Journal of Bioinformatics and Computational Biology

PEAK DETECTION IN MASS SPECTROMETRY BY GABOR FILTERS AND ENVELOPE ANALYSIS

Nha Nguyen [1,2], Heng Huang [1,*], Soontorn Oraintara [2], An Vo [3]

[1] *Department of Computer Science and Engineering,* [2] *Department of Electrical Engineering, University of Texas at Arlington, TX, USA.*

[3] *The Feinstein Institute for Medical Research at North Shore LIJ, New York, USA.*

Email: nha.nguyen@mavs.uta.edu, heng@uta.edu, oraintar@uta.edu, anphuocnhu.vo@mavs.uta.edu.

Received (Day Month Year)
Revised (Day Month Year)
Accepted (Day Month Year)

Mass Spectrometry (MS) is increasingly being used to discover diseases related proteomic patterns. The peak detection step is one of most important steps in the typical analysis of MS data. Recently, many new algorithms have been proposed to increase true position rate with low false discovery rate in peak detection. Most of them follow two approaches: one is the denoising approach and the other one is the decomposing approach. In the previous studies, the decomposition of MS data method shows more potential than the first one. In this paper, we propose two novel methods named GaborLocal and GaborEnvelop, both of which can detect more true peaks with a lower false discovery rate than previous methods. We employ the Gaussian local maxima to detect peaks, because it is robust to noise in signals. A new approach, peak rank, is defined at the first time to identify peaks instead of using the signal-to-noise ratio. Meantime the Gabor filter is used to amplify important information and compress noise in the raw MS signal. Moreover, we also propose the envelope analysis to improve the quantification of peaks and remove more false peaks. The proposed methods have been performed on the real SELDI-TOF spectrum with known polypeptide positions. The experimental results demonstrate our methods outperform other common used methods in the Receiver Operating Characteristic (ROC) curve.

Keywords: Mass Spectrometry; Peak Detection; Peak Rank; Gabor Filter; Gaussian Local Maxima; Envelope Analysis.

1. INTRODUCTION

Mass Spectrometry (MS) is an analytical technique that has been widely used to discover diseases related proteomic patterns. From these proteomic patterns, researchers can identify bio-markers, make a early diagnosis, observe disease progression, response to treatment and so on. Peak detection is one of most important steps in the analysis of mass spectrum because its performance directly affects the

*to whom correspondence should be addressed

other processing steps and final results such as profile alignment [1], bio-marker identification [2], and protein identification [3].

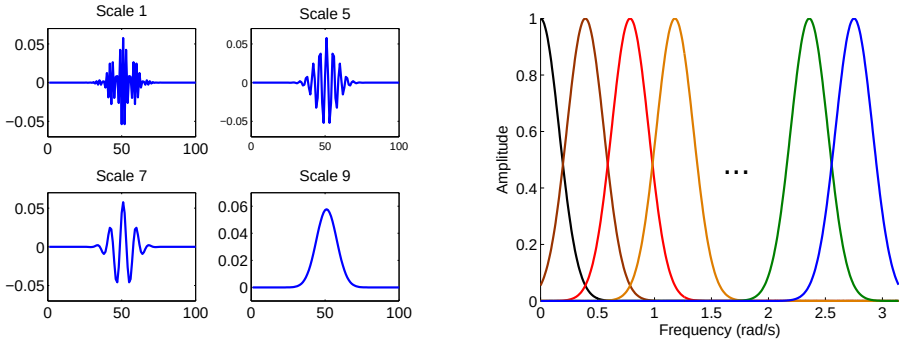
There are two types of peak detection approaches: denoising [4,5] and non-denoising (or decomposing) [6,7] methods. There are several similar steps between these two approaches such as baseline correction, alignment of spectrograms, and normalization. They also use local maxima to detect peak positions and use some rules to quantify peaks. Specially, both approaches use the signal to noise ratio (SNR) to remove the small energy peaks whose SNR values are less than a threshold. However, in the denoising approach, before detecting peaks, a denoising step is added to reduce the noise of mass spectrum data. In the non-denoising approach, a decomposition step is used to analyze mass spectrum into different scales before the peak detection by local maxima. When the smoothing step is applied into the denoising approach, it possibly removes both noise and signal. If the real peaks are removed by smoothing step, they can never be recovered in the other processing steps. As a result, we lose some important information and introduce error into MS data analysis. Thus, the way we will introduce to decompose a signal into many scales without denoising is a really better approach with great potentials.

The SNR is used to identify peaks in both denoising and non-denoising methods. Du *et al* [6] estimated the SNR in the wavelet space and got much better results than the previous work. However, they still failed to detect some peaks with small SNRs [6]. This problem came from the SNR value estimation and all previous methods estimated the SNR value by using the relationship between the peak amplitude and the surrounding noise levels. Since some sources of noise can also have high amplitudes, the high amplitude peak does not always guarantee to be real peak. On the other hand, some low amplitude peaks can also be real peaks. It is clear that the way using SNR to quantify peaks is not efficient and not accurate.

In this paper, we propose two novel robust MS peak detection approaches: GaborLocal and GaborEnvelop. First we use the Gabor filters to create many scales from original signal without smoothing. The Gaussian local maxima is exploited to detect peaks in the GaborLocal method instead of the local maxima that is less robust to the noise of mass spectrum. Furthermore, the envelope analysis is also proposed and applied to detect peaks in the GaborEnvelop method. Finally, we use the peak rank (PR) to remove some false peaks instead of the SNR. The real SELDI-TOF spectrum with known polypeptide composition and position is used to evaluate our method. The experimental results show that our new approaches can detect both high amplitude and small amplitude peaks with a low false discovery rate and are much better than the previous methods. We also compare two proposed methods in section 3.3.

2. METHODS

Our proposed methods are integrations of Gabor filters and Gaussian local maxima or envelope analysis. In this section, we first introduce the basic knowledge of Gabor



(a) (b)

Fig. 1. The uniform Gabor filters (a) The real parts, (b) The frequency supports.

filters and envelope analysis. After that, our proposed methods will be detailed. At last, we will use some examples to demonstrate how our methods work.

2.1. *Gabor Filters*

The Gabor filters [8] were developed to create Gaussian transfer functions in the frequency domain. Thus, taking the inverse Fourier transform of this transfer function, we get a filter closely resembling to the Gabor filters. The Gabor filters have been shown to have optimal combined localization in both the spatial and spatial-frequency domains [9,10]. In certain applications, this filtering technique has been demonstrated to be robust and fast [11] and the recursive implementation of 1D Gabor filtering has been shown in paper [12]. This recursive algorithm for the Gabor filter possibly achieves the fastest implementation. For a signal consisting of N samples, this implementation requires $O(N)$ multiply-and-add (MADD) operations. A generic one-dimensional Gabor function and its Fourier transform are given by:

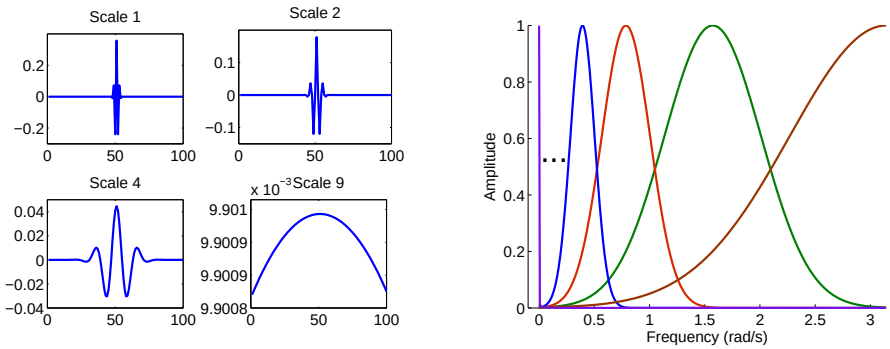
$$h(t) = \frac{1}{\sqrt{2\pi}\sigma} \exp(-\frac{t^2}{2\sigma^2}) \exp(j2\pi F_i t), \tag{1}$$

$$H(f) = \exp(-\frac{(f - F_i)^2}{2\sigma_f^2}), \tag{2}$$

where $\sigma_f = 1/(2\pi\sigma)$ represents the bandwidth of the filter and F_i is the central frequency.

