Title

Peak detection and random forests classification software for gas chromatography/differential mobility spectrometry (GC/DMS) data

Authors

Yeap, Danny

McCartney, Mitchell M

Rajapakse, Maneeshin Y

et al.

Publication Date

2020-08-01

DOI

10.1016/j.chemolab.2020.104085

Peer reviewed

Published in final edited form as:
Chemometr Intell Lab Syst. 2020 August 15; 203: . doi:10.1016/j.chemolab.2020.104085.

Peak detection and random forests classification software for gas chromatography/differential mobility spectrometry (GC/DMS) data

Danny Yeap[a], **Mitchell M. McCartney**[a], **Maneeshin Y. Rajapakse**[a], **Alexander G. Fung**[a], **Nicholas J. Kenyon**[b,c,d], **Cristina E. Davis**[a,*]

[a]Department of Mechanical and Aerospace Engineering, University of California Davis, Davis, CA, 95616, USA

[b]Department of Internal Medicine, 4150 V Street, Suite 3400, University of California, Davis, Sacramento, CA, 95817, USA

[c]Center for Comparative Respiratory Biology and Medicine, University of California, Davis, CA, 95616, USA

[d]VA Northern California Health Care System, 10535 Hospital Way, Mather, CA, 95655, USA

Abstract

Gas Chromatography/Differential Mobility Spectrometry (GC/DMS) is an effective tool to discern volatile chemicals. The process of correlating GC/DMS data outputs to chemical identities requires time and effort from trained chemists due to lack of commercially available software and the lack of appropriate libraries. This paper describes the coupling of computer vision techniques to develop models for peak detection and can align chemical signatures across datasets. The result is an automatically generated peak table that provides integrated peak areas for the inputted samples. The software was tested against a simulated dataset, whereby the number of detected features highly correlated to the number of actual features ($r^2 = 0.95$). This software has also been

[*]Corresponding author. cedavis@ucdavis.edu (C.E. Davis).

CRediT authorship contribution statement

Danny Yeap: Software, Conceptualization, Methodology, Data curation, Visualization, Writing - original draft, Writing - review & editing, Formal analysis, Validation, Investigation, Resources. **Mitchell M. McCartney**: Resources, Conceptualization, Writing - original draft, Writing - review & editing, Validation, Investigation. **Maneeshin Y. Rajapakse**: Conceptualization, Methodology, Formal analysis, Writing - original draft, Writing - review & editing. **Alexander G. Fung**: Writing - original draft, Writing - review & editing, Validation. **Nicholas J. Kenyon**: Funding acquisition, Project administration, Writing - review & editing. **Cristina E. Davis**: Funding acquisition, Project administration, Writing - review & editing.

Software Information

The AnalyzeIMS (AIMS) software is available on GitHub, including a new release associated with this manuscript (version 1.4). Please refer to Professor Cristina Davis' webpage for more information. This material is available as open source for research and personal use under a PolyForm Noncommercial License 1.0.0 (https://polyformproject.org/licenses/noncommercial/1.0.0/). Commercial licensing may be available, and a license fee may be required. The Regents of the University of California own the copyrights to the software. Future published scientific manuscripts or reports using this software and/or hardware designs must cite the original publications (DOI: https://doi.org/10.1007/s12127-016-0200-9 and https://doi.org/10.1021/acs.analchem.9b01428) and this updated configuration report (DOI: https://doi.org/10.1016/j.chemolab.2020.104085).

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.chemolab.2020.104085.

developed to include random forests, a discriminant analysis technique that generates prediction models for application to unknown samples with different chemical signatures. In an example dataset described herein, the model achieves 3% classification error with 12 trees and 0% classification error with 48 trees. The number of trees can be optimized based on the computational resources available. We expect the public release of this software can provide other GC/DMS researchers with a tool for automated featured extraction and discriminant analysis capabilities.

1. Introduction

Chemical separation methods [1–3], including gas chromatography (GC), mass spectrometry (MS), and differential mobility spectrometry (DMS), are key to detecting individual chemical components in a sample mixture. The combination of DMS with GC provides orthogonal information from initial separation by polarity through a GC column prior to entering the DMS [4–6]. The oscillating electric field of a DMS can further separate ions by their charge mobility for subsequent detection and is used in many applications such as explosive [7] and drug detection [8].

GC/DMS provides a robust and reliable method of separating a variety of complex compounds [9].

The advantage with using GC/DMS compared to other instruments such as GC/MS is portability and atmospheric operability. When looking at the data output from these instruments, the common goal is peak detection. There are many different computation methods for peak detection; this paper shows a methodology of performing peak detection for GC/DMS data.

GC/DMS devices generate 3-dimensional plots with the axes that represent retention time (RT), compensation voltage (CV) and ion signal intensity. Compounds will elute through the GC portion of the GC/DMS with varying lengths of time, known as the retention time (RT) [10]. The temperature profile and column phase are examples of variables that determine the retention time of a given compound. As compounds elute from the GC column and reach the DMS, they are ionized and enter an oscillating electric field. The DMS offsets this electric field through the compensation voltage, which acts as an ion filter, dictating which ions reach the detector based on ion mobility. The CV is not held constant but alternates across a CV range on the order of 1 Hz. At a certain CV, ions will pass through the electric field and reach the detector, generating a signal with the signal strength relating to ion abundance. Each compound is represented as a 3-dimensional elliptical conical structure or peak whose volume denotes prominence of the compound. These peaks can be characterized by their CV and RT position (x and y coordinates) and volume (z coordinate).

Peak detection in GC/DMS data is necessary for chemical detection; however, to our knowledge, there is no publicly available software for GC/DMS data analysis. There are other GC/DMS analyses [11] that uses gradient, absolute thresholding, and overlapping ratio. These methods must tune numerous hyperparameters and require many validations steps to perform peak detection. The methods shown in this paper utilize a different approach by exploiting the conical nature of the GC/DMS peaks and uses less hyperparameters to perform peak detection compared to current techniques [12,13].

In a previous publication [14], we have developed a software, AnalyzeIMS, to aide GC/DMS researchers. This version of AnalyzeIMS allowed users to visualize GC/DMS spectra, perform preprocessing techniques such as baseline correction and smoothing, and perform two statistical analyses: principal components analysis (PCA) and partial least squares-discriminant analysis (PLS-DA). However, the software did not have the capability of performing peak detection and alignment. PCA and PLS-DA were performed by considering every data point of the GC/DMS spectra, rather than only incorporating data points for an entire resolved chemical peak that contained chemical information. As such, the power of this analysis was weakened.

