Feature Selection Based on SVM in Photo-Thermal Infrared (IR) Imaging Spectroscopy Classification With Limited Training Samples

NIAN ZHANG and KEENAN LEATHAM
Department of Electrical and Computer Engineering, University of the District of Columbia
4200 Connecticut Avenue, NW, Washington, D.C. 20008 USA
nzhang@udc.edu, keenan.leatham@udc.edu

Abstract: - In this paper, we propose a kernel based SVM algorithm with variable models to adapt to the high-dimensional but relatively small samples for remote explosive detection on photo-thermal infrared imaging spectroscopy (PT-IRIS) classification. The algorithms of the representative linear and nonlinear SVM are presented. The response plot, predicted vs. actual plot, and residuals plot of the linear, quadratic, and coarse Gaussian SVM are demonstrated. A comprehensive comparison of Linear SVM, Quadratic SVM, Cubic SVM, Fine Gaussian SVM, Median Gaussian SVM, Coarse Gaussian SVM is performed in terms of root mean square error, R-squared, mean squared error, and mean absolute error. The excellent experimental results demonstrated that the kernel based SVM models provide a promising solution to high-dimensional data sets with limited training samples.

Key-Words: - Feature selection, Support vector machine, SVM, High-dimensional, Classification, Photo-thermal infrared imaging spectroscopy

1 Introduction

Recent advances in modern technologies, such as photo-thermal infrared (IR) imaging spectroscopy technology in the application of remote explosive detection, 4D CT-scans technology, and DNA microarrays have produced numerous massive and imbalanced data. The needs of classification ubiquitously exist in real-world data-intensive applications, ranging from civilian applications such as cancer diagnoses and outlier detection in stock market time series, to homeland security or defense related applications such as remote explosive detection, illegal drug detection, and abnormal behavior recognition.

In the situation when the dimensionality of data is high but with few data, feature selection usually becomes imperative to the learning algorithms because high-dimensional data tends to negatively affect the efficiency of most learning algorithms. Feature selection is an efficient dimensionality reduction technique that selects an optimal subset of the original features that provide the best predictive power in modeling the data. They are the most distinct features that can be used to differentiate samples into different classes.

There are a large number of state-of-the-art feature selection methods. A simultaneous spectral-spatial feature selection and extraction algorithm was proposed for hyperspectral images spectral-spatial feature representation and classification. However, it lacks of kernel version and thus its performance on complex datasets is unknow [1]. A regularized regression based feature selection classifier was modified into a cost-sensitive classifier by generating and assigning different costs to each class. Features will be selected according to the classifier with optimal F-measure in order to solve the class imbalance problem [2]. A feature selection algorithm using AdaBoost was presented to deal with Haar-like features for vehicle detection. The normalized feature vector set is used to train the RBF-SVM classifier with cross-validation to select the optimal parameters [3]. A support vector machine (SVM) was applied as a classifier to identify residual functional abnormalities in athletes suffering from concussion using a multichannel EEG data set. The total accuracy of the classifier using the 10 features was 77.1% [4]. A multiple instance learning (MIL) was adopted to describe different kinds of actions from complexity data sources

and present a boosted exemplar learning (BEL) method to learn the similarity metric and select some representative exemplars from the Web for action recognition. It takes about 98 *ms* to train a multiple instance SVM (mi-SVM) for one exemplar. The proposed mi-SVM has much better result 65.37% than the 61.53% using SVM classifier [5].

Different from the above SVM based learning algorithm, we propose a kernel based SVM algorithm with variable models to adapt to the high-dimensional but relatively small samples for remote explosive detection on photo-thermal infrared imaging spectroscopy (PT-IRIS) classification.

The rest of the paper is organized as follows. In Section 2, the SVM models including the linear SVM, quadratic SVM, and coarse Gaussian SVM are discussed. In Section 3, photo-thermal infrared imaging spectroscopy (PT-IRIS) data set is introduced. In Section 4, the classification performance of linear, quadratic, and coarse Gausian SVM are demonstrated. In addition, a comparison of Linear SVM, Quadratic SVM, Cubic SVM, Fine Gaussian SVM, Median Gaussian SVM, Coarse Gaussian SVM is presented in terms of various model statistics. Finally, the paper is concluded in Section 5.

2 Types of SVM Algorithms

Support Vector Machine (SVM) learning algorithms has been an active research topic within the computer intelligence community. Support vector machine (SVM) analysis is a popular machine learning tool for classification and regression, first identified by Vladimir Vapnik and his colleagues in 1992 [6]. SVM regression is considered a nonparametric technique because it relies on kernel functions. Support Vector Machine algorithms are utilized in many real world applications such as Face Detection, Text Categorization, and Bioinformatics. Support Vector Machine algorithm (SVM) is a supervised machine learning algorithm, which can be used for either classification or regression challenges. The different types of SVM learning algorithms are Linear SVM, Quadratic SVM, and Cubic SVM. Each Support Vector Machine Algorithm has their advantages in terms of providing solutions on a data set. For each algorithm we will be (1) Training a data set with Linear SVM, Quadratic SVM, and Cubic SVM, (2) Plotting the behavior of each algorithm figuring out the RSME, R-Squared Value, MSE, MAE, Prediction Speed, Training Time, and (3) Analyzing the results of each Support Machine Algorithm to see the similarities and differences of the data. The purpose of these trials is to see if we can find some interesting behaviors, so we can find different methods to optimize SVM algorithms. Shown below are the different behaviors of each SVM.

2.1 Linear SVM

Linear SVM is the newest fast machine learning data mining algorithm for solving multiclass classification problems from ultra large data sets; that implements an original proprietary version of a cutting plane algorithm for designing a linear support vector machine. Linear SVM is a linearly scalable routine meaning that it creates an SVM model in a CPU time, which scales the size of the training data set linearly. Our comparisons with other known SVM models clearly show its performance is highly accurate, implemented with large data sets.

It is ideal for a Linear SVM to be utilized in contemporary applications such as digital advertisement-commerce, web page categorization, text classification, bioinformatics, proteomics, banking services and many other areas. It provides solutions of multiclass classification problems with any number of classes with high dimensional data in both sparse and dense formats. There is no need for expensive computing resources other than a standard platform while implementing this algorithm.

The algorithm of the Linear SVM is illustrated as follows.

Algorithm of the Linear SVM
Input:
1. A training data set of the form: $(\vec{x}_1, y_1