The Gabor filter can be viewed as a Gaussian modulated by a complex sinusoid (with center frequencies F_i). This filter responds to some frequency, but only in a localized part of the signal. The coefficients of Gabor filters are complex. Therefore, the Gabor filters have one-side frequency support as shown in Fig. 1 and Fig. 2. We also illustrate the real parts of the Gabor filters in Fig. 1 and Fig. 2.

Given a certain number of sub-bands, in order to obtain a Gabor filter bank, the central frequencies F_i and bandwidths σ_f of these filters are chosen to ensure



(a) (b)

Fig. 2. The non-uniform Gabor filters (a) The real parts, (b) The frequency supports.

that the half-peak magnitude supports of the frequency responses touch each other as shown in Fig. 1 and Fig. 2. The Gabor filter bank can be designed to be uniform (in Fig. 1) or non-uniform (in Fig. 2). In our experiments, we use the Gabor filter bank with nine non-uniform sub-bands.

After decomposing a MS signal, nine sub-bands are created as follows:

$$y_i(t) = h_i(t) * x(t), \tag{3}$$

where $x(t)$ is the input signal, i = 1,2,...,9, and $*$ is the 1D convolution. This is an over-complete representation with the redundant ratio of 9.

2.2. *Envelope Analysis*

Envelope analysis and its theory will be described in this section. We first introduce Gaussian local maxima, minima and interpolation before studying envelope analysis.

Gaussian local maxima and minima: We assume that we want to find local maxima and local minima of $y(t)$. We should follow two steps: computing derivative of $y(t)$ and finding zero crossing. The derivative of $y(t)$ is approximated by the finite difference as follows:

$$\frac{d(y(t))}{dt} = \lim_{h->0} \frac{y(t+h) - y(t)}{h} \approx y(t+1) - y(t). \tag{4}$$

At $t = t_0$, if the derivative of $y(t)$ equals to zero and has a change from positive to negative or from negative to positive, we have zero-crossing. If the derivative of $y(t)$ changes from positive to negative at t_0, we have local maxima at t_0. Otherwise, if the derivative of $y(t)$ changes from negative to positive at t_0, we have local minima at t_0. With discrete signal, Eq. 4 can be rewritten as follows

$$\frac{d(y(n))}{dn} = y(n+1) - y(n) = y(n) * [1 \quad -1]. \tag{5}$$

Table 1. The value of vector v(n) with different lengths.

length	n = 1	2	3	4	5	6	7	8	9
5	0.0007	0.2824	0	-0.2824	-0.0007				
6	0.0007	0.1259	0.7478	-0.7478	-0.1259	-0.0007			
7	0.0007	0.0654	0.6572	0	-0.6572	-0.0654	-0.0007		
8	0.0007	0.0388	0.4398	0.6372	-0.6372	-0.4398	-0.0388	-0.0007	
9	0.0007	0.0254	0.2824	0.7634	0	-0.7634	-0.2824	-0.0254	-0.0007

Unfortunately, MS data always have noise. Thus, we use Gaussian filter $g(t,\sigma)$ to make our methods more robust to noise in MS data. This is not a denoising step because the noise is not removed. Finally, derivative of $y(t) * g(t,\sigma)$ will replace the derivative of $y(t)$ as follows

$$\frac{d(y(t) * g(t,\sigma))}{dt} = \frac{d(\int (y(\tau).g(t-\tau,\sigma)d\tau))}{dt} \tag{6}$$

$$= \int (y(\tau).\frac{d(g(t-\tau,\sigma))}{dt}d\tau) = y(t) * \frac{d(g(t,\sigma))}{dt}, \tag{7}$$

where

$$g(t,\sigma) = \exp(-\frac{t^2}{2\sigma^2}). \tag{8}$$

Taking the derivative of $g(t,\sigma)$ in Eq. 8, we have

$$\frac{d(g(t,\sigma))}{dt} = \frac{-t}{\sigma^2}\exp(-\frac{t^2}{2\sigma^2}). \tag{9}$$

From Eq. 6 and Eq. 9, we have

$$\frac{d(y(t) * g(t,\sigma))}{dt} = y(t) * (\frac{-t}{\sigma^2}\exp(-\frac{t^2}{2.\sigma^2})). \tag{10}$$

Instead of finding zero crossing of $\frac{d(y(t))}{dt}$, we find zero-crossing of $\frac{d(y(t)*g(t,\sigma))}{dt}$ by Eq. 10. With discrete signal, Eq. 10 can be rewritten as follows

$$\frac{d(y(n) * g(n,\sigma))}{dn} = y(n) * v(n), \tag{11}$$

where $v(n)$ is listed in the table 1. Using Gaussian filters makes the Gaussian local maxima and minima method more robust with noise.

Interpolation: Interpolation is a process to estimate new data points that lie between known data points. Polynomial interpolation and FFT-based interpolation are two basic one-dimensional interpolation techniques which can balance speed of execution and memory usage. In this paper, we use the simplest method of polynomial interpolation which is linear interpolation. This method fits a different linear function between each pair of existing data points (x_a, y_a) and (x_b, y_b) and returns the value of the relevant function at the points specified by x_i as follows

$$y_i = y_a + \frac{(x_i - x_a)(y_b - y_a)}{x_b - x_a}. \tag{12}$$

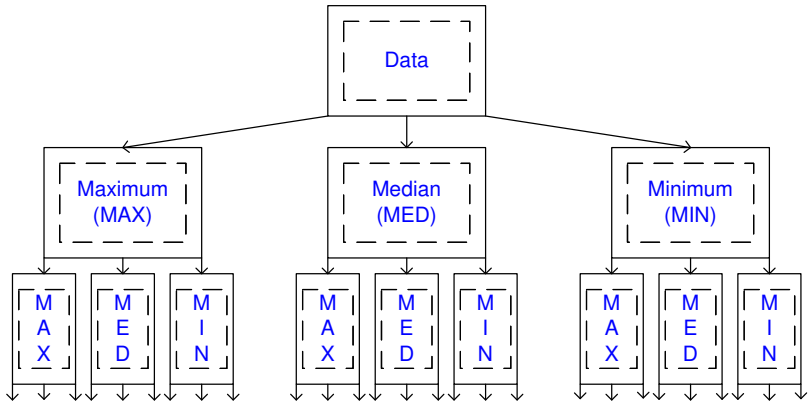


Fig. 3. Flowchart of envelope Analysis. Data can be decomposed into three envelope signals: Maximum envelope (MAX), Median envelope (MED), Minimum envelope (MIN).

Linear interpolation uses less memory and execution time than cubic interpolation and the interpolated data are continuous.

Before going to detail of envelope analysis, we should consider three following definitions (1, 2, 3).

Definition 1. Maximum envelope (MAX) of a signal $y(t)$ is a signal which is created from $y(t)$ by two steps.
Step 1: finding Gaussian local maxima of $y(t)$ and their indices: Max($y(t)$).
Step 2: taking interpolation of signal obtained from step 1 so that MAX will have the same length as $y(t)$ as follows: MAX = Interp(Max($y(t)$)).

