supported by simply looking at differences in measurement characteristics. Thus, the sensitivity analysis provides hints to assessing the model's validity, a major issue in data-based modeling as only hold-out test data are available to estimate future real-world performance.

On a methodological level, we plan to incorporate further domain knowledge like the characteristic shape of peaks or even known biomarker into our deep neural network. For example, the biomarkers described in Section 3.3 could be used to guide the behavior of the network. Moreover, better regularization methods are required to deal with tasks of small sample size. Even in this study we frequently observed large gaps between training and test performance due to overfitting. Hence, advanced regularization approaches beside the applied weight decay and dropout could be crucial to establish deep learning for general IMS classification tasks. One way to constrain the model would be to rely only on a smaller number of peaks, which could be checked by the proposed sensitivity analysis. Currently, the *IsotopeNet* is sensitive to a large number of peaks (Section 3.3).

In future work we also aim at developing methods to account for the high biological and technical variation commonly observed in IMS data, see for example Alexandrov and Kobarg (2011) where the pixel-to-pixel variation is discussed. This may require a thorough study of the sources of technical errors like misalignment or baseline artifacts, as well as an idea of the variation induced by each patient. Furthermore, experimental study designs involving data from various institutions, patients, operators and devices, where the target values (class labels) are not aligned with design choices (e.g. the device) could be a path forward to avoid model fitting based on technical errors. At last, the application of the proposed models to other IMS data domains is necessary to better understand its strengths and weaknesses.

Acknowledgements

Tissue samples were kindly provided by Dr M. Kriegsmann (Institute of Pathology, Heidelberg University Hospital), Dr A. Warth (Thoracic Pathology, Heidelberg University Hospital), Prof. Dr H. Dienemann (Thoracic Surgery, Heidelberg University Hospital) and Prof. Dr W. Weichert (Institute of Pathology, Technical University of Munich) through the tissue bank of the National Center for Tumor Diseases (Heidelberg, Germany) in accordance with the regulations of the tissue bank and the approval of the ethics committee of Heidelberg University. The authors acknowledge that the used protocol for processing FFPE tissue, as well as the method of MS-based differentiation of tissue states is subject to patents held by or exclusively licensed by Bruker Daltonics GmbH, Bremen, Germany.

Funding

The authors gratefully acknowledge the financial support from the German Federal Ministry of Education and Research ('KMU-innovativ: Medizintechnik' program) [contract number 13GW0081] and the German Science Foundation for RTG 2224 'π^3: Parameter Identification—Analysis, Algorithms, Applications'.

Conflict of Interest: none declared.

References

Aichler, M. and Walch, A. (2015) MALDI imaging mass spectrometry: current frontiers and perspectives in pathology research and practice. *Lab. Investig.*, **95**, 422–431.

Alexandrov, T. and Kobarg, J.H. (2011) Efficient spatial segmentation of large imaging mass spectrometry datasets with spatially aware clustering. *Bioinformatics*, **27**, i230–i238.

Alexandrov, T. (2012) MALDI imaging mass spectrometry: statistical data analysis and current computational challenges. *BMC Bioinformatics*, **13**, S11.

Benninghoven, A. and Loebach, E. (1971) Tandem mass spectrometer for secondary ion studies. *Rev. Sci. Instrum.*, **42**, 49–52.

Boskamp, T. *et al.* (2016) A new classification method for MALDI imaging mass spectrometry data acquired on formalin-fixed paraffin-embedded tissue samples. *Biochim. Biophys. Acta.*, **1865**, 916–926.

Caprioli, R.M. *et al.* (1997) Molecular imaging of biological samples: localization of peptides and proteins using MALDI-TOF MS. *Anal. Chem.*, **69**, 4751–4760.

Casadonte, R. and Caprioli, R.M. (2011) Proteomic analysis of formalin-fixed paraffin-embedded tissue by MALDI imaging mass spectrometry. *Nat. Protoc.*, **6**, 1695–1709.

Casadonte, R. *et al.* (2014) Imaging mass spectrometry to discriminate breast from pancreatic cancer metastasis in formalin-fixed paraffin-embedded tissues. *Proteomics*, **14**, 956–964.

Casadonte, R. *et al.* (2017) MALDI IMS and cancer tissue microarrays. *Adv. Cancer Res.*, **134**, 173–200.

Deininger, S.O. *et al.* (2011) Normalization in MALDI-TOF imaging datasets of proteins: practical considerations. *Anal. Bioanal. Chem.*, **401**, 167–181.

Galli, M. *et al.* (2016) Machine learning approaches in MALDI-MSI: clinical applications. *Expert Rev. Proteomics*, **13**, 685–696.

Glorot, X. *et al.* (2011) Deep sparse rectifier neural networks. *Aistats*, **15**, 275.

Goodfellow, I. *et al.* (2016) *Deep Learning*. MIT Press.

Harn, Y.C. *et al.* (2015) Deconvolving molecular signatures of interactions between microbial colonies. *Bioinformatics*, **31**, i142–i150.

Hastie, T. *et al.* (2001) *The Elements of Statistical Learning*. Springer.

He, K. *et al.* (2015) Deep residual learning for image recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.

Inglese, P. *et al.* (2017) Deep learning and 3D-DESI imaging reveal the hidden metabolic heterogeneity of cancer. *Chem. Sci.*, **8**, 3500–3511.

Ioffe, S. and Szegedy, C. (2015) Batch normalization: accelerating deep network training by reducing internal covariate shift. In: *International Conference on Machine Learning*, pp. 448–456.

Kingma, D. *et al.* (2012) Adam: a method for stochastic optimization. In: *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, 2015.

Kriegsmann, J. *et al.* (2015) MALDI TOF imaging mass spectrometry in clinical pathology: a valuable tool for cancer diagnosis. *Int. J. Oncol.*, **46**, 893–906.

Kriegsmann, M. *et al.* (2016) Reliable entity subtyping in non-small cell lung cancer by matrix-assisted laser desorption/ionization imaging mass

spectrometry on formalin-fixed paraffin-embedded tissue specimens. *Mol. Cell. Proteomics*, **15**, 3081–3089.

Krizhevsky, A. *et al*. (2012) ImageNet classification with deep convolutional neural networks. *Adv. Neural Inf. Proc. Syst*., **25**.

LeCun, Y. *et al*. (1989) Backpropagation applied to handwritten zip code recognition. *Neural Comp*., **14**, 541–551.

LeCun, Y. *et al*. (2015) Deep learning. *Nature*, **521**, 436–444.

Simonyan, K. *et al*. (2013) Deep inside convolutional networks: visualising image classification models and saliency Maps. In: *Proceedings of the 2nd International Conference on Learning Representations (ICLR)*, 2014.

Srivastava, N. *et al*. (2014) Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res*., **15**, 1929–1958.

Stoeckli, M. *et al*. (2001) Imaging mass spectrometry: a new technology for the analysis of protein expression in mammalian tissues. *Nat. Med*., **7**, 493–496.

Theano Team (2016) Theano: a Python framework for fast computation of mathematical expressions. *arXiv preprint arXiv: 1605.02688*.

Thomas, S. *et al*. (2016) Dimensionality reduction of mass spectrometry imaging data using autoencoders. *IEEE Symp. Ser. Comp. Intel*., 1–7.

Yang, C. *et al*. (2009) Comparison of public peak detection algorithms for MALDI mass spectrometry data analysis. *BMC Bioinformatics*, **10**, 4.

Zeiler, M. and Fergus, R. (2013) Visualizing and understanding convolutional networks. *arXiv preprint arXiv: 1311.2901*.