12th International Conference on Information Fusion
Seattle, WA, USA, July 6-9, 2009

Comparison of Raman Spectra Estimation Algorithms

Mahendra Mallick[a], Barry Drake[a], Haesun Park[b], Andy Register[a], Dale Blair[a], Phil West[a]
Ryan Palkki[b], Aaron Lanterman[b]
Darren Emge[c]

[a]Georgia Tech Research Institute
Sensors and Electromagnetic Applications Laboratory
[a,b]Georgia Institute of Technology
Atlanta, GA 30332, U.S.A.

[c]Edgewood Chemical Biological Center,
Aberdeen Proving Ground, MD 21010, U.S.A.

{mahendra.mallick, barry.drake, andy.register, dale.blair, phil.west}@gtri.gatech.edu
hpark@cc.gatech.edu, {palkki, lanterman}@ece.gatech.edu, darren.emge@us.army.mil

Abstract – *Raman spectroscopy is a powerful and effective technique for analyzing and identifying the chemical composition of a substance. Two types of Raman spectra estimation algorithms exist: supervised and unsupervised. In this paper, we perform a comparative analysis of five supervised algorithms for estimating Raman spectra. We describe a realistic measurement model for a dispersive Raman measurement device and observe that the measurement error variances vary significantly with bin index. Monte Carlo analyses with simulated measurements are used to calculate the bias, root mean square error, and computational time for each algorithm. Our analyses show that it is important to use correct measurement weights and enforce the nonnegative constraint in parameter estimation.*

Keywords: Chem/Bio Detection, Raman Spectroscopy, Machine Learning, Classification, Constrained Parameter Estimation, Classical Weighted Least Squares, Nonnegative Weighted Least Square, Generalized Likelihood Ratio Test, Measures of Performance.

1 Introduction

The Raman effect or Raman scattering represents the inelastic quantum scattering of a photon by molecules in liquids, gases, or solids [2-3]. When light is incident on a molecule, most photons are scattered elastically so that the energy or frequency of the scattered photon is the same as that of the incident photon. This is known as the Rayleigh scattering. A small fraction (about one in a million) is scattered inelastically, causing the frequency of the scattered photon to be different (usually lower) from the frequency of the incident photon. This is known as Raman scattering. The frequency change is due to the change in energy levels of the vibrational or rotational energy of the molecule. Therefore, Raman spectroscopy is a powerful tool for analyzing the chemical composition of liquids, gases, or solids using a laser [2], [13-15], [12].

A Raman spectrum is a plot of the intensity of the scattered photon as a function of frequency shift. The measured Raman spectrum can be used as a fingerprint to uniquely identify the chemical composition of a substance. Application of Raman spectroscopy to analyze chemical compositions of various substances has seen a rapid growth in recent years [2], [13-15], [12]. This is primarily due to the development of inexpensive and effective lasers and charge-coupled device (CCD) detectors [2]. Raman spectroscopy is also popular because measurement collection is fast and does not require contact with the chemical substance.

Suppose we have the measured Raman spectrum of a substance and we are interested in determining the chemical composition of the substance. The measured spectrum contains various error sources. Therefore, it is necessary to use a statistical measurement model that expresses the measurement as a function of the true spectrum and dominant error sources.

Raman spectrum estimation algorithms are of two types, *supervised* and *unsupervised* machine learning algorithms [11]. In the supervised approach, a library of reference Raman spectra are used and the true target spectrum is expressed as a linear combination of the reference spectra. Each reference spectrum is assumed to be error-free. In practice, this is not feasible. If the errors in a measured reference spectrum are very small compared with the signal values, then it is a good approximation to treat the measured reference spectrum as error-free. Otherwise, one must model the errors in the reference spectra. Supervised algorithms assume that the library contains all reference spectra that may be encountered in data collection. A supervised algorithm estimates the nonnegative expansion coefficients or mixing coefficients using the reference spectra and a statistical measurement model. The unsupervised approach estimates the spectra and mixing coefficients directly from measurements.

978-0-9824438-0-4 ©2009 ISIF

Report Documentation Page	*Form Approved* *OMB No. 0704-0188*

1. REPORT DATE **JUL 2009**	2. REPORT TYPE	3. DATES COVERED **06-07-2009 to 09-07-2009**

4. TITLE AND SUBTITLE
Comparison of Raman Spectra Estimation Algorithms

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Georgia Tech Research Institute,Sensors and Electromagnetic Applications Laboratory,Georgia Institute of Technology,Atlanta,GA,30332

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
See also ADM002299. Presented at the International Conference on Information Fusion (12th) (Fusion 2009). Held in Seattle, Washington, on 6-9 July 2009. U.S. Government or Federal Rights License.

14. ABSTRACT

Raman spectroscopy is a powerful and effective technique for analyzing and identifying the chemical composition of a substance. Two types of Raman spectra estimation algorithms exist: supervised and unsupervised. In this paper, we perform a comparative analysis of five supervised algorithms for estimating Raman spectra. We describe a realistic measurement model for a dispersive Raman measurement device and observe that the measurement error variances vary significantly with bin index. Monte Carlo analyses with simulated measurements are used to calculate the bias root mean square error, and computational time for each algorithm. Our analyses show that it is important to use correct measurement weights and enforce the nonnegative constraint in parameter estimation.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Public Release**	**8**	

This paper examines estimation of Raman spectra using the supervised approach and performs a comparative analysis of five Raman spectra estimation algorithms:

(i) classical weighted least squares (CWLS),

(ii) nonnegative weighted least squares (NNWLS),

(iii) fast combinatorial NNWLS (FCNNWLS),

(iv) block pivoting NNWLS (BPNNWLS), and

(v) NNLS or NNWLS using generalized likelihood ratio test (GLRT).