Definition 2. Minimum envelope (MIN) of a signal $y(t)$ is a signal which is created from $y(t)$ by two steps.
Step 1: finding Gaussian local minima of $y(t)$ and their indices: Min($y(t)$).
Step 2: taking interpolation of signal obtained from step 1 so that MIN will have the same length as $y(t)$ as follows: MIN= Interp(Min($y(t)$)).

Definition 3. Median envelope (MED) of a signal $y(t)$ is a signal which is created from $y(t)$ by two steps.
Step 1: finding non Gaussian local maxima and non Gaussian local minima of y(t) and their indices: Med($y(t)$)
Step 2: taking interpolation of signal obtained from step 1 so that MED will have the same length as $y(t)$ as follows: MED = Interp(Med($y(t)$)).

Envelope Analysis: For many signals, the important property is energy. If we analyze these signals using Fourier or Wavelet Transforms, we just use the frequency property. It is really difficult for some applications which need using energy property

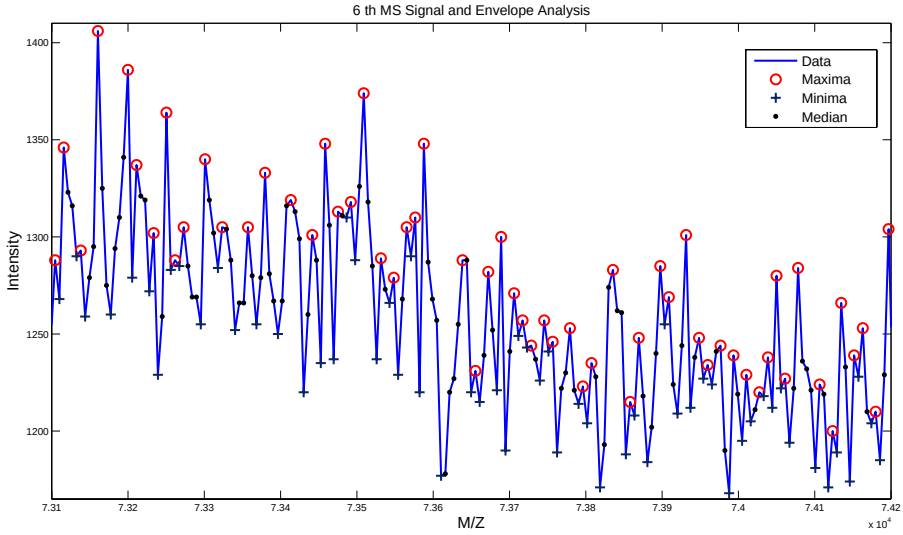


Fig. 4. One example of finding local maxima, local median and local minima of Raw MS data which is the 6^{th} MS signal of CAMDA 2006. Only a part of signal is shown. Circle, point and plus signs represent for maxima, median and minima positions

to classify or to detect some important information.

Any finite energy signal $y(t)$ can be analyzed into three envelope signals (Fig. 3) including MAX (M_{11}), MED (M_{12}), and MIN (M_{13}) at the first level. Each of three above envelope signals will be decomposed into three envelope signals at the second level and we get $3^2 = 9$ envelope signals totally. This process is iterated and at the i^{th} level, $y(t)$ is decomposed into 3^i envelope signals from M_{i1} to M_{i3^i}. So, we can formulate the envelope analysis as follows

$$\begin{aligned} Level_1 &= [Env(y(t))] = [M_{11}; M_{12}; M_{13}] \\ Level_2 &= [Env(M_{11}); Env(M_{12}); Env(M_{13})] = [M_{21}; M_{22}; ...; M_{29}] \\ Level_i &= [Env(M_{(i-1)1}); Env(M_{(i-1)2}); Env(M_{(i-1)3}); .....Env(M_{(i-1)3^{i-1}})] \\ &= [M_{i1}; M_{i2}; ...; M_{i3^i}], \end{aligned} \tag{13}$$

where $Env(y(t))$=[M_{11}=Interp(Max($y(t)$)); M_{12}=Interp(Med($y(t)$)); M_{13}=Interp(Min($y(t)$))].

In this paper, we just use MAX and MED of envelope analysis to detect peaks because MIN does't contain any peaks. Eq. 14 describes the structure of envelope analysis which we use in our proposed method.

$$\begin{aligned} Level_1 &= [\texttt{MAX1}; \texttt{MED1}], \\ Level_2 &= [\texttt{MAX2}; \texttt{MED2}], \\ Level_i &= [\texttt{MAXi}; \texttt{MEDi}]. \end{aligned} \tag{14}$$

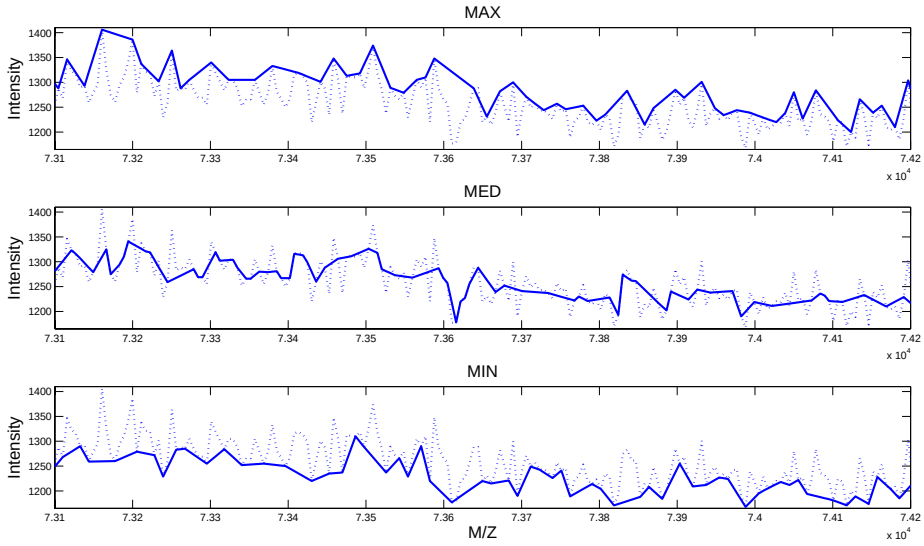


Fig. 5. One example of maximum envelope (MAX), median envelope (MED) and minimum envelope (MIN) at the first level. The input data is the 6^{th} MS signal of CAMDA 2006.

Illustration: Fig. 4 and Fig. 5 show the envelope analysis of an example. First, at the Fig. 4, the Gaussian local maxima, median, and minima are detected from the 6_{th} MS signal of CAMDA 2006. At level 1, after interpolating from the above maxima, median and minima positions, we can get three envelope signals (MAX, MED and MIN) as in Fig. 5. If we continue using these above envelope signals as the input signal of envelope analysis, we will get the next level, level 2.

2.3. *GaborLocal and GaborEnvelop Methods*

Our main idea is to amplify the true signal and compress the noise of mass spectrum by using the Gabor filter bank. After that, we use the Gaussian local maxima to detect peaks and the peak rank which will be defined later to quantify peaks. This method is named as Gabor filter - Gaussian local maxima (GaborLocal). We can also use envelope analysis to detect and quantify peaks and we call this method as Gabor filter - envelope analysis (GaborEnvelop). Fig. 6(a) and (b) are the flowchart of our GaborLocal and GaborEnvelop methods. Each method can be detailed into the four steps including the full frequency MS signal generation, the peak detection, the peak quantification, and the intersection. Both methods have the same first step (full frequency MS signal generation) and the same last step (intersection). They are different at the peak detection and the peak quantification steps.