This paper outlines a methodology using computer vision to extract chemical features from GC/DMS spectra. Computer vision has shown to useful in many applications [15–17] to differentiate objects from a background, making it favorable for GC/DMS peak detection.

This software data pipeline has three sequential stages: noise reduction, peak detection, and peak alignment. Noise reduction uses is performed using top hat filtering and thresholding. Peak detection is performed by using watershed, a computer vision technique, to discern chemical features from spectra background. Peak alignment assesses if constituent chemicals are found in other samples within a provided dataset. This upgraded software described adds peak detection in order to provide additional advanced computational methods of analyzing GC/ DMS data. All algorithms stated in the paper are now incorporated into the newest version of the software described by the results in this paper.

Further, this software version includes a machine learning technique, random forests [18], for binary classification prediction. Random forests make predictions by using an array of trained decisions tree. A decision tree is a model of the relationships of the peak volumes. Each tree is given a “vote” and the collective response denotes the classifications accuracy [19,20]. Researchers have shown that classification techniques have varying success on the same dataset, and random forests has out-performed other discriminant techniques in VOC-based data [21]. Random forests has been used in other GC/DMS studies [21–23].

2. Material and methods

2.1. Simulated data

In order to test the peak detection algorithm, simulated GC/DMS spectra were generated. Simulated data helps test the robustness of the algorithms because the true number of peaks generated can be checked against the number of peaks detected by the software. The method of generating these peaks are outlined in another paper [11]. In brief, the peaks were generated by using a 2-D gaussian distribution with normal distribution for randomly generated CV/RT locations and noise. A total of 300 GC/DMS spectra were generated where each had 10–40 peaks. The top hat radius and threshold value are chosen iteratively for one sample and the same value is applied to the rest of the samples.

2.2. Sample data

To demonstrate random forests as a binary classifier, we used a previously collected GC/DMS dataset that contained the volatile organic compound (VOC) measurements of isolated *Rhododendron* plants. *Rhododendron* are nursery ornamentals that are susceptible to root infections of *Phytophthora ramorum*, an oomycete invasive to the United States. Our group previously showed that the VOC profile of *Rhododendron* shifts upon infection with *Phytophthora ramorum.* [24] While the exact method used to capture VOC measurements of these plants is described elsewhere [4], briefly, 12 *Rhododendron* plants were grown inside a growth chamber. Six plants were mock inoculated and served as healthy controls and six had been inoculated with *P. ramorum.* Infection were confirmed *ex post*. An enclosure was placed around each plant, which allowed for *Rhododendron* VOC emissions to concentrate inside, isolating each plant's volatile profile from the growth chamber background. A sample pump was used to sip air from each enclosure and pass it through a preconcentrator trap packed with sorbent to capture the VOC profile, and the trap was thermally desorbed into the GC/DMS system for chemical separation and detection.

2.3. Data analysis

2.3.1. Software structure—A typical GC/DMS plot is 3-dimensional with axes representing compensation voltage (x-axis), retention time (y-axis), and chemical intensity (z-axis) (Fig. 1A). High intensity features localized at −20 V compensation voltage represent the reactant ion peak (RIP), an expected feature of DMS [25]. Peaks representing compounds detected in this sample can be seen scattered between CV of −15 and 15 V.

To extract chemical features, the following processes are used: baseline correction, smoothing, gray-scaling, RIP removal, top-hat filtering, thresholding, and watershed. Baseline correction and smoothing are both used to correct instrument drift and large fluctuations between data points. Gray-scaling is used to normalize all samples in the dataset. Then RIP is removed to emphasize the compound peaks in the dataset. Top-hat filtering is to reduce the background noise while retaining compound peaks. Thresholding is used to further reduce background noise and convert the gray-scale image to binary image. Watershed helps label the distinct peaks and corresponding pixels of the compound peaks.

Baseline correction of the data is performed by asymmetric least squares and smoothing is performed by Savitzky-Golay filtering as incorporated in earlier versions of AIMS. The raw data image (Fig. 1A) and corrected and smoothed image (Fig. S1) show a large reduction of noise. Once the data is corrected and smoothed, the intensities range from 0 to 0.17 V. Converting to a grayscale image remaps the values from 0–0.17 V to 0–1 V, thus normalizing the image set. The result not only increases the compound peak intensities but also the background noise intensities (Fig. 2A). The advantages with grayscale images are important because it provides a method to not only emphasize the background noise which appeared not present (Fig. S1) but also normalizes the intensity values across all samples. The computer vision algorithm will also improve feature detection because the signal strength is increased.

RIP intensities can be significantly higher than that of chemical compound peaks, which makes separating the compound peaks from the noise when thresholding difficult. The automated algorithms may retain the RIP information and remove the compound peaks, an undesirable effect. To avoid these difficulties, the RIP was cropped from the dataset (Fig. 2A) by only including × axis values between −18 and 15 V.

Then top hat filtering [26] is applied to filter background noise and preserve peak information in the data (Fig. 2B). This approach performs well in settings where the relevant and irrelevant objects are different shapes, which is true for the differences in detector noise and chemical peaks in GC/DMS spectra. To further remove background noise, thresholding is used to convert the from grayscale to binary images in black and white (Fig. 2C). This technique is advantageous because compound peaks are more prominent than background noise. Finally, watershed segmentation is performed on the images to index individual peaks and separate compounds that are overlapping one another (Fig. 2D). Watershed segmentation works well on objects of circular nature and performs separation using watershed lines found through the shape of the peaks.

Once the peaks have been detected, a peak alignment algorithm generates a peak table. Peak alignment is necessary to ensure that the same compound peak in one sample refers to the same compound peak in another sample. A peak table is tabular representation of individual compound found in a sample. Each cell in the row and column represent the max intensity, the columns represent a unique compound, the rows represent the CV and RT location of the max intensity of the peak detected. Peak alignment involves three stages: setting CV and RT boundaries, determining coordinates for all detected chemical features and computing the volume of all peaks detected. The algorithm determines whether chemical features in samples within the dataset are the same chemical. For instance, if chemical feature *n* is found in one sample, the algorithm looks for chemical features in other GC/DMS spectra that have the same coordinate, plus/minus the provided CV and RT boundaries.

At this point, the algorithm has performed all functions on the dataset and has summarized chemical information into a peak table. This table can then be exported for the user to perform statistical analysis and data interpretation. Should the user seek discriminant analysis on their dataset, a random forests option is now included in the software. Model generation via random forests [18] uses peak table data to find relationship between samples based on inputted discriminate categories. These models are the prediction mechanism used for binary classification of each sample.