We use simulated data and perform Monte Carlo simulations to calculate measures of performance (MoP) for each algorithm. MoP used in this study include bias in the estimator, root mean square error (RMSE), bias in the measurement residual, RMSE for the measurement residual, and computation time.

These algorithms are implemented in a software simulation benchmark system designed specifically for analysis of detection algorithms. Each algorithm is inserted into the benchmark software using a well defined interface. The benchmark's native language is MATLAB®. For a fair comparison of run times, the algorithms were all coded in MATLAB®. Nothing precludes an algorithm from being implemented using a language that can be incorporated into the MATLAB® environment, i.e., Java, C/C++, or FORTRAN. However, for purposes of comparison, MATLAB® was used for all algorithms.

The outline of the paper is as follows. Sections 2 and 3 describe the measurement model and measurement function for Raman spectra, respectively. We summarize various Raman spectra estimation algorithms in Section 4. Finally, Sections 5 and 6 present numerical results and conclusions.

2 Measurement Model for Raman Spectrum

The Raman spectroscopy sensor system transmits a laser pulse and produces a measured Raman spectrum from the energy scattered by the chemical substance. The spectrum is spread across the bins of a CCD detector. The response on each bin corresponds to the amount of energy scattered at a particular frequency or wave number.

Let $\mathbf{y} \in \Re^M$ denote a measured spectrum with values at M bins

$$\mathbf{y} := [y_1 \quad y_2 \quad \dots \quad y_M], \tag{1}$$

where ":=" is used to define a quantity. The measurement model [16-17], [11] for each element of $\mathbf{y}$ is described by

$$y_i = n_i^{\mathrm{s}} + n_i^{\mathrm{b}} + g_i, \;\; i = 1,2,\dots,M, \tag{2}$$

where $n_i^{\mathrm{s}}, n_i^{\mathrm{b}}$, and g_i represent the signal, background noise, and Gaussian noise, respectively. The variables n_i^{s} and n_i^{b} are modeled as discrete random variables (RVs) whereas g_i is a continuous RV. We assume that n_i^{s}, n_i^{b}, and g_i are independent. The noise g_i is introduced by the on-chip amplifier and is modeled as Gaussian with mean m and variance σ^2

$$g_i \sim N(g_i; m, \sigma^2), \tag{3}$$

$$E\{(g_i - m)(g_j - m)\} = \delta_{ij}\sigma^2. \tag{4}$$

The background noise n_i^b is Poisson distributed

$$n_i^{\mathrm{b}} \sim p_{\mathrm{Poisson}}(n_i^{\mathrm{b}}; \lambda_i^{\mathrm{b}}), \tag{5}$$

where λ_i^{b} represents the expected number of counts for the background noise

$$E\{n_i^{\mathrm{b}}\} = \lambda_i^{\mathrm{b}}, \tag{6}$$

and

$$p_{\mathrm{Poisson}}(x; \lambda) = \frac{e^{-\lambda}\lambda^x}{x!}, \quad x = 0,1,2,\dots \tag{7}$$

We note that the variance of the Poisson distribution $p_{\mathrm{Poisson}}(x;\lambda)$ is also λ. The signal n_i^{s} is Poisson distributed with parameter λ_i^{s}

$$n_i^{\mathrm{s}} \sim p_{\mathrm{Poisson}}(n_i^{\mathrm{s}}; \lambda_i^{\mathrm{s}}), \tag{8}$$

where λ_i^{s} represents the expected number of counts for the signal. Since n_i^{s} and n_i^{b} are assumed to be independent, $n_i^{\mathrm{s}} + n_i^{\mathrm{b}}$ is Poisson distributed with mean and variance of $\lambda_i^{\mathrm{s}} + \lambda_i^{\mathrm{b}}$

$$n_i^{\mathrm{s}} + n_i^{\mathrm{b}} \sim p_{\mathrm{Poisson}}(n_i^{\mathrm{s}} + n_i^{\mathrm{b}}; \lambda_i^{\mathrm{s}} + \lambda_i^{\mathrm{b}}). \tag{9}$$

The vector measurement model is

$$\mathbf{y} = \mathbf{n}^{\mathrm{s}} + \mathbf{n}^{\mathrm{b}} + \mathbf{g}, \tag{10}$$

where $(\mathbf{n}^{\mathrm{s}}, \mathbf{n}^{\mathrm{b}}, \mathbf{g}) \in \Re^M$ are defined similarly as in (1).

Large Signal Approximation

If $\lambda_i^{\mathrm{s}} + \lambda_i^{\mathrm{b}}$ is large, then $n_i^{\mathrm{s}} + n_i^{\mathrm{b}}$ is well approximated by Gaussian distributions

$$n_i^{\mathrm{s}} + n_i^{\mathrm{b}} \sim N(n_i^{\mathrm{s}} + n_i^{\mathrm{b}}; \lambda_i^{\mathrm{s}} + \lambda_i^{\mathrm{b}}, \lambda_i^{\mathrm{s}} + \lambda_i^{\mathrm{b}}). \tag{11}$$

Using (2) and (11)

$$y_i \sim N(n_i^{\mathrm{s}} + n_i^{\mathrm{b}} + g_i; \lambda_i^{\mathrm{s}} + \lambda_i^{\mathrm{b}} + m, \lambda_i^{\mathrm{s}} + \lambda_i^{\mathrm{b}} + \sigma^2). \quad (12)$$