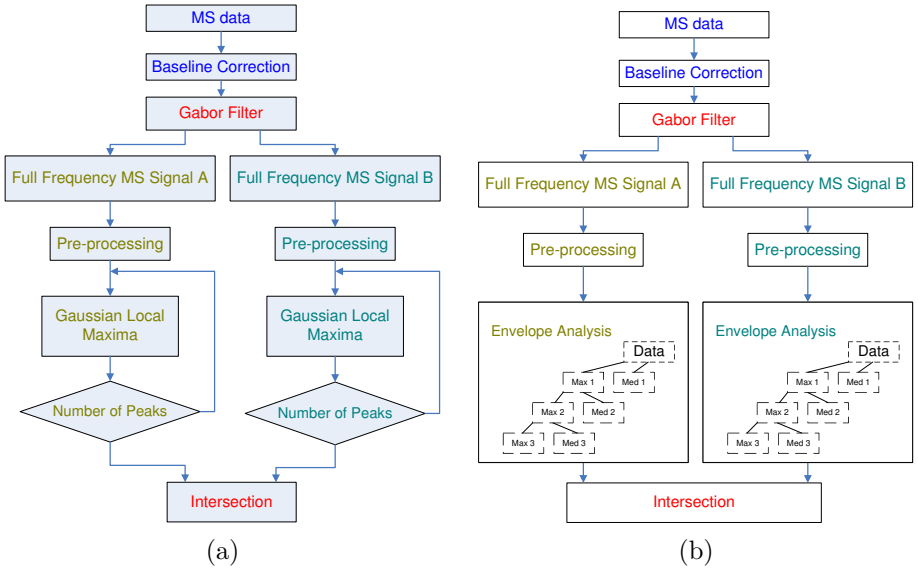


Fig. 6. Flowchart of our two peak detection methods in the MS data. (a) GaborLocal, (b) GaborEnvelop.

2.3.1. *Full frequency MS signal generation*

Mass spectrum is decomposed to many scales by using the Gabor filters after the baseline correction. Our purpose is to emphasize some hidden peaks buried by noise. When we analyze 60 MS signals of the CAMDA 2006 in the frequency domain, we notice that the valuable information of these signals locate from zero to around 0.06 (rad/s) and the noises locate from 0.06 to π (rad/s).

Therefore, the bandwidth σ_f of the Gabor filters which enhances peaks must be less than 0.06. In our experiments, we use $\sigma_f = 0.01$. If the uniform Gabor filter is used, the number of scales must be

$$N = \frac{\pi}{0.01} \approx 314 \; scales. \tag{15}$$

With 314 scales in Eq. 15, we know that the uniform Gabor filter is not efficient. If the non-uniform Gabor filter is used, the number of scales should be calculated as follows:

$$\sigma_f = \frac{\pi}{2^N}, \tag{16}$$

$$N = log_2(\frac{\pi}{\sigma_f}), \tag{17}$$

$$N \approx 8.3 \; scales \; with \; \sigma_f = 0.01. \tag{18}$$

Based on the Eq. 18, we use the non-uniform Gabor filters with 9 scales to decompose the MS data (we use CAMDA 2006 data [13] for experiments). If we

transform $y_i(t)$, $h_i(t)$ and $x(t)$ in Eq. 3 into the frequency domain, we get

$$Y_i(f) = X(f) \cdot H_i(f), \tag{19}$$

where $X(f)$ is the frequency response of the raw MS signal, $H_i(f)$ is the frequency response of the i^{th} Gabor filter, and $Y_i(f)$ is the frequency response of the i^{th} scale. After getting 9 signals according to 9 frequency sub-bands in complex values, the full frequency signal A will be created by summing above signals in complex values first and taking their absolute values at the final. To create the full frequency signal B, we take the absolute values for each sub-band and then sum all these sub-bands. After this step, we have two full frequency signals A and B. Let's denote $y(t)$ and $Y(f)$ as the full frequency signal in time domain and frequency domain, respectively.

$$Y(f) = \sum_{i=N_i} Y_i(f), \tag{20}$$

where N_i are the scales which are used to create the full frequency signal. From Eq. 19 and 20, we get

$$Y(f) = \sum_{i=N_i} X(f)H_i(f) \tag{21}$$

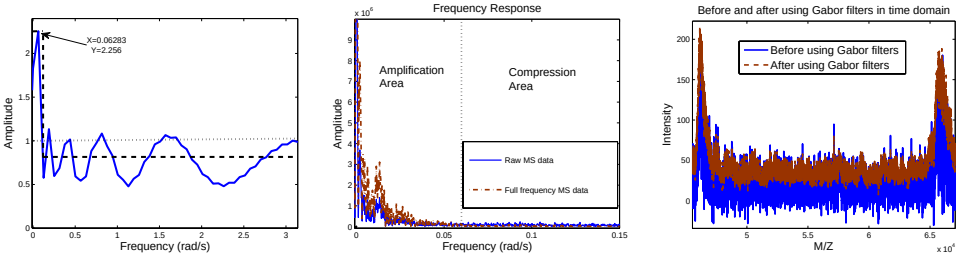
$$= X(f) \sum_{i=N_i} H_i(f) = X(f)H_s(f), \tag{22}$$

where $H_s(f) = \sum_{i=N_i} H_i(f)$ is called the summary filter. From Eq. 2, the summary filter can formulated as follows

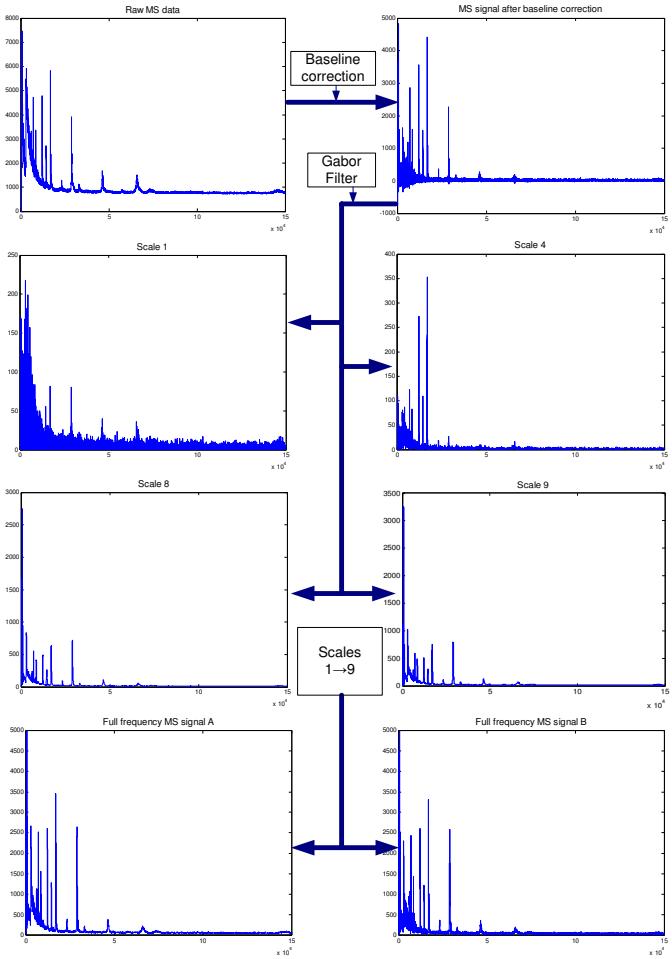
$$H_s(f) = \sum_{i=N_i} \exp(-\frac{(f-F_i)^2}{2\sigma_f^2}). \tag{23}$$

Illustration: Intuition using Gabor filters Our purpose in this step is to amplify the true signal and to compress the noise. The black line in the Fig. 7(a) is $H_s(w)$ which can amplify the true signal from 0 to 0.06 $\frac{rad}{s}$ and compress noise from 0.06 to π. In this case, if we use $N_i = [1\ 2\ ...\ 9]$ we can get the summarized filter represented by the blue line in Fig. 7(a). The Fig. 7(b) shows the frequency response of the 19^{th} raw MS signal (blue line) and that of full frequency signal (red line). We can see that the signal from 0 to 0.06 is amplified and the noise from 0.06 to π is compressed. In Fig. 7(c), after using Gabor filters, the intensity values of true peaks have increased and the standard deviations of noise have decreased in time domain. Therefore, in both full frequency MS signal A and B, all peaks have been emphasized to help the next peak detection step. In this step, baseline correction is also used before applying Gabor filters and is detailed as follows

Baseline correction: The chemical noise or the ion overloading is the main reason causing a varying baseline in mass spectrometry data. Baseline correction is an important step before using Gabor filter to get the full frequency MS signals.