2.3.2. AIMS—AnalyzeIMS (AIMS) v1.4 for Matlab 2017a was utilized to visualize and analyze GC/DMS data [4]. In this work, we describe how we updated AIMS to incorporate top hat image filtering, watershed peak detection, peak alignment, and random forests for automated peak extraction and model generation from GC/DMS plots. The updated version of AIMS allows the user to select values such as filter size and threshold values relevant to the dataset to detect peaks. The filter size and threshold values are chosen such that the maximum noise is reduced while retaining the compound peaks. The software has visual plots that can be used to help tune these parameters. These parameters are key components in top hat filtering and thresholding. The software is built to allow users to inspect how images are transformed as these parameters are tuned. Random forests parameters such as the number of trees and number of columns to select are also tunable by the users to find the optimal classification accuracy for different datasets.

3. Theory

3.1. Top hat

Top hat filtering is a method of removing given shapes while maintaining the integrity of the image. An example use case of top hat filtering is to remove specific objects, making non-uninform lighting constant. It can be used on a binary and gray-scale image, but the underlying purpose is to enhance an image. This algorithm is useful when objects in an image are similar in shape but differ in size. This occurs in GC/DMS plots where the compound peaks and noise peaks are circular-like objects and differ in significantly in size. Top hat filtering uses a structuring element, any user chosen shape, to perform its computation. At a high level, the algorithm filters out all shapes that are smaller than the structuring element. There are two stages top hat filtering: opening an image and subtracting the opened image from the original image. The opened image is computed by performing image erosion and dilation. In a gray scale image, image erosion is computed by iteratively passing through each pixel in the original image and setting the pixel value to the minimum intensity within the pixel’s neighbors. The pixel’s neighbors are defined by the structuring element selected. This effectively reduces the intensity gradient across all pixels in the image. Only applying erosion to an image tends to be destructive to an image thus it is important to dilate the image.

The computation of image dilation is similar to that of image erosion. Image dilation iteratively passes through each pixel in the eroded image and sets the pixel value to the maximum intensity within the pixel’s neighbors. Similarly, the pixel’s neighbors are defined by the same structuring element selected in image erosion. This process essentially restores the intensities that were lost in the eroded image. In general, opening an image, helps removes small objects in the image because erosion will remove objects smaller than the structuring element. The dilation portion restores the remaining objects which leads to less image destruction.

The last computation step is to subtract the opened image from the original image. The subtraction reduces all pixel intensities of positive pixel values in the opened image. Higher intensity value in the opened image results in larger intensity reduction in the original image. In GC/DMS data, this process effectively removes the smaller background noise and retains the large circular compound peaks in the image.

The structuring element selected for this study is a circle. GC/DMS peaks tend to be elliptical and circular, however it is more advantageous to use a circle to filter GC/DMS data. A circle can be used to filter both ellipses and circles as long as the radius is the greater than or equal to the major axis of the ellipse. Applying top fat filtering to the gray-scale image (Fig. 2A), it can be seen that the compound ions are retained, and noise background reduced (Fig. 2B). As a result, it is advantageous to filter GC/DMS data using top hat filtering.

3.2. Thresholding

After top hat filtering, thresholding is applied to the image to remove additional background noise. Thresholding converts a grayscale image to a binary image by selecting a user defined threshold. This is computed by setting all pixel values below the threshold to 0 and all pixel values above threshold to 1 in an attempt to completely separate the chemical features from the background. Equation (1) denotes the equation for thresholding.

$$I(x, y) = \begin{cases} 0, & if\ I(x, y) < thershold \\ 1, & otherwise \end{cases} \tag{1}$$

I denotes the new intensity value assigned to each pixel with coordinates (*x, y*) in a GC/DMS spectrum for the inputted *threshold*. The threshold value is determined by the user. This technique was found to be useful because the noise intensities tend to be significantly lower than peak intensities.

3.3. Watershed

Watershed segmentation is a computer vision technique that can index compound peaks. It uses a binary image as an input and outputs the pixels that belong to each respective chemical feature. The advantage is that watershed segmentation can separate overlapping objects, which often occurs with compounds that are neither fully resolved by the GC column and/or the DMS electric field. At times, compound peaks may overlap making it difficult to differentiate between unique peaks. The overlap of peaks is caused by two chemical species arriving to the detector at nearly the same time. This might be because the GC column failed to resolve the two species and their ion mobilities were too similar for the DMS to separate. Watershed is computed with two steps: creating a distance transform and label compound peaks separated by watershed lines.

The distance transform is computed by taking the binary image and changing all the pixel values to represent the Euclidean distance to the nearest background pixel. As a result, the background pixels are all computed to 0 and the compound peak pixels are all positive Euclidean distance values. The largest distance can be found at the center of the compound peak because these pixels are the farthest from the background. The smallest distance will be the outer edge of the compound peak because these pixels are closest to the background pixel.

The labelling of compound peaks is performed by finding all the local maximum in the image. As previously mentioned in the distance transform, larger values are found at the center of compound peaks which become the local maximum. Each maximum is defined as a separate peak and each maximum is changed to reflect a unique peak. As a result, the maximum pixel value is changed to a unique peak number. For the algorithm to determine which peak the remaining pixels belong, the algorithm finds the nearest maximum and change its value to the respective compound maximum. This is the key to separating overlapping peaks. Overlapping peaks are pinched at a particular point and the distance transform is able to figure out which peak the pixel belongs to. With all compound peaks identified, the peak alignment algorithm is used to generate a peak table for random forest. The specifics of the algorithm are described in a further section.

3.4. Random forests

Using the peak table, random forests can build a machine learning model for binary classification. In random forests, models are represented as a group of decision trees. A decision tree contains nodes and edges, where the nodes represents a feature or in this case peak volume in the dataset and the edges represent the two possible decisions to traverse down the tree. A decision tree is similar to a binary tree, a common computer science data structure, in which each node can have two edges. As the algorithm traverses down the decision tree, the determine whether or not the peak volume at that node is less than the threshold volume. The threshold value is determined by iteratively going through all bisection points for a particular peak volume. The bisection point with the highest entropy is used as the threshold volume. The leaf nodes of a binary tree represent the classification of the dataset.

In order to build the model, random forests requires two parameters to be defined: the number of peaks, b, to randomly select from the peak table and the number of decision trees, m, to use to build the model. The first parameter, b, controls how similar the trees will be. Thus, a larger b is prone to overfitting because it increases the likelihood the nodes will be split across the same peak. A smaller b is prone to excess randomness due to inability to capture peak correlations. The second parameter, m, denotes the number of trees the random forests will contain. As m increases, the number of decision trees increases as does the computational time to perform training and testing. Thus, it is important to select an appropriate number of trees to prevent long training times.