Alternatively,

$$y_i = \lambda_i^{\mathrm{s}} + \lambda_i^{\mathrm{b}} + m + v_i, \quad (13)$$

$$v_i \sim N(0, \lambda_i^{\mathrm{s}} + \lambda_i^{\mathrm{b}} + \sigma^2). \quad (14)$$

Using the large signal approximation, (13) can be written as

$$\mathbf{y} = \boldsymbol{\lambda}^{\mathrm{s}} + \boldsymbol{\lambda}^{\mathrm{b}} + \mathbf{m} + \mathbf{v}, \quad (15)$$

where

$$\mathbf{m} := m[1 \quad 1, \ldots, 1]', \quad (16)$$

$$\boldsymbol{\lambda}^{\mathrm{s}} := [\lambda_1^{\mathrm{s}} \quad \lambda_2^{\mathrm{s}} \ldots, \lambda_M^{\mathrm{s}}]', \quad (17)$$

$$\boldsymbol{\lambda}^{\mathrm{b}} := [\lambda_1^{\mathrm{b}} \quad \lambda_2^{\mathrm{b}} \ldots, \lambda_M^{\mathrm{b}}]', \quad (18)$$

$$\mathbf{v} \sim N(\mathbf{v};\ \mathbf{0}_{M\times 1}, \mathbf{R}), \quad (19)$$

$$\mathbf{R} := \mathrm{diag}(\lambda_1^{\mathrm{s}} + \lambda_1^{\mathrm{b}} + \sigma^2, \ldots, \lambda_M^{\mathrm{s}} + \lambda_M^{\mathrm{b}} + \sigma^2). \quad (20)$$

3 Measurement Function for Raman Spectra

Suppose we have N reference spectra $\{\mathbf{s}_j \in \Re^M\}_{j=1}^N$ in our library corresponding to N chemical substances. Then the true target spectra $\mathbf{s}$ can be expressed as a linear combination of the reference spectra by

$$\mathbf{s} = \sum_{j=1}^{N} x_j \mathbf{s}_j. \quad (21)$$

We can write (21) in the matrix form

$$\mathbf{s} = \mathbf{A}\mathbf{x}, \quad (22)$$

where

$$\mathbf{x} := [x_1 \quad x_2 \quad \ldots \quad x_N]', \quad (23)$$

$$x_j \geq 0, \quad j = 1, 2, \ldots, N, \quad (24)$$

$$\mathbf{A} := [\mathbf{s}_1 \quad \mathbf{s}_2 \quad \ldots \quad \mathbf{s}_N]. \quad (25)$$

Then

$$\boldsymbol{\lambda}^{\mathrm{s}} = \mathbf{P}\mathbf{s}, \quad (26)$$

where $\mathbf{P}$ is the $M \times M$ *point spread function matrix* of the diffraction grating used to spread the spectral energy across the CCD's bins. Substitution of (22) in (26) gives

$$\boldsymbol{\lambda}^{\mathrm{s}} = \boldsymbol{\Phi}\mathbf{x}, \quad (27)$$

where

$$\boldsymbol{\Phi} = \mathbf{P}\mathbf{A}. \quad (28)$$

Not all photons that hit the CCD array are converted to photoelectrons. The quantum efficiency or flat-field response varies along the CCD array. This non-uniform detector efficiency is modeled by

$$\lambda_i^{\mathrm{s}} = \beta_i (\boldsymbol{\Phi}\mathbf{x})_i, \quad (29)$$

where β_i is known from calibration measurements. We can write (29) in the matrix form

$$\boldsymbol{\lambda}^{\mathrm{s}} = \mathbf{C}\,\mathbf{x}, \quad (30)$$

where

$$\mathbf{C} := \mathbf{B}\boldsymbol{\Phi} = \mathbf{B}\mathbf{P}\mathbf{A}, \quad (31)$$

$$\mathbf{B} := \mathrm{diag}(\beta_1, \beta_1, \ldots, \beta_M). \quad (32)$$

Under the large signal approximation, substitution of (30) in (15) gives

$$\mathbf{y} = \mathbf{C}\,\mathbf{x} + \boldsymbol{\lambda}^{\mathrm{b}} + \mathbf{m} + \mathbf{v}. \quad (33)$$

Define the new measurement vector $\mathbf{z}$

$$\mathbf{z} := \mathbf{y} - \boldsymbol{\lambda}^{\mathrm{b}} - \mathbf{m}. \quad (34)$$

Then

$$\mathbf{z} = \mathbf{C}\,\mathbf{x} + \mathbf{v}. \quad (35)$$

Thus, under the large signal approximation, the measurement model is linear with additive Gaussian measurement noise. An estimate $\hat{\mathbf{x}}$ of $\mathbf{x}$ can be obtained using the maximum likelihood estimator (MLE) [4], [10] or weighted least squares (WLS) [4], [10] with the nonnegativity constraint (24). Thus, the estimation problem is a constrained estimation problem due to (24). Use of classical MLE or WLS would yield approximate results.

4 Raman Spectra Estimation Algorithms

This paper addresses estimation of Raman spectra under the large signal assumption. Future work will address the more general case where the large signal assumption is not valid.