(a) (b) (c)



(d)

Fig. 7. (a) The frequency response of the summary filter. (b) The frequency response of one MS data before and after using the summary filter. In amplification area, amplitude of full frequency MS signal is higher than raw MS signal. In compression area, amplitude of full frequency MS signal is smaller than raw MS signal.(c) One example is used to show how Gabor filters to affect MS signal in time domain. The intensity values of peaks are gained and noise is compressed after using Gabor filters. (d) One example of the step named full frequency MS signal generation. Raw MS data which is used in (b), (c) and (d) is the 19^{th} MS signal of CAMDA 2006.

The raw MS signal x_{raw} includes some real peaks x_p, the baseline x_b, and the noise x_n.

$$x_{raw} = x_p + x_b + x_n. \tag{24}$$

The baseline correction is used to remove the artifact x_b. In this paper, we use 'msbackadj' function of MATLAB to remove baseline. The msbackadj function estimates a low-frequency baseline first which is hidden among the high-frequency noise and the signal peaks and then subtracts the baseline from the spectrogram. This function follows the algorithms in Andrade *et al.*'s paper [14].

Illustration: In order to understand this step easier, one example of the way to create full frequency MS signal is shown in Fig. 7(d). In this example, the 19^{th} MS signal of CAMDA 2006 is chosen as raw MS data. After the baseline correction, MS signal is used as the input of the Gabor filters. A Gabor filter bank with 9 non-uniform sub-bands is employed to create 9 MS signals with 9 different frequency sub-bands. In Fig. 7(d), the signals of scale 1, 4, 8 and 9 are visualized. Some noises in high frequency are separated from the MS signal of the scale 1, 2, ..., 5. In the MS signal under the scales 6, ..., 9, all high intensity peaks are still kept. After combining the MS signals of all scales in two ways, the full frequency MS signal A and B are created. The comparison between the raw MS and full frequency signal in frequency and time domain is shown in Fig. 7(a)(b)(c). These figures show our purpose which amplifies the important signal and compresses the noise has been achieved. We should remember that this is just a compression of noise instead of removing noise. As the outputs, two full frequency MS signal A and B will be used to detect peaks in the next step instead of raw MS data.

2.3.2. *Peak detection and peak quantification in GaborLocal*

All peaks are detected as many as possible by using Gaussian local maxima with the full frequency MS signal A as well as the full frequency MS signal B. The Gaussian local maxima is used instead of local maxima because Gaussian local maxima is robust with noise in peak detection. Before detecting peaks, pre-processing step is also applied such as peak elimination in the low-mass region.

After detecting many peaks in full frequency MS signals, a new signal is obtained from these peaks. This new signal will be the input of the next peak detection loop where the Gaussian local maxima method is also applied. Then, many loops are repeated until the number of peaks obtained is less than a threshold. Now, we define the peak rank of peaks as follows:

Peak rank in GaborLocal: We assume n loops are used and get m_1 peaks at the loop 1, m_2 peaks at loop 2,...and m_n peaks at the loop n. We have $m_1 > m_2 > ... > m_n$. Peak rank (PR) is defined as the table 2.

We have m_n peaks with $PR = 1$, $m_{n-1} - m_n$ peaks with $PR = 2$,...and $m_1 - m_2$ peaks with $PR = n$. In our algorithm, the probability of the true peaks with $PR = i$ is higher than with $PR > i$.

Table 2. Definition of peak rank in GaborLocal. Y means that the peak can be detected at that loop. N means that the peak can not be detected at that loop. The peak with the peak rank equaling to 1 is able to be detected at all of the loops. The peak with the peak rank equaling to n only appeared at the first loop.

Peak Rank	*Loop* 1	*Loop* 2	*Loop* 3	*Loop* 4	... *Loop* $(n-1)$	*Loop* n
1	Y	Y	Y	Y	... Y	Y
2	Y	Y	Y	Y	...Y	N
...	...	...	...	...	...	...
n	Y	N	N	N	...N	N

Demonstration: Fig. 8(a) shows an example of the step named the peak quantification by using the peak rank. First, the full frequency MS signal A is used to detect peaks by using Gaussian local maxima. At the loop 1, we can detect 1789 peaks. From these 1789 peaks, we create a new signal with 1789 positions. At the next loops 2, 3, 4, we can detect 509, 143, 39 peaks, respectively. At the loop 5, 15 peaks can be detected. Because we choose a threshold of 16 and $number\ of\ peaks = 15 < 16$, we stop at the loop 5. Actually, we can select the threshold from 38 to 16 and also get 15 peaks at the final loop. Now, we get 15 peaks with $PR = 1$, $39 - 15 = 24$ peaks with $PR = 2$, $143 - 39 = 104$ peaks with $PR = 3$, $509 - 143 = 366$ peaks with $PR = 4$ and $1789 - 509 = 1280$ peaks with $PR = 5$. In this case, we only keep 15 peaks with $PR = 1$. We also do the same on the full frequency MS signal B and can get 12 peaks with $PR = 1$ at the last loop.

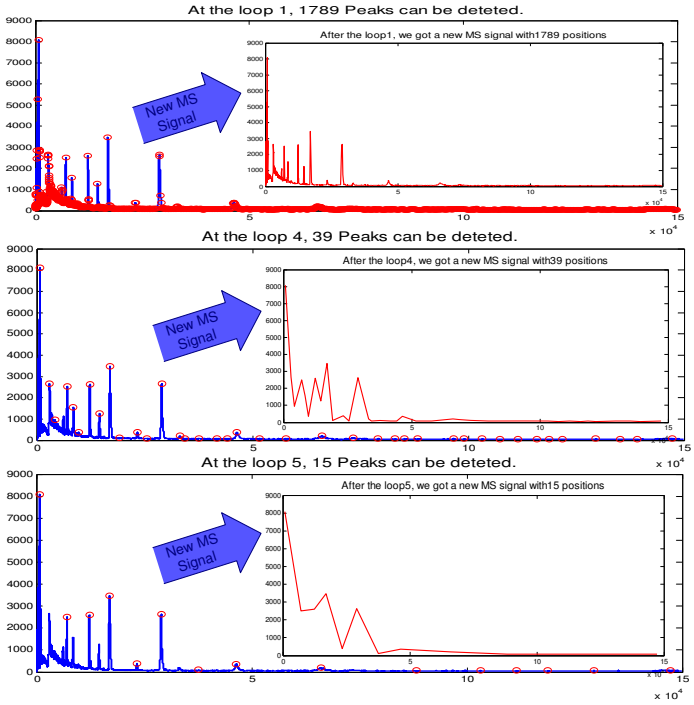
2.3.3. *Peak detection and peak quantification in GaborEnvelop*

After using Gabor filter, we get visible peaks in the full frequency MS signal A and the full frequency MS signal B. Instead of using Gaussian local maxima to detect peaks, we take envelope analysis of both signal A and signal B and use MAX and MED at the final level to find peaks. Fig. 6(b) shows that we only use MAX and MED instead of MAX, MED, and MIN because peaks of data don't appear at MIN envelope. Of course, before taking envelope analysis of signal A and B, peak elimination in low-mass region is also applied.