4. Results and discussion

4.1. Top hat filtering

Top hat filtering was applied to the preprocessed image (Fig. 2A). The outputs (Fig. 3A–C) show the effectiveness of reducing the background noise. The process of selecting an appropriate radius starts by selecting a small radius such as radius of 2 (Fig. 3A). Comparing the input (Fig. 2A) and the output (Fig. 3A) shows background detector noise is still present which can complicate the selection of an appropriate threshold value in the next step. Slowly increasing the radius of the can filter out more noise (Fig. 3B). At some point, the radius is large enough to remove nearly all detector noise, but at the expense of losing nearly all chemical features (Fig. 3C). The software increases the radius until the compound peaks disappear (Fig. 3C) then it steps back one radius and uses this value to filter out the background noise. The software is also built for users to tune a different radius. If top hat filtering is not included in the filtering process, then thresholding will cut off the outer edges of the compound peak. This is undesirable because the volumes of these peaks are already extremely low ~10^{-3}. Further clipping of the outer edges prevents random forest from generating robust models.

4.2. Thresholding

Thresholding involves taking the input (Fig. 3B), the image resulting from top hat filtering, and generating an output (Fig. 4A–C). Similar to finding top hat filter’s circle radius, the optimal threshold value is found using an iterative process. A small initial threshold, such as

0.2 V, retains more pixels in the outputted image, but also includes detector noise (Fig. 4A). This is undesirable because model generation may become difficult with extra noise detected. It can also lead to over-segmentation when watershed is applied to the binary image. High threshold values will result in loss of compound peak information (Fig. 4C). To automate this process, the user must select a patch of noise background in the image. The max intensity of the background is used as the threshold value. This is done by computing the max intensity across all samples. This method outputs a binary image (Fig. 4B) such that compound peaks is retained and noise background is reduced.

The resulting image after thresholding (Fig. 4B) contains silhouettes of detected chemical features as presented in a binary image. The white regions are the pixels that belong to one peak and the black regions represent the background (Fig. 4B). At this point, the image contains peaks that are not fully resolved by either the GC column or the DMS electric field, which is addressed during watershed. Without thresholding, watershed would pick up extraneous noise peaks. This is called over-segmentation. For example, in a GC/DMS sample containing 200 actual compound peaks, watershed could generate 3000 detected peaks if thresholding was not applied. A majority of these peaks belonged to the background. This is due to the grayscale image generated from top hat filtering. As a result, it is important to generate a binary image to ensure that watershed does not pick up the noise peaks.

4.3. Watershed

As watershed indexes chemical features, it also separates overlapping peaks, determining which pixels belong to which respective peak. The input (Fig. 5A) to watershed is a binary image that resulted from thresholding. To observe watershed’s effect, a magnified image (Fig. 5B) of three peaks are shown: one distinct peak and two overlapping. The distance transform used in watershed can find a bisection point or watershed lines to separate unresolved peaks. Watershed segmentation not only successfully separated the peaks (Fig. 5C) but also labels all pixels that belong to a unique peak.

4.4. Peak detection

4.4.1. Simulated data—

To test how well the sequential use of these algorithms perform, 300 GC/DMS spectra with 10–40 peaks were mathematically generated. Each spectrum was individually inputted into the software. With a true number of chemical features known for each, we compared this to the number of peaks detected by the software. The results are shown in Fig. 6. A correlation coefficient of r = 0.95 was computed on the fitted line. This signifies that algorithm is at tracking peaks 95% of the time.

Generally, our peak detection algorithm tends to underestimate the number of peaks. The fitted line shows that the algorithm tends to over detect features in datasets with 15 or less generated peaks, while underestimating in spectra with more than 15 peaks. Underestimation is likely due to the difficulty of detecting low signal peaks that are not resolved from the background noise. This has been seen in a different type of peak detection algorithm [11]. Further, peaks that have strong slope gradient but small radius may be lost during top hat filtering, as larger radii inadvertently removes thinner chemical features.

Our software was designed to give users flexibility to match their preference to approach chemical feature detection. For instance, users with datasets containing measurements of high concentration volatiles might only be interested in corresponding high intensity chemical features. Low concentration chemicals might be present in their samples, but to the researcher are considered contaminants and are not of their interest. Thus, they might prefer that the software underestimate the number of actual chemical features, as they are interested in filtering out chemical noise in their samples. In another example, a researcher might have a dataset in which they need to perform untargeted chemical analysis, attempting to capture as many chemical features as possible. This user could input less strict values into the algorithm, hoping to capture as many real chemical features as possible, at the expensive of possibly including detector noise into their resulting peak table.

4.4.2. Peak alignment—The alignment of peaks is crucial for comparison across samples, as it determines whether which samples contain the same chemical features. For this software, alignment is a factor of the retention time and compensation voltage, which is determined as the x and y coordinate of the maximum intensity of a peak. In general, a peak should not shift a significant amount across the compensation voltage and retention time. This is associated with inherent drift of the GC/DMS instrument. The user must first determine proper compensation voltage and retention time bounds. Ideally, the GC/DMS user would separately quantify the expected RT and CV drift of their instrument. This could occur through repeated measurements of the same chemical compound or compounds, and the user can determine the expected range of RT and CV drift for a given dataset. As a result, the user must perform additional repeated measurements on one sample to observe how large these shifts are. The algorithm relies on this knowledge to properly align peaks.

With established bounds, the algorithm selects the CV and RT of the max intensity of each peak in a sample to populate into the peak table. For simplicity, Samples A and B (Fig. 7) will be used to explain how the peak table's columns are generated. The algorithm will start with Sample A and randomly select the peak marked with a green circle. The algorithm will then use the CV and RT of that peak's maximum intensity and store it in the column of the peak table. The CV and RT bounds defined for the chosen peak is used to search for the same peak in sample B. A peak within bounds is found in sample B, circled in green (Fig. 7), and considered the same compound as that populated for sample A. Thus, another column will not be created for this peak circled in green in sample B. The peaks circled in green are removed from the dataset and cannot be reselected to create another column in the peak table. The process continues until all peaks in sample A are added to the column. Once all peaks are added the algorithm moves onto the next sample and performs the same operation until all samples have gone through this process. Since the peak circled in red in Sample B will not have been added to the peak table it will be added when the algorithm is run on this sample.