Since the measurement model in (35) is linear with additive Gaussian noise, the estimates from the MLE and WLS are the same provided the weight matrix $\mathbf{W}$ in WLS is equal to $\mathbf{R}^{-1}$ [4], [10]. Then using (20),

$$\mathbf{W} := \mathrm{diag}(w_1, w_2, \ldots, w_M), \quad (36)$$

$$w_i := 1/(\lambda_i^{\mathrm{s}} + \lambda_i^{\mathrm{b}} + \sigma^2), \quad i = 1, 2, \ldots, M. \quad (37)$$

The true weights in (37) are not known and must be estimated. Our future work will address this issue.

The cost function for the parameter estimation problem is

$$J(\mathbf{x}) := (\mathbf{z} - \mathbf{C}\,\mathbf{x})'\mathbf{W}(\mathbf{z} - \mathbf{C}\,\mathbf{x}). \qquad (38)$$

We can rewrite (38) as

$$J(\mathbf{x}) := (\boldsymbol{\eta} - \mathbf{D}\,\mathbf{x})'(\boldsymbol{\eta} - \mathbf{D}\,\mathbf{x}), \qquad (39)$$

where the weighted measurement vector $\boldsymbol{\eta} \in \Re^{M}$ and weighted measurement matrix $\mathbf{D} \in \Re^{M \times N}$ are defined by

$$\eta_i := \sqrt{w_i}\, z_i, \quad i = 1,2,..., M, \qquad (40)$$

$$d_{ij} := \sqrt{w_i}\, c_{ij}, \quad i = 1,2,..., M,\ j = 1,2,\ldots, N. \qquad (41)$$

4.1 Classical Weighted Least Squares

Classical WLS (CWLS) [4], [10] solves the following problem without the nonnegative constraint (24)

$$\min_{\mathbf{x}} J(\mathbf{x}). \qquad (42)$$

Since the CWLS does not enforce the nonnegative constraint, the estimate obtained using the CWLS is expected to have lower accuracy compared to the nonnegative WLS (NNWLS).

4.2 Nonnegative Weighted Least Squares

For the current measurement model (35), the NNWLS or nonnegative MLE (NNMLE) solves the problem

$$\min_{\mathbf{x} \geq 0} J(\mathbf{x}). \qquad (43)$$

There are three commonly used algorithms for solving the NNWLS problem in (43). These algorithms were proposed by Lawson and Hanson [9] (LHNNLS), Bro and De Jong [1], and Van Benthem and Keenan [18] (VKNNLS). Although these three algorithms considered the same measurement variances for all measurements, i.e. $\mathbf{W} = \sigma_v^2 \mathbf{I}$, they can be easily modified to handle non-uniform weights. The latter two algorithms are improvements over the original algorithm of Lawson and Hanson for handling multiple measurement vectors, $\mathbf{z}_k \in \Re^{M}$, $k = 1,2,\ldots, K$. It can also be shown that for multiple measurement vectors $\{\mathbf{z}_k \in \Re^{M}\}$, solving the problem by processing one measurement vector at a time is much less efficient than collecting a number of measurement vectors and processing them at once using a *column parallel* algorithm [18]. Next, we summarize the three NNWLS algorithms.

4.2.1 Lawson-Hanson Algorithm

The standard algorithm for computing $\hat{\mathbf{x}}_{\text{NNWLS}}$ is that of Lawson and Hanson [9], which is included in the MATLAB® function, "lsqnonneg."

Algorithm 1 LHNNWLS [9]

```
Let $S_{\text{p}}$ and $S_{\text{a}}$ denote the passive and active index sets, respectively.
Given $\mathbf{D}, \boldsymbol{\eta}$, find $\hat{\mathbf{x}}$ that solves (43) with (39).
    1. Set $S_{\text{p}}$ = NULL , $S_{\text{a}} = \{1, 2 \ldots N\}$, and $\mathbf{x} = \mathbf{0}$.
    2. Compute the N-vector $\mathbf{w} = \mathbf{D}'(\boldsymbol{\eta} - \mathbf{Dx})$.
    3. If the set $S_{\text{a}}$ is empty or if $w_j \leq 0$ for all $j \in S_{\text{a}}$,
         then go to Step 12.
    4. Find an index $t \in S_{\text{a}}$ such that
         $w_t = \max\{w_j : j \in S_{\text{a}}\}$.
    5. Move the index t from set $S_{\text{a}}$ to set $S_{\text{p}}$.
    6. Let $\mathbf{D}_{\text{p}}$ denote the $m_2 \times N$ matrix defined by
         Column j of $\mathbf{D}_{\text{p}} = \begin{Bmatrix} \text{column } j \text{ of } \mathbf{D} \text{ if } j \in S_{\text{p}} \\ 0 \quad \text{if } j \in S_{\text{a}} \end{Bmatrix}$.
         Compute the N-vector $\boldsymbol{\eta}$, as a solution of the CLS problem $\mathbf{D}_{\text{p}}\mathbf{x} \cong \boldsymbol{\eta}$.
         Note that only the components $x_j, j \in S_{\text{p}}$, are determined by this problem.
         Define $x_j = 0$ for $j \in S_{\text{a}}$.
    7. If $x_j > 0$ for all $j \in S_{\text{p}}$, set $\mathbf{x} = \boldsymbol{\eta}$ and go to Step 2.
    8. Find an index
         $q \in S_{\text{p}}$ such that $x_q/(x_q - \eta_q) =$
         $\min\{x_j/(x_j - x_j) : x_j \leq 0, j \in S_{\text{p}}\}$.
    9. Set $\alpha = x_q/(x_q - x_q)$.
    10. Set $\mathbf{x} = \mathbf{x} + \alpha(\boldsymbol{\eta} - \mathbf{x})$.
    11. Move from set $S_{\text{p}}$ to set $S_{\text{a}}$ all indices $j \in S_{\text{p}}$ for
         which $x_j = 0$. Go to step 6.
    12. Computation completed
$\hat{\mathbf{x}}_{\text{NNWLS}} = \mathbf{x}$ is the solution such that: $\begin{Bmatrix} x_j > 0, j \in S_{\text{p}} \\ x_j = 0, j \in S_{\text{a}} \end{Bmatrix}$.
```

End Algorithm 1 LHNNLS

This algorithm has some disadvantages for multiple measurement vectors because of the repeated solution of the CLS problem $\mathbf{D}_{\text{p}}\mathbf{x} \cong \boldsymbol{\eta}$ in Step 6 within each of the iterations.