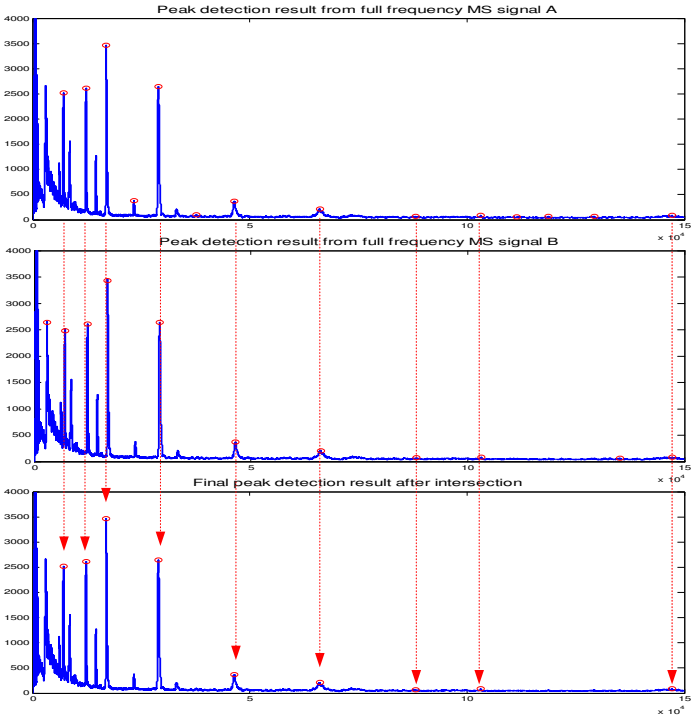
Peak rank in GaborEnvelop: We take envelope analysis of data at the level n. Because only MAX is used at the first level, we have $2n - 1$ groups of peaks corresponding $2n - 1$ thresholds of peak. Let's assume we get m_1 peaks at the MAX1, m_2 peaks at group of MAX2 and MED2,...and m_n peaks at the MAXn. We have $m_1 > m_2 > ... > m_n$. Peak rank (PR) is defined in table 3.

We have m_n peaks with $PR = 1$, $m_{n-1} - m_n$ peaks with $PR = 2$,...and $m_1 - m_2$ peaks with $PR = n$. In our algorithm, the probability of the true peaks with $PR = i$ is higher than with $PR > i$.

Demonstration: Fig. 9 shows an example of envelope analysis of the 39^{th} MS signal at level 6 and 7. Only MAX and MED are used in this case. The input signal of envelope analysis is full frequency MS signal A without pre-processing. At the



(a)



(b)

Fig. 8. One example of GaborLocal in peak detection (a) peak detection and quantification step, (b) intersection step.

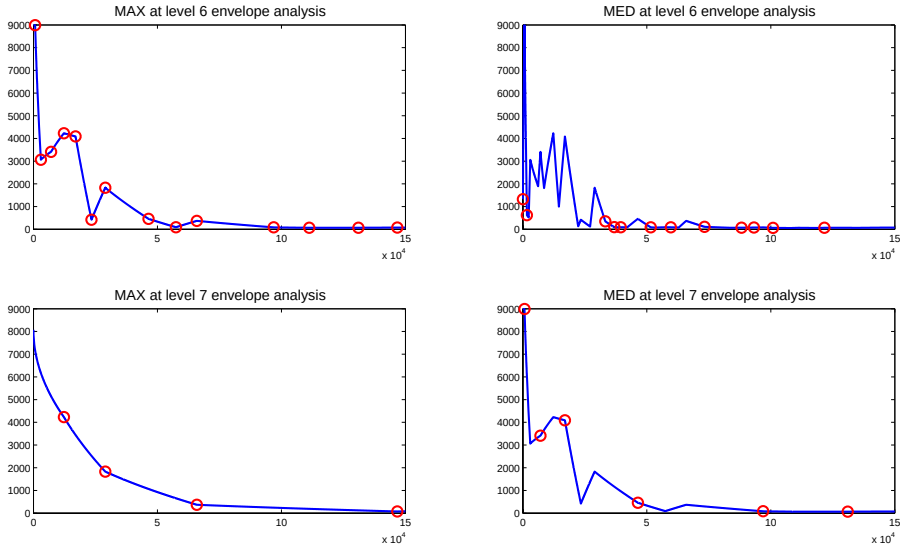


Fig. 9. One example of envelope analysis. Maximum envelope, median envelope, and minimum envelope of a MS data at the level 6 and the level 7. The input MS data is the 39^{th} MS signal of CAMDA 2006. The circle sign represents peaks which are from the MAX1 (peaks of the input MS signal) and are identified peaks

Table 3. Definition of peak rank (PR) in GaborEnvelop. MAXi column means that we just use peaks of MAX envelope at level i. If we use peaks in both of MAX and MED envelope at level i, that column is named as MAXi, MEDi. "Y" is the peak can be detected. N means that the peak can not be detected. The peak with the peak rank equaling to 1 is able to be detected at all of envelopes. The peak with the peak rank equaling to n only appeared at the envelope of level 1.

PR	$MAX1$	$MAX2, MED2$	$MAX2$	$MAX3, MED3$	... $MAXn, MEDn$	$MAXn$
1	Y	Y	Y	Y	... Y	Y
2	Y	Y	Y	Y	...Y	N
...	...	...	...	...	...	...
n	Y	N	N	N	...N	N

MAX scale of level 6, we detect 14 peaks. If continuing level 7, we get 4 peaks at the MAX scale and 6 peaks at the MED scale. If pre-processing is applied to full frequency MS signal A, we just get 5 peaks at the MED scale of level 7. Finally, we get $4 + 5 = 9$ peaks from signal A and 19 peaks from signal B.

2.3.4. *Intersection*

Now, we have two results of peak detection from two full frequency MS signals. The intersection of two above results will be the final result. For example, Fig. 8(b)

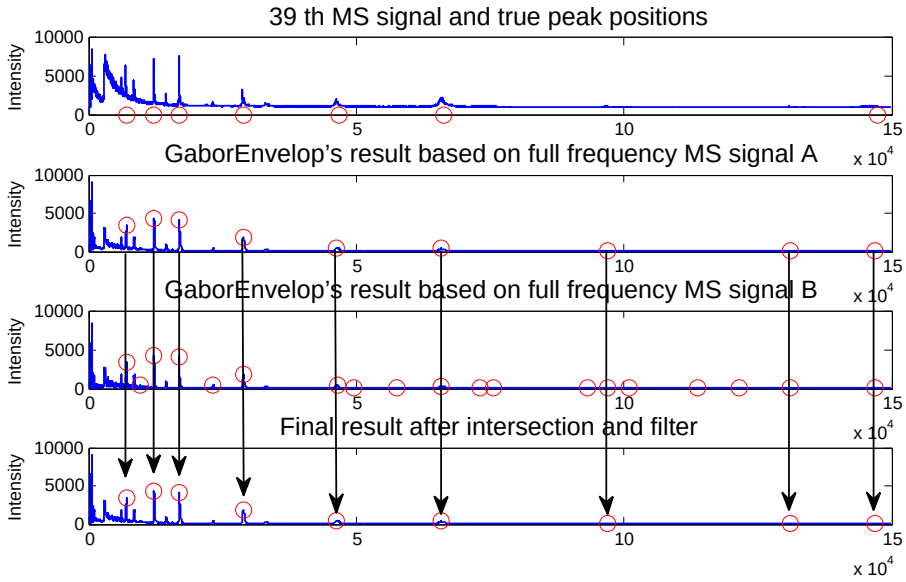


Fig. 10. One example of GaborEnvelop at intersection step. Raw MS data is the 39^{th} MS signal of CAMDA 2006.

shows how to do the intersection of two results. We have 15 peaks in the signal A and 12 peaks in the signal B, but we only get 9 peaks as the final result. With this result, we get 7 true peaks and 2 false peaks. With example in Fig. 10, after intersection, we get 9 peaks. We also get 7 true peaks and 2 false peaks. These results show that the true position rate (or sensitivity) equals to $\frac{7}{7} = 1$ and the false discovery rate equals to $\frac{2}{9} \approx 0.22$.