With all columns created, the last step is to integrate and store the peak volumes for each chemical feature in each sample. The algorithm uses the first peak's CV and RT and stores all peak volumes that are within the bound of the user-chosen CV and RT. If a peak is not present in the sample, a 0.00 is stored. This is done for every peak and sample until a complete peak table is generated (Table 1). The unit for peak volume is technically voltage squared times seconds; however, this unit is not intuitive because it suggests that peak volume has a time component. Thus, the standard unit for peak volume in this paper is arbitrary unit (a.u.) – it is subjective to interpretation. In this case, it is easier to visualize peak volume as units cubed. Appended to the end of the peak table is a row for CV and RT (Table 1). These rows represent the mean and standard deviation of all max CV and RT found across the samples. Both mean and standard deviation represent how much the peaks are shifting and can be used as a means to understand the dataset.

The advantage with this technique compared with another peak alignment technique [11] is the number of manual user inputs is comparably less. Our peak alignment requires only the CV and RT bounds while the other technique [11] requires CV and RT bounds, reference sample, mathematical dot products, etc. More inputs lead to precise adjustments for reliable performance. These additional inputs are not required herein because the computer vision algorithms previously used to identify peaks reduce the complexity of the GC/DMS datasets by removing noise and separating overlapping peaks.

To test the peak algorithm, simulated data was randomly shifted to compute the consistency of the peak alignment algorithm. Majority of random shifting was performed within 1 step of the CV and 10 steps of the RT, a typical range for GC/DMS plots. The peak alignment algorithm showed a 94% rate of correctly aligning peaks showing that peak alignment is reliable and effective even though the peaks are shifting. The remaining peaks not properly aligned because of the random shifting. The random shifting causes peaks to move out of range or drift to another compound peak. Moving out of range prevents the algorithm from detecting the compound and drifting to another compound peak causes the algorithm to choose the closest peak for alignment.

4.5. Random forests

To show an application of the methodology, a *Rhododendron* sample data is used. This dataset contains 24 GC/DMS samples, with 12 infected by *P. ramorum* and 12 healthy controls. Peak detection was performed using the following values: 12 for top hat, 0.43 for threshold. The peak alignment algorithm found a total of 200 number of chemical features and were stored in a peak table. To train a random forest mode, the number of features must be chosen to maximize the accuracy of the model. An example of how number of features affects accuracy (Fig. 8) shows that a small *b*, number of features, value does not perform well. The misclassification error converges to approximately 7% which is high. Small values of *b* prevent random forests from finding peak relationships between the samples because the probability of selecting the appropriate peaks for compound prediction is low. As a result, the majority of decision trees become poor prediction models lowering the overall accuracy. Choosing medium value of 16 peaks shows the best results for this dataset because the accuracy plot is consistently below the low and high *b* values. A reasonable value tends to be around square root of the total number of peaks detected. Higher values overfit the model because it increases the similarities between the trees in the random forest. Consequently, the lack of generalization does not allow the model to accurately provide predictions. Accuracies start to drop and eventually do not perform well. The software is designed to allow the user to tune these this value. The number of trees to be generated is another parameter can affect the performance of random forest. A small number of trees in random forest generally performs worse than large number of trees. Each tree is trained differently which allows for generalization. With small amounts of tree, we lose generality and the model has difficulty finding relationships between the classification data and the peaks. However, as the number of decision trees increases as does the computational time to perform training and testing. Fig. 8 shows *m* 1–60 trees used for model generation. It can be seen that as the number of trees increase the average classification error decreases. This is expected because less decision trees translates to less chance that the model will fully capture the behavior of the GC/DMS datasets. It may be tempting to select an extremely number of grown trees; however, doing so will dramatically increases the computation time to train random forests. It is advantageous to start with less trees and slowly increase while observing the accuracy. It can be seen that random forest is a good classifier for performing binary classification on rhododendron plants. To validate this entire process, a 10-fold cross validation was used on the 24 *Rhododendron* plants. One third of the samples were set aside for as the test set and the last two thirds were used to build the model. An average accuracy was computed to be 94%. This suggests the process of using top hat filtering, thresholding, and random forest is robust in predicting unknown *Rhododendron* samples. Previously [4], PLS-DA was used to confirm the VOC profiles of infected and healthy *Rhododendron* plants show a difference as well. Using random forest further confirms what is found in that software can be used to build models to separate the VOC profiles of *Rhododendron*.

5. Conclusion

This paper uses techniques from computer vision and machine learning to show how they can be used for peak detection and to perform random forests binary classification. Grayscale mapping, removal of RIP, top hat filtering and thresholding were used to remove noise in the background and convert the image to a binary image. Watershed segmentation helped detect and label each compound peak ion as a different compound. A peak alignment algorithm using compensation voltage and retention time bounds generated a peak table that summarized the compounds found. Random forests, a machine learning model, showed high accuracy showing that random forests is a robust model for predicting binary classification on GC/DMS samples. Future work on this study can compare different machine learning algorithms to evaluate which is better on portable devices or hardware with limited memory and processing power. Adding multiclass classifications could also produce models with the capability to detect compounds in more complex mixtures.

Supplementary Material

Refer to Web version on PubMed Central for supplementary material.

Acknowledgements

This study was partially supported by: NIH award U01 EB0220003-01 [CED, NJK]; the NIH National Center for Advancing Translational Sciences (NCATS) through grant UL1 TR000002 [CED, NJK]; NIH award 1P30ES023513-01A1 [CED, NJK]; NIH award UG3-OD023365 [CED, NJK]; USDA award Animal and Plant Health Inspection Service (APHIS) Plant Protection and Quarantine (PPQ) program grant #15-8130-0563-CA [CED]; CITRIS and the Banatao Institute award 2019-0092 [CED, NJK]; NYSEARCH award M2019-004-0 [CED]. The authors once again thank members of Richard Bostock's laboratory, especially Tatiana Roubtsova, for making their *Rhododendron* plants available for volatile analysis. The contents of this manuscript are solely the responsibility of the authors and do not necessarily represent the official views of the funding agencies.

References

[1]. Sasser M, Identification of Bacteria by Gas Chromatography of Cellular Fatty Acids, 1990.

[2]. Wang H, Liu J, Cooks RG, Ouyang Z, Paper spray for direct analysis of complex mixtures using mass spectrometry, Angew. Chem. Int. Ed 49 (5) (2010) 877–880.

[3]. Makinen MA, Anttalainen OA, Sillanpää ME, Ion Mobility Spectrometry and its Applications in Detection of Chemical Warfare Agents, ACS Publications, 2010.