4.2.2 Fast Combinatorial NNWLS Algorithms (FCNNWLS)

The efficiency of the NNWLS can be improved by processing multiple measurement vectors in a block, $(\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_k)$, for unweighted measurements, and $(\boldsymbol{\eta}_1, \boldsymbol{\eta}_1, \ldots \boldsymbol{\eta}_k)$, for weighted measurements. For multiple measurement vectors, define the weighted measurement matrix **G** and parameter matrix **X**

$$\mathbf{G} := [\boldsymbol{\eta}_1 \ \ \boldsymbol{\eta}_2 \cdots \boldsymbol{\eta}_k], \tag{44}$$

$$\mathbf{X} := [\mathbf{x}_1 \ \ \mathbf{x}_2 \cdots \mathbf{x}_k]. \tag{45}$$

Then the NNWLS problem for multiple measurement vectors is

$$\min_{\mathbf{X} \geq 0} \left\| \mathbf{DX} - \mathbf{G} \right\|_F^2, \tag{46}$$

where F in (46) represents the Frobenius norm [9]. This is possible since common least squares (LS) problems across multiple observation vectors can be processed simultaneously. The resulting modification to LHNNLS is referred to as the *column parallel* form of processing the passive set associated with multiple measurement vectors. The redundant computations can be taken advantage of and the pseudoinverse [9] of $\mathbf{D}_p$ in Step 6 can be computed for some of the multiple measurement vectors with common structure [6,7,18]. All measurement vectors that correspond to a common pseudoinverse are grouped together. Thus, the pseudoinverse is computed only once for each group of measurement vectors sharing this common pseudoinverse.

Modifications to algorithm LHNNLS are made to the CLS part of the code in Step 6. These entail finding the multiple measurement vectors with a common pseudoinverse and solving the CLS problem. This pseudoinverse may be used in subsequent computations as well. As an example, given 3 measurement vectors, the number of pseudoinverse computations may be reduced from 7 to 4 (see [20] for details). Two pseudoinverse computations are common to two of the three iterations and one of those is used twice in the second iteration. The sequence is

1. {1, 1, 1}
2. {2, 3, 2}
3. {2, 3, 4}

Where the same pseudoinverse is applied to all three columns in step 1, one less pseudoinverse is required in step 2, and two of those are reused in step 3. The savings in computations will become even more significant as the number of measurement vectors becomes larger.

4.2.3 Block Pivoting NNWLS (BPNNWLS)

Further computational savings can be realized by generalizing the active set method of LHNNLS, which is a single principal pivoting algorithm, to a block principal pivoting algorithm [8]. This method modifies LHNNLS using a block exchange rule to move blocks of columns from the "passive" to "active" sets. The quotes are due to the fact that the sets employed in the block principal pivoting method do not necessarily correspond to the active and passive sets of algorithm LHNNLS.

4.2.4 Weighted Generalized Likelihood Ratio Test (WGLRT) Algorithm

The subspace version of the generalized likelihood ratio test (GLRT) [5] has been applied to analyze Raman spectroscopy data [14-15]. In this formulation, the likelihood function $p(\mathbf{z}; \hat{\mathbf{x}}, \mathbf{R}, H_1)$ for the measurement model (35) is calculated under the hypothesis H_1, where all reference spectra are included in constructing **C** and $\hat{\mathbf{x}}$ is the NNWLS estimate. Then the i^{th} reference spectra is removed and the likelihood function $p(\mathbf{z}; \hat{\mathbf{x}}_0, \mathbf{R}_0, H_0)$ for the measurement model

$$H_0: \quad \mathbf{z} = \mathbf{C}_0\,\mathbf{x}_0 + \mathbf{v}_0, \tag{49}$$

$$\mathbf{v}_0 \sim N(\mathbf{v}_0; \mathbf{0}, \mathbf{R}_0), \tag{50}$$

is calculated under the hypothesis H_0, where $\mathbf{C}_0$ is the corresponding measurement matrix and $\hat{\mathbf{x}}_0$ is the NNWLS estimate. If the log-likelihood ratio exceeds a threshold, i.e.,

$$\ln p(\mathbf{z}; \hat{\mathbf{x}}, \mathbf{R}, H_1) - \ln p(\mathbf{z}; \hat{\mathbf{x}}_0, \mathbf{R}_0, H_0) > \alpha, \tag{51}$$

then the i^{th} substance is assumed to contribute to the target substance. This process is repeated for each reference spectra and the indices $\{i_1, i_2, \ldots, i_p\}$ are determined for which the test (51) succeeds. Using these indices, the measurement matrix $\mathbf{C}_p$ is formed and the NNWLS estimate $\hat{\mathbf{x}}_p$ is calculated.