In general, the GaborEnvelop includes the GaborLocal. In envelope analysis, if we just use MAX envelope signals, the GaborEnvelop will become the GaborLocal method which uses many loops to quantify peaks. The GaborEnvelop uses both MAX and MED envelopes to keep the number of true peaks (TPR) and decrease the number of false peaks (FDR).

3. EXPERIMENTAL RESULTS AND DISCUSSIONS

In this section, our GaborLocal and GaborEnvelop methods will be compared to two other most commonly used methods: the Cromwell [4,5] and the CWT [6]. We will evaluate the performance of those methods by using the ROC curve that is the standard criterion in this area.

3.1. *Cromwell Method*

Cromwell method is implemented as a set of MATLAB scripts which can be downloaded from [15]. The algorithms and the performance of the Cromwell were described in [5,4]. The main idea of the Cromwell method can be summarized as follows

(a) Denoise the individual spectrum using the undecimated discrete wavelet transform. The hard thresholding method was used to reset small wavelet coefficients to zero. In these papers, the authors used the median absolute deviation (MAD) to estimate the thresholding.
(b) Estimate and remove the baseline artifact by using a monotone local minimum curve on the smoothed signal.
(c) Normalize the spectrum by dividing the total ion current, defined to be the mean intensity of the denoised and baseline corrected spectrum.
(d) Identify peaks by using local maxima and signal to noise ratio (SNR).
(e) Match peaks across spectrum and quantify peaks using either the intensity of the local maximum or computing the area under the curve for the region defined to be the peaks.

3.2. *CWT Method*

The algorithm of CWT method has been implemented in R (called as 'MassSpecWavelet') and the Version 1.4 can be downloaded from [16]. This method was proposed by Pan Du *et al.* [6] in 2006 and can be summarized as follows:

(a) Identify the ridges by linking the local maxima. Continuous wavelet transform (CWT) is used to create many scales from one mass spectrum. The local maxima at each scale is detected. The next step is to link these local maxima as lines.
(b) Identify the peaks based on the ridge lines. There were three rules to identify the major peaks. They are the scale with the maximum amplitude on the ridge line, the SNR being larger than a threshold and the length of ridge being larger than a threshold. We should notice that the SNR is estimated in the wavelet space. This is a nice motivation of this method.
(c) Refine the peak parameter estimation.

3.3. *Evaluation Using ROC Curve*

The CAMDA 2006 dataset [13] of all-in-1 Protein Standard II (Ciphergen Cat. # $C100-007$) is used to evaluate four algorithms: the Cromwell, the CWT, and our two methods. Because we know polypeptide composition and position, we can estimate the true position rate (TPR or sensitivity) and the false discovery rate (FDR). Another advantage of this dataset is that it is real data and better than the simulated data in evaluation.

The TPR is defined as the number of identified true peaks divided by the total number of true peaks. The FDR is defined as the number of falsely identified peaks divided by the total number of identified peaks. We call an identified

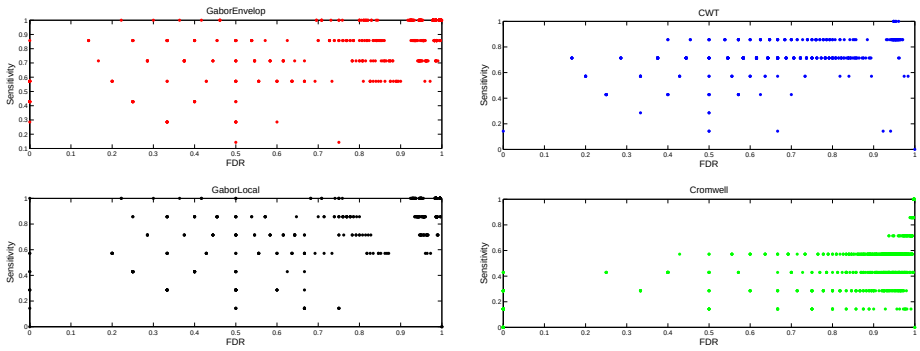


Fig. 11. Detailed ROC curves obtained from 60 MS signals using Cromwell, CWT, and our GaborLocal and GaborEnvelop methods. The sensitivity is the true position rate.

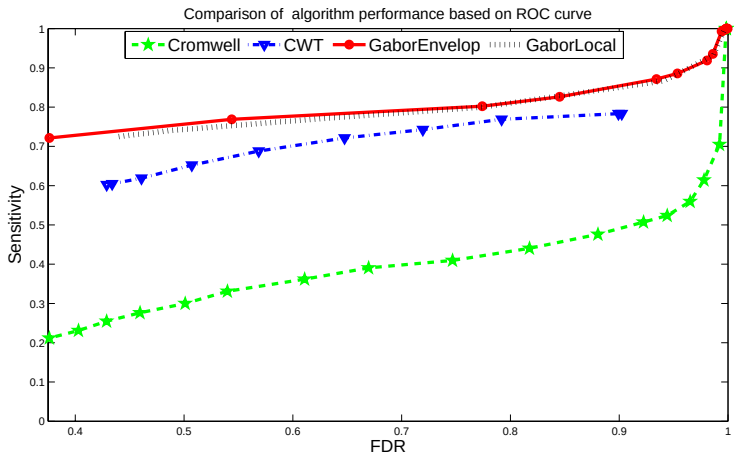


Fig. 12. Average receiver operating characteristic (ROC) curves obtained from 60 MS signals using Cromwell, CWT, and our GaborLocal and GaborEnvelop methods. The sensitivity is the true position rate.

peak as true peak if it is located within the error range of 1% of the known m/z value of true peaks. There are seven polypeptides which create seven true peaks at 7034, 12230, 16951, 29023, 46671, 66433 and 147300 of the m/z values. Fig. 11 shows the TPR and the FDR of four above methods with an assumption that there is only one charge. To calculate the ROC curve of Cromwell and CWT methods, the SNR thresholding values are changed. The SNR thresholding values are chosen from 0 to 20 for Cromwell method, from 0 to 65 for CWT method. In GaborLocal method, the threshold for the number of peaks is changed from 2000 to 10 to create the ROC curve. In GaborEnvelop method, the level is changed from seven to one to build the ROC curve. In Fig. 11, the performance of Cromwell method is much

worse than CWT and our GaborLocal and GaborEnvelop methods. Most of ROC points of Cromwell method locate at the bottom of right corner and most of ROC points of CWT, GaborLocal, and GaborEnvelop methods are well placed on the top regions. In our methods, some ROC points appear at the top line with $TPR = 1$ and some ROC points go with $TPR = 1$ and $FDR = 0$. However, it does not happen to the CWT. Therefore, GaborLocal and GaborEnvelop are better than CWT and Cromwell in peak detection.

If we take the average of those detailed ROC results in Fig. 11, we get the average ROC curve as the Fig. 12. We should notice that we take average of all ROC points with the same SNR threshold (for Cromwell and CWT) and with the same peak threshold and the same level (for our methods). From Fig. 12, the results of our methods and CWT are much better than the Cromwell's one. Therefore, the decomposing approach without smoothing (SWT, GaborLocal and GaborEnvelop) is more efficient than the denoising approach (like Cromwell). At the same FDR, the TPRs of our methods are consistently higher than the TPRs of CWT. Because the peak rank was used to identify peaks in the GaborLocal and GaborEnvelop methods instead of the SNR. It is clear that the utilizing peak rank to identify peak gives out valuable results. These methods have a significant contribution to detect both high energy and small energy peaks. The other advantage of these methods is that the threshold for the number of peaks can be created easier than the SNR. Therefore, the GaborLocal and GaborEnvelop method are more efficient and accurate methods for real MS data peak detection.