[4]. Anishchenko IM, McCartney MM, Fung AG, Peirano DJ, Schirle MJ, Kenyon NJ, Davis CE, Modular and reconfigurable gas chromatography/differential mobility spectrometry (GC/DMS) package for detection of volatile organic compounds (VOCs), Int. J. Ion Mobil. Spectrom 21 (4) (2018) 125–136. [PubMed: 31086501]

[5]. McCartney MM, Spitulski SL, Pasamontes A, Peirano DJ, Schirle MJ, Cumeras R, Simmons JD, Ware JL, Brown JF, Poh AJ, Coupling a branch enclosure with differential mobility spectrometry to isolate and measure plant volatiles in contained greenhouse settings, Talanta 146 (2016) 148–154. [PubMed: 26695246]

[6]. Davis CE, Bogan MJ, Sankaran S, Molina MA, Loyola BR, Zhao W, Benner WH, Schivo M, Farquar GR, Kenyon NJ, Analysis of volatile and non-volatile biomarkers in human breath using differential mobility spectrometry (DMS), IEEE Sensor. J. 10 (1) (2009) 114–122.

[7]. Eiceman G, Krylov E, Krylova N, Nazarov E, Miller R, Separation of ions from explosives in differential mobility spectrometry by vapor-modified drift gas, Anal. Chem 76 (17) (2004) 4937–4944. [PubMed: 15373426]

[8]. O'Donnell RM, Sun X, P.d.B. Harrington, Pharmaceutical applications of ion mobility spectrometry, Trac. Trends Anal. Chem 27 (1) (2008) 44–53.

[9]. Lu Y, Harrington PB, Forensic application of gas chromatography–differential mobility spectrometry with two-way classification of ignitable liquids from fire debris, Anal. Chem 79 (17) (2007) 6752–6759. [PubMed: 17683164]

[10]. Krebs MD, Tingley RD, Zeskind JE, Holmboe ME, Kang J-M, Davis CE, Alignment of gas chromatography–mass spectrometry data by landmark selection from complex chemical mixtures, Chemometr. Intell. Lab. Syst 81 (1) (2006) 74–81.

[11]. Fong SS, Rearden P, Kanchagar C, Sassetti C, Trevejo J, Brereton RG, Automated peak detection and matching algorithm for gas Chromatography–differential mobility spectrometry, Anal. Chem 83 (5) (2011) 1537–1546. [PubMed: 21204557]

[12]. Jarman KH, Daly DS, Anderson KK, Wahl KL, A new approach to automated peak detection, Chemometr. Intell. Lab. Syst 69 (1–2) (2003) 61–76.

[13]. D'Addario M, Kopczynski D, Baumbach JI, Rahmann S, A modular computational framework for automated peak extraction from ion mobility spectra, BMC Bioinf. 15 (1) (2014) 25.

[14]. Peirano DJ, Pasamontes A, Davis CE, Supervised semi-automated data analysis software for gas chromatography/differential mobility spectrometry (GC/DMS) metabolomics applications, Int. J. Ion Mobil. Spectrom 19 (2–3) (2016) 155–166. [PubMed: 27799845]

[15]. Cheng M-M, Mitra NJ, Huang X, Torr PH, Hu S-M, Salient Object Detection and Segmentation, 2011.

[16]. Caselles V, Kimmel R, Sapiro G, Sbert C, Minimal surfaces based object segmentation, IEEE Trans. Pattern Anal. Mach. Intell 19 (4) (1997) 394–398.

[17]. Yeap D, Hichwa PT, Rajapakse MY, Peirano DJ, McCartney MM, Kenyon NJ, Davis CE, Machine vision methods, natural language processing, and machine learning algorithms for automated dispersion plot analysis and chemical identification from complex mixtures, Anal. Chem 91 (16) (2019) 10509–10517. [PubMed: 31310101]

[18]. Breiman L, Random forests, Mach. Learn 45 (1) (2001) 5–32.

[19]. Pal M, Random forest classifier for remote sensing classification, Int. J. Rem. Sens 26 (1) (2005) 217–222.

[20]. Lindner C, Bromiley PA, Ionita MC, Cootes TF, Robust and accurate shape model matching using random forest regression-voting, IEEE Trans. Pattern Anal. Mach. Intell 37 (9) (2014) 1862–1874.

[21]. Purcaro G, Rees CA, Wieland-Alter WF, Schneider MJ, Wang X, Stefanuto P-H, Wright PF, Enelow RI, Hill JE, Volatile fingerprinting of human respiratory viruses from cell culture, J. Breath Res 12 (2) (2018), 026015. [PubMed: 29199638]

[22]. Van Gaal N, Lakenman R, Covington J, Savage R, De Groot E, Bomers M, Benninga M, Mulder C, De Boer N, De Meij T, Faecal volatile organic compounds analysis using field asymmetric ion mobility spectrometry: non-invasive diagnostics in paediatric inflammatory bowel disease, J. Breath Res 12 (1) (2017), 016006. [PubMed: 28439048]

[23]. Coy SL, Krylov EV, Schneider BB, Covey TR, Brenner DJ, Tyburski JB, Patterson AD, Krausz KW, Fornace AJ, Nazarov EG, Detection of radiation-exposure biomarkers by differential mobility prefiltered mass spectrometry (DMS–MS), Int. J. Mass Spectrom 291 (3) (2010) 108–117. [PubMed: 20305793]

[24]. McCartney MM, Roubtsova TV, Yamaguchi MS, Kasuga T, Ebeler SE, Davis CE, Bostock RM, Effects of Phytophthora ramorum on volatile organic compound emissions of Rhododendron using gas chromatography–mass spectrometry, Anal. Bioanal. Chem 410 (5) (2018) 1475–1487. [PubMed: 29247382]

[25]. Kendler S, Lambertus GR, Dunietz BD, Coy SL, Nazarov EG, Miller RA, Sacks RD, Fragmentation pathways and mechanisms of aromatic compounds in atmospheric pressure studied by GC–DMS and DMS–MS, Int. J. Mass Spectrom 263 (2–3) (2007) 137–147.

[26]. Zeng M, Li J, Peng Z, The design of top-hat morphological filter and application to infrared target detection, Infrared Phys. Technol 48 (1) (2006) 67–76.