5 Numerical Simulation and Results

We used 67 reference Raman spectra, $\{\mathbf{s}_j \in \Re^M\}_{j=1}^N$, N=67. All the components of **x** were zero except $x_{60} = 1.0$. Each spectrum has values at M = 1024 bins. In the Monte Carlo simulations, the mean and variance of the Gaussian measurement noise are 10 and 225, respectively. We used a constant value of 256 for the Poisson parameter λ_i^{b} for all bin values. We then calculated $\boldsymbol{\lambda}^{\text{s}}$ by selecting and substituting a true **x** vector into (30). Figure 1 shows the variation of the measurement error variance with bin index for the current scenario. We observe that the measurement error variance changes significantly with the bin index.

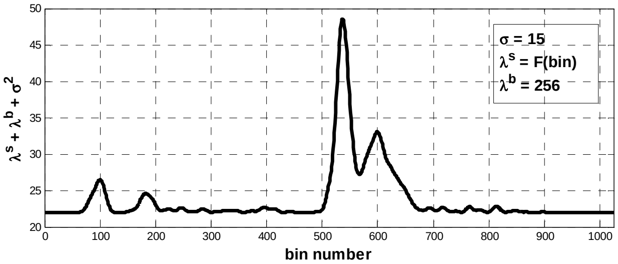

Figure 1. Variation of measurement error variance with bin index.

One hundred Monte Carlo trials were used to calculate measures of performance (MoP) for each spectral estimation algorithm. The equations and resulting MoP results are presented here.

Let M_s be the total number of Monte Carlo simulations and $\hat{\mathbf{x}}_m$ the estimate of $\mathbf{x}$ in the m^{th} Monte Carlo simulation. The estimation error in the j^{th} component of $\mathbf{x}$ in the m^{th} Monte Carlo simulation is defined by

$$\tilde{x}_{m,j} := x_j - \hat{x}_{m,j}, \quad j = 1,2,\ldots,N. \tag{47}$$

The bias error for the j^{th} coefficient and overall bias error for the coefficients are defined, respectively, by

$$b_{x,j} := \frac{1}{M_s}\sum_{m=1}^{M_s} \tilde{x}_{m,j}, \quad j = 1,2,\ldots,N, \tag{48}$$

$$b_x := \frac{1}{NM_s}\sum_{j=1}^{N}\sum_{m=1}^{M_s} \tilde{x}_{m,j}. \tag{49}$$

The root mean square error (RMSE) for the j^{th} coefficient and the overall RMSE for the coefficients are defined, respectively, by

$$\text{RMSE}_{x,j} := \left[\frac{1}{M_s}\sum_{m=1}^{M_s} \tilde{x}_{m,j}^2\right]^{1/2}, \quad j = 1,2,\ldots,N, \tag{50}$$

$$\text{RMSE}_x := \left[\frac{1}{NM_s}\sum_{j=1}^{N}\sum_{m=1}^{M_s} \tilde{x}_{m,j}^2\right]^{1/2}. \tag{51}$$

Let $\hat{z}_{m,i}$ denote the predicted measurement at the i^{th} bin in the m^{th} Monte Carlo simulation. Then

$$\hat{z}_{m,i} := (\mathbf{C}\hat{\mathbf{x}}_m)_i, \quad i = 1,2,\ldots,M. \tag{52}$$

The measurement residual at the i^{th} bin in the m^{th} Monte Carlo simulation is defined by

$$\tilde{z}_{m,i} := z_{m,i} - \hat{z}_{m,i}, \quad i = 1,2,\ldots,M. \tag{53}$$

The bias error of the measurement residual at the i^{th} bin and the overall bias of the measurement residual are defined, respectively, by

$$b_{z,i} := \frac{1}{M_s}\sum_{m=1}^{M_s} \tilde{z}_{m,i}, \quad i = 1,2,\ldots,M, \tag{54}$$

$$b_z := \frac{1}{MM_s}\sum_{i=1}^{M}\sum_{m=1}^{M_s} \tilde{z}_{m,i}. \tag{55}$$

The RMSE of the measurement residual at the i^{th} bin and the overall RMSE for the measurement residual are defined, respectively, by

$$\text{RMSE}_{z,i} := \left[\frac{1}{M_s}\sum_{m=1}^{M_s} \tilde{z}_{m,i}^2\right]^{1/2}, \quad i = 1,2,\ldots,M. \tag{56}$$

$$\text{RMSE}_z := \left[\frac{1}{MM_s}\sum_{i=1}^{M}\ \sum_{m=1}^{M_s} \tilde{z}_{m,i}^2\right]^{1/2}. \tag{57}$$

Table 1 summarizes unweighted bias and RMS error results for both parameter estimation and measurement residual using 100 Monte Carlo trials and one measurement vector. Recall that unweighted refers to the use of a weight matrix equal to the identity matrix.

Table 1. Overall errors for unweighted versions of the algorithms.

Algorithm	Estimation		Residual	
	Bias	RMSE	Bias	RMSE
CLS	-1.5904e-5	0.0928	-0.1041	7.4593
NNLS	1.4883e-4	0.0031	0.9737	3.1252
FCNNLS	1.4883e-4	0.0031	0.9737	3.1252
BPNNLS	1.4883e-4	0.0031	0.9737	3.1252
NNGLRT	8.8715e-13	4.6835e-4	-1.04e-6	0.7288

Table 2 summarizes the bias and RMS error results when the algorithms use the weight matrix defined in (36-37). Comparing the results between Table 1 and Table 2 shows a significant reduction in the measurement residual bias error and RMSE when the weight matrix is used.

Table 2. Overall errors for weighted versions of the algorithms.