As shown in Fig. 12, GaborEnvelop is slightly better than GaborLocal in ROC curve. With the same TPR, GaborEnvelop gives out smaller FDR than GaborLocal. However, peak quantification step using many loops in GaborLocal method is simpler than using envelope analysis in GaborEnvelop one. If we need a simple method which also can detect most true peaks, GaborLocal is a good option. During more complicated analysis, GaborEnvelop can be employed to improve the results of peak detection in MS signal. Since the number of detected peaks increase gradually when the peak rank increases, GaborEnvelop is useful for many applications, e.g. protein identification. These are also the reasons that we propose two methods in this paper.

4. CONCLUSION

In this paper, we proposed two new approaches including GaborLocal and GaborEnvelop to solve peak detection problem in MS data with promising results. Our GaborLocal method is a combination of the Gabor filter and Gaussian local maxima approach. The integration of Gabor filter and envelope analysis is further developed as GaborEnvelop method. The peak rank method is presented and used at the first time to replace the previous SNR method to identify true peaks. With real MS dataset, our method gave out a much better performance in the ROC curve comparison with two other most common used peak detection methods. In

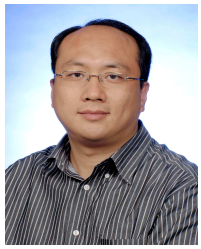
our future work, we will develop new protein identification method based on our GaborLocal and GaborEnvelop methods.

References

1. N. Jeffries, "Algorithms for alignment of mass spectrometry proteomic data," *Bioinformatics*, vol. 21, pp. 3066–3073, 2005.
2. J. e. Li, "Independent validation of candidate breast cancer serum biomarkers identified by mass spectrometry," *Clin Chem*, vol. 51, pp. 2229–2235, 2005.
3. T. e. Rejtar, "Increased identification of peptides by enhanced data processing of high-resolution maldi tof/tof mass spectra prior to database searching," *Anal Chem*, vol. 76, pp. 6017–6028, 2004.
4. J. Morris, K. Coombes, J. Koomen, K. Baggerly, and R. Kobayashi, "Feature extraction and quantification for mass spectrometry in biomedical applications using the mean spectrum," *Bioinformatics*, vol. 21, no. 9, pp. 1764–1775, 2005.
5. K. Coombes and *et al.*, "Improved peak detection and quantification of mass spectrometry data acquired from surface-enhanced laser desorption and ionization by denoising spectra with the undecimated discrete wavelet transform," *Proteomics*, vol. 5, no. 16, pp. 4107–4117, 2005.
6. P. Du, W. Kibble, and S. Lin, "Improved peak detection in mass spectrum by incorporating continuous wavelet transform-based pattern matching," *Bioinformatics*, vol. 22, no. 17, pp. 2059–2065, 2006.
7. E. Lange and *et al.*, "High-accuracy peak picking of proteomics data using wavelet techniques," in *Processdings of Pacific Symposium on Biocomputing*, 2006, pp. 243–254.
8. D. Gabor, "Theory of communication," *J. Inst. Elec. Engr*, vol. 93, no. 26, pp. 429–457, Nov 1946.
9. J. Kamarainen, V. Kyrki, and H. Kalviainen, "Invariance properties of gabor filter-based features-overview and applications," *IEEE Transactions on Image Processing,*, vol. 15, no. 5, pp. 1088–1099, May 2006.
10. J. Daugman, "Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters," *Journal of the Optical Society of America A*, vol. 2, pp. 1160–1169, 1985.
11. D. Tsai and C. Lin, "Fast defect detection in textured surfaces using 1d gabor filters," *The International Journal of Advanced Manufacturing*, vol. 20, no. 9, pp. 664–675, Oct. 2002.
12. I. Young and M. G. L. Vliet, "Recursive gabor filtering," *IEEE Transactions on Signal Processing*, vol. 50, no. 11, pp. 2798–2805, Nov 2002.
13. CDC-Chronic-Fatigue-Syndrome-Research-Group, "Camda 2006 conference contest datasets." [Online]. Available: *http://camda.duke.edu/camda06/datasets/index.html.*
14. *L. Andrade and L. Manolakos, "Signal background estimation and baseline correction algorithms for accurate dna sequencing,"* Journal of VLSI, special issue on Bioinformatics*, vol. 35, pp. 229–243, 2003.*
15. *UTMD-Anderson-Cancer-Center, "The new model processor for mass spectrometry data." [Online]. Available: http://bioinformatics.mdanderson.org/cromwell.html.*
16. *P. Du, "Mass spectrum processing by wavelet-based algorithms." [Online]. Available: http://bioconductor.org/packages/2.1/bioc/html/MassSpecWavelet.html.*

Nha Nguyen received his B.S and M.S degrees in Electrical Engineering from HCMC University of Technology, Viet Nam, in 1996 and 2000, respectively. He worked in the Sai Gon Technology University, Viet Nam, as a lecturer in the Department of Electrical Engineering from 2001 to 2007.

He is currently a Ph.D student at the University of Texas at Arlington, USA.
His research interests include telecommunication, signal processing in bioinformatics and biomedical image analysis.



Heng Huang received his Ph.D. in Computer Science from Dartmouth College in 2006, and his M.S. and B.S. in Information Measurement Technology and Instruments and Automation from Shanghai Jiao Tong University, respectively. His researches focus on bioinformatics, biomedical image analysis, computer vision, and pattern recognition. He joined Computer Science and Engineering deparmtent at University of Texas, Arlington as an assistant professor in 2007.
He is the director of Biomedical Computing and Scientific Visualization (BIOVIZION) lab.

Soontorn Oraintara received the B.E. degree (with first-class honors) from the King Monkuts Institute of Technology Ladkrabang, Bangkok, Thailand, in 1995 and the M.S. and Ph.D. degrees, both in electrical engineering, respectively, from the University of Wisconsin, Madison, in 1996 and Boston University, Boston, MA, in 2000.

He joined the Department of Electrical Engineering, University of Texas at Arlington (UTA), as an Assistant Professor in July 2000, where he has become an Associate Professor in September 2006. His current research interests are in the field of digital signal processing: wavelets, filterbanks, and multirate systems and their applications in data compression, signal detection and estimation, communications, image reconstruction, and regularization and noise reduction.
Dr. Oraintara received the Technology Award from Boston University for his invention on Integer DCT (with Y. J. Chen and T. Q. Nguyen) in 1999. He is an Associate Editor for the IEEE TRANSACTIONS ON SIGNAL PROCESSING. In 2003, he received the College of Engineering Outstanding Young Faculty Member Award from UTA. He represented Thailand in the International Mathematical Olympiad competitions and, respectively, received the Honorable MentionAward in Beijing,

China, in 1990 and the Bronze Medal in Sigtuna, Sweden, in 1991.

An Vo received her B.S and M.S degrees in Electrical Engineering from HCMC University of Technology, Viet Nam, in 1997 and 2000, respectively and Ph.D degree in Electrical Engineering from the University of Texas at Arlington, USA, in 2008.

She worked in the HCMC University of Technology, Viet Name, as a lecturer in the Department of Electrical Engineering from 1997 to 2004.

She is currently a Postdoctoral Research Fellow with the Feinstein Institute for Medical Research at North Shore LIJ, New York, USA. Her researches focus on complex wavelet transforms, statistical image modeling, pattern recognition and applications in digital signal/image processing, bioinformatics, biomedical images.