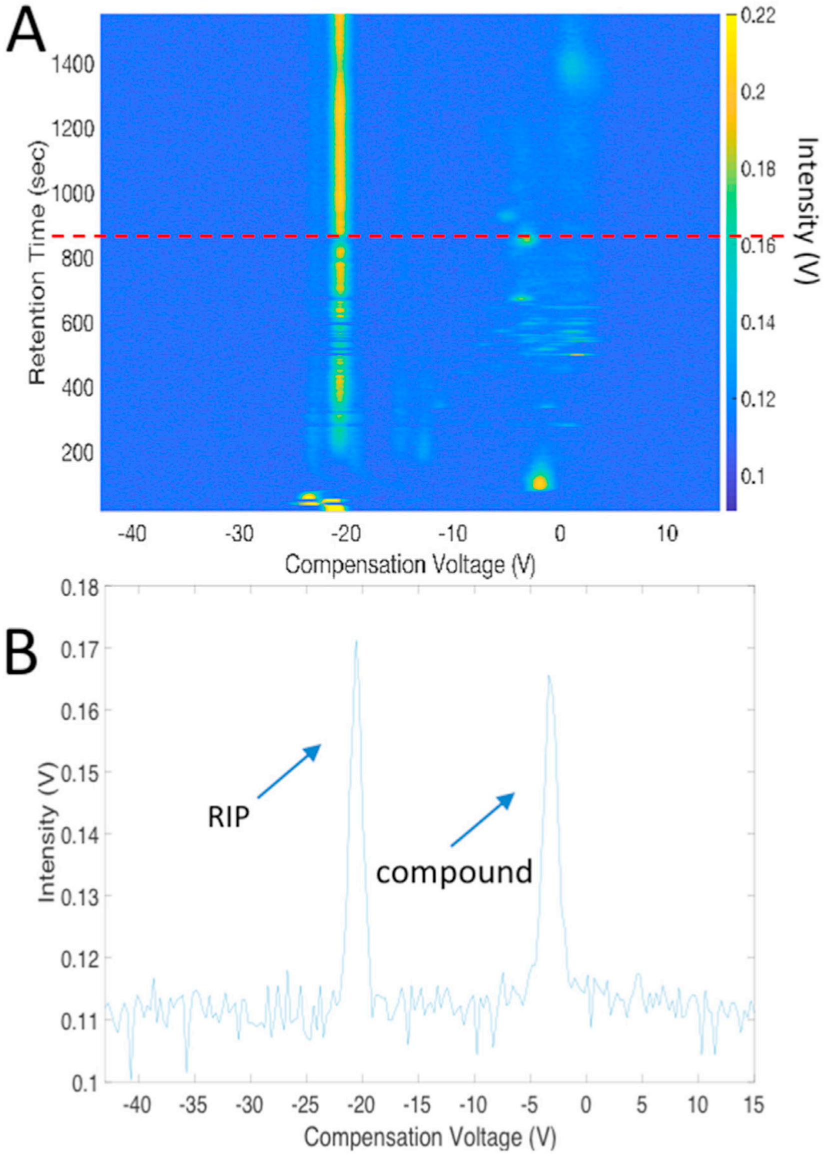

Fig. 1.

Original GC/DMS plot: (A) Top down (intensity) and (B) cross section of GC/DMS plot along dotted red line (RT = 840 s). (For interpretation of the references to colour in this figure legend, the reader is referred to the Web version of this article.)

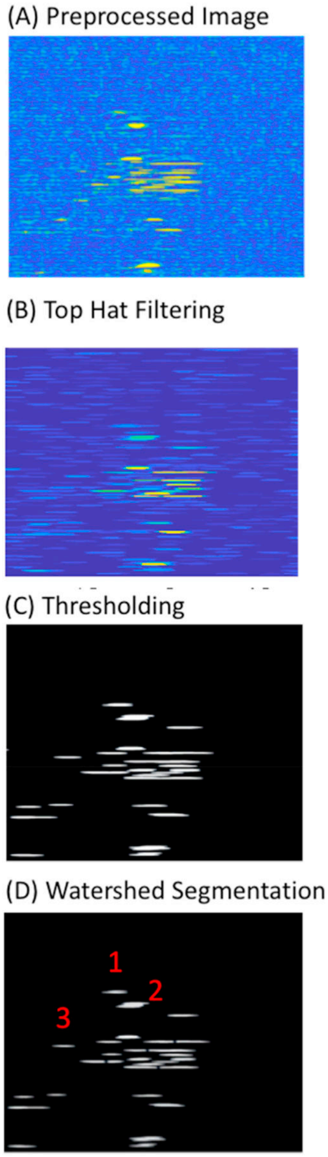

Fig. 2.

Diagram of the computer vision algorithms used to extrapolate peaks. (A) Cropped image that has RIP removed. (B) Top hat filtered applied to reduce noise and retain compound peaks. (c) Thresholding applied to convert to binary image (d) Watershed applied to label all pixels belong to each respective compound peak.

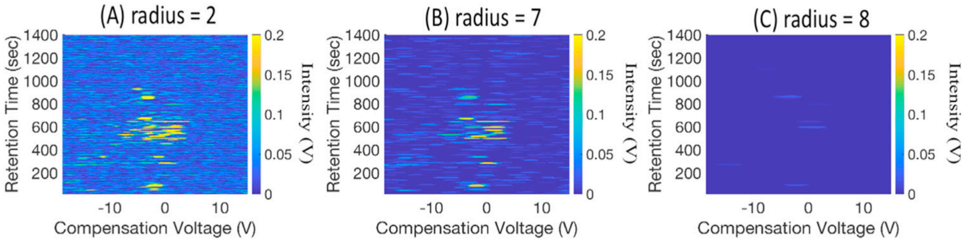

Fig. 3.

Top hat filtering applied to GC/DMS plot with different circle radius.

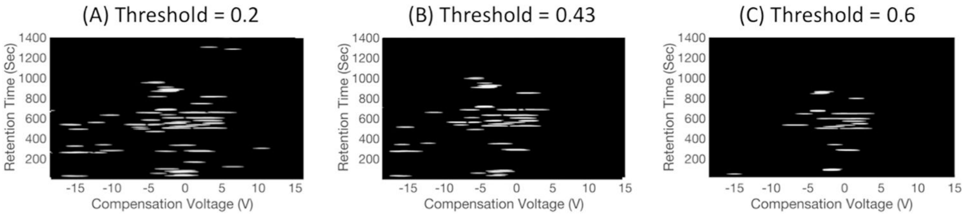

Fig. 4.

Thresholding applied to GC/DMS plot with different threshold values.

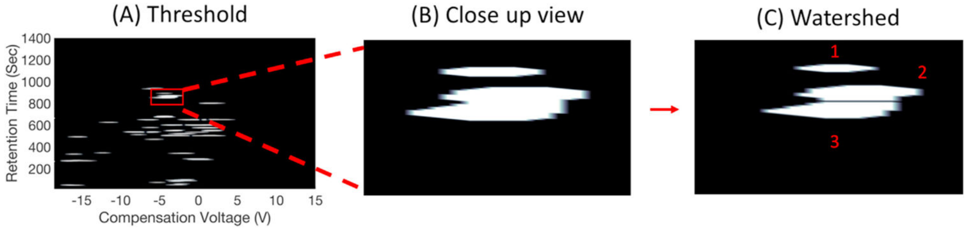

Fig. 5.