Algorithm	Estimation		Residual	
	Bias	RMSE	Bias	RMSE
CWLS	-1.6241e-5	0.0918	-0.0053	0.8240
NNWLS	1.1793e-4	0.0028	0.0421	0.1801
FCNNWLS	1.1793e-4	0.0028	0.0421	0.1801
BPNNWLS	1.1793e-4	0.0028	0.0421	0.1801
NNWGLRT	1.102e-14	4.322e-4	0.0018	0.0256

The results for parameter estimation are nearly the same for the first four algorithms. In addition, the NNWGLRT algorithm has the best performance.

Tables 3 and 4 show the average CPU times for 100 Monte Carlo trials. The LS-based methods have lower CPU times than the likelihood-based method. For multiple measurement vectors the block pivoting method outperformed the other constrained least squares methods. The CLS method is faster but the estimation and residual errors are very large compared to the constrained methods.

Table 3. Average CPU time over 100 Monte Carlo trials.

Algorithm	CPU Time (seconds)	
	Unweighted	Weighted
CLS	3.18	3.26
NNLS	3.20	3.23
FCNNLS	3.76	3.80
BPNNLS	3.32	3.34
NNGLRT	11.68	11.77

Table 4. Average CPU time over 100 Monte Carlo trials. Algorithms used to process multiple measurement vectors.

Algorithm	CPU Time (seconds)		
	Unweighted	Weighted	
	20 MVs	20 MVs	100 MVs
CLS	2.64	2.69	2.84
NNLS	4.89	5.07	11.08
FCNNLS	5.54	5.67	13.02
BPNNLS	3.31	3.38	5.90

These results demonstrate that great care needs to be taken when specifying the measurement model since this impacts the formulation of the detection algorithms. As can be seen from the tables and the graphs, the residual errors can be reduced significantly with the proper measurement model and measurement error covariance. Note the decrease in the residual error bias and RMSE in Tables 1 and 2. Plotting MoP values versus bin index averaged over the 100 Monte Carlo trials in Figures 2, 3, and 4 clearly illustrates the decrease.

Figure 2 compares the bias errors of measurement residuals for the CLWS, NNWLS, and NNWGLRT algorithms. The NNWGLRT algorithm has nearly zero bias.

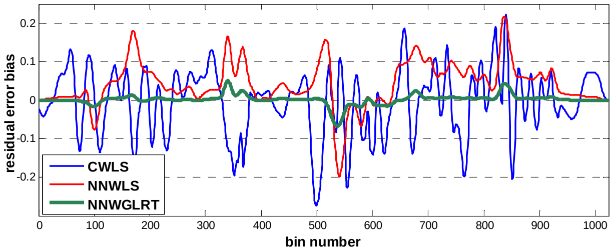

Figure 2. CWLS, NNWLS and NNWGLRT residual error bias.

In Figure 3, we observe that CLS has the highest RMSE for measurement residual and NNWLS has the lowest. We also note that CWLS outperforms NNLS. This observation underscores the importance of including error sources in the measurement model. Figure 4 shows that both the NNWLS and NNWGLRT exhibit good performance with respect to residual error RMSE. The NNWGLRT results exhibit almost no residual error. It is also interesting to note that the unweighted NNGLRT algorithm performs better than the NNLS.

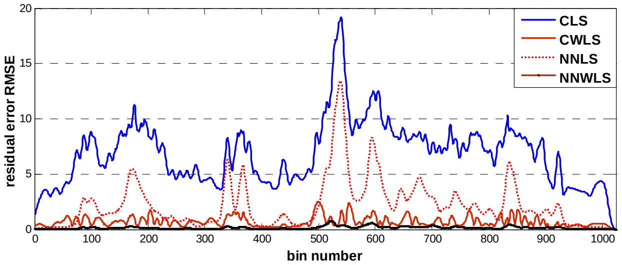

Figure 3. CLS, CWLS, NNLS, and NNWLS residual error RMSE.

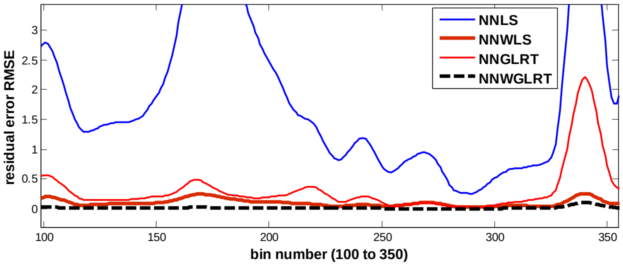

Figure 4. NNLS, NNWLS, NNGLRT, and NNWGLRT residual error RMSE zoomed to bins 100 to 350.

Figures 5 and 6 show the bias and RMSEs for parameter estimation using the CWLS, NNWLS, and NNWGLRT algorithms. We observe that the use of nonnegative constraints significantly improves the accuracies of parameter estimation.

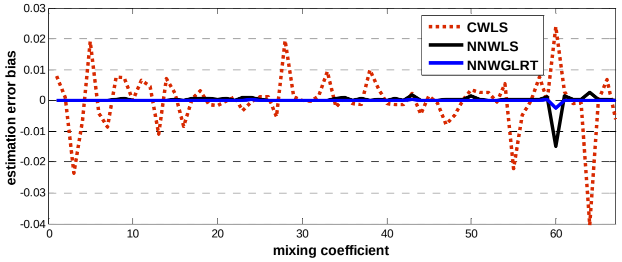

Figure 5. Estimation error bias.

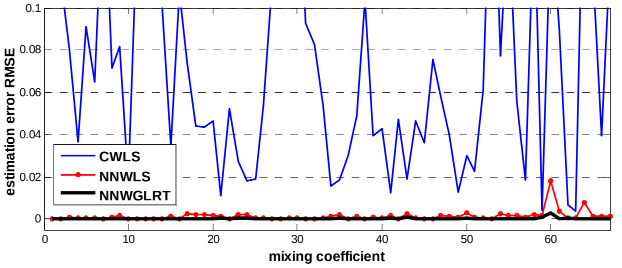

Figure 6. Parameter estimation RMSE.

6 Conclusions

This paper compared five supervised algorithms for estimating Raman spectra in the large signal domain. In this domain, the measurement model can be approximated as being linear with additive Gaussian noise. The measurement error variances vary significantly with bin index. The parameter estimation problem is constrained because the elements of the parameter vector must be nonnegative. We have presented numerical results for the bias error and RMSE of the estimated parameter and measurement residual.

Since the measurement error variances vary significantly with bin index, it is important to use the correct weights or measurement error variances in parameter estimation. It is also important to enforce the nonnegativity constraint in the estimation of the mixing coefficients. The NNWLS, FCNNWLS, and BPNNWLS algorithms yield nearly the same accuracy and NNWGLRT has the best accuracy, but the worst computational performance. The CPU times of NNWLS, FCNNWLS, and BPNNWLS are also similar when only one measurement vector is processed at a time. When a block of data is processed together, BPNNWLS shows considerable computational advantage.

Our future work will focus on three areas. First, the more general case when the large signal approximation is not valid will be investigated. Second, a more computationally efficient NNGLRT algorithm will be developed. Third, a similar investigation and comparison with unsupervised algorithms will be carried out so that their potential for Chemical/Biological detection capability can be evaluated.

Acknowledgment

This work was supported in part by ONR Grants N00014-07-1-0378 and N00014-07-1-1074.

References

[1] R. Bro and S. De Jong, “A fast non-negativity-constrained least squares algorithm”, *J. Chemometrics*, Vol. 11, pp. 393–401, 1997.

[2] J. R. Ferraro, K. Nakamoto, and C. W. Brown, *Introductory Raman Spectroscopy*, 2nd ed. Academic Press, 2003.

[3] http://en.wikipedia.org/wiki/Raman_scattering.

[4] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Prentice Hall, 1993.

[5] S. M. Kay, *Fundamentals of Statistical Signal Processing: Detection Theory*. Prentice Hall, 1998.

[6] H. Kim and H. Park, “Non-negative Matrix Factorization based on Alternating Non-negativity Constrained Least Squares and Active Set Method” *SIAM J. on Matrix Analysis and Applications*, Vol 30, No. 2, 2008.

[7] H. Kim and H. Park, “Sparse Non-negative Matrix Factorizations via Alternating Non-negativity-constrained Least Squares for Microarray Data Analysis” *Bioinformatics*, Vol. 23, No. 12, 2007.

[8] J. Kim and H. Park, “Toward Faster Nonnegative Matrix Factorization: A New Algorithm and Comparisons” *Proceedings of the IEEE International Conference on Data Mining*, 2008.

[9] C. L. Lawson and R. J. Hanson, *Solving Least Squares Problems*, Prentice-Hall, 1974.

[10] J. M. Mendel, *Lessons in Estimation Theory for Signal Processing Communications, and Control*, Prentice-Hall, 1995.

[11] R. D. Palkki and A. D. Lanterman, “Algorithms and Performance Bounds for Chemical Identification under a Poisson Model for Raman Spectroscopy,” *Twelfth International Conference on Information Fusion*, Seattle, July 6-9, 2009, (accepted).

[12] P. Ponsardin, S. Higdon, T. Chyba, W. Armstrong, A. Sedlacek, S. Christesen and A. Wong, “Expanding applications for surface-contaminants sensing using the laser interrogation of surface agents (LISA) technique, ” Chemical and Biological Standoff Detection. *Proceedings of the SPIE*, Vol. 5268, pp. 321- 327 (2004).

[13] M-A. Slamani, T. Chyba, H. LaValley, and D.Emge, “Spectral unmixing of agents on surfaces for the Joint Contaminated Surface Detector (JCSD)” *Proceedings of the SPIE*, Vol. 6699, (2007).

[14] M-A Slamani, B. Fisk, and T. Chyba, D. Emge, and S. Waugh, “An algorithm benchmark data suite for chemical and biological (chem/bio) defense applications,” *Proc. Signal and Data Processing of Small Targets*, Vol. 6969, March 18-20, 2008, Orlando, FL, USA.

[15] A. Sedlacek, S. Christesen, T. Chyba, P. Ponsardin, “Application of UV-Raman spectroscopy to the detection of chemical and biological threats.” Chemical and Biological Point Sensors for Homeland Defense. *Proceedings of the SPIE*, Vol. 5269, pp. 23-33. (2004).

[16] D. L. Snyder, A. M. Hammoud, and R. L. White, "Image recovery from data acquired with a charge-coupled-device camera," *J. Opt. Soc. Am.*, A, Vol. 10, pp. 1014- 1023, May 1993.

[17] Snyder, D. L., Schultz, T., and O’Sullivan, J., “Deblurring subject to nonnegativity constraint,” *IEEE Trans. Signal Process.*, vol. 40, pp. 1143-1150, 1992.

[18] M. H. Van Benthem and M. R. Keenan, “Fast algorithm for the solution of large-scale non-negativity-constrained least squares problems” *J. Chemometrics*, Vol. 18, 2004.