Watershed applied to GC/DMS plot.

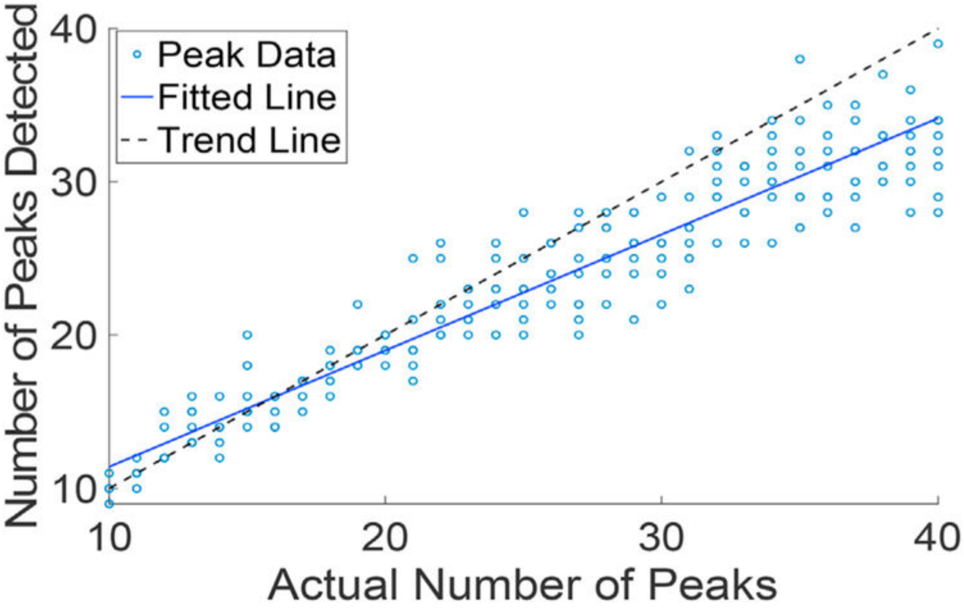

Fig. 6.

Scatter plot of number of actual peaks vs. number of peaks detected.

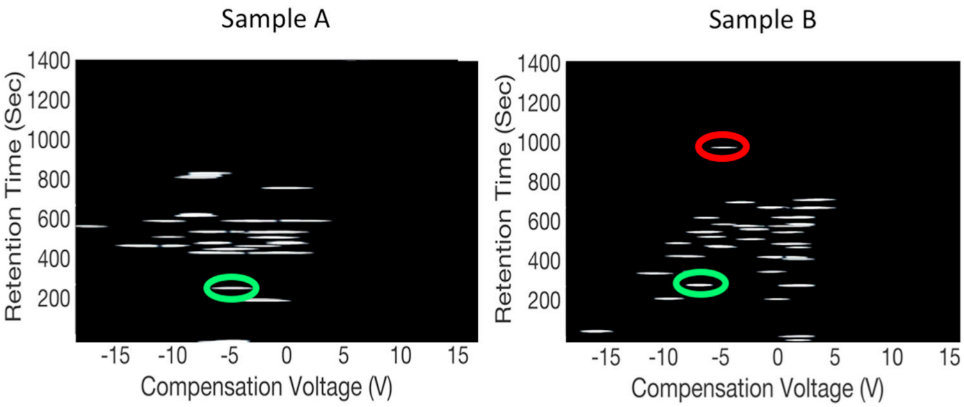

Fig. 7.

Green peak are the same compound and red peak is unique compound. (For interpretation of the references to colour in this figure legend, the reader is referred to the Web version of this article.)

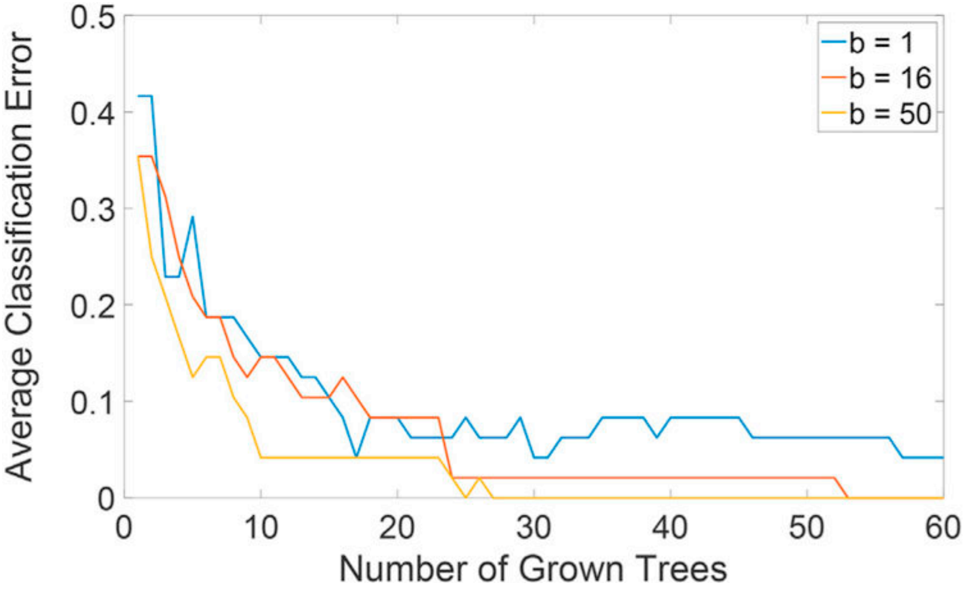

Fig. 8.

Random forest accuracy with different *b* and *m* values.

Table 1

Example of a generated peak table. Cells represent the volume under the peak for chemical feature *m* present in each sample *n* in arbitrary units (a.u.). In addition, the average ± standard deviation for the maximum intensity of each chemical feature's CV (x coordinate) and RT (y coordinate) are provided. Cells with a peak volume of 0.00 a.u. indicate that the sample was determined to not contain the corresponding chemical.

	Chemical feature 1	**Chemical feature 2**	**Chemical feature *m***
Sample 1	1.60 a.u.	0.00 a.u.	0.00 a.u.
Sample 2	0.00 a.u.	0.00 a.u.	0.00 a.u.
Sample *n*	0.74 a.u.	0.19 a.u.	0.16 a.u.
CV	23.43 ± 0.13 V	11.25 ± 0.20 V	3.14 ± 0.13 V
RT	37.93 ± 2.69 s	340.76 ± 2.21 s	645.58 ± 1.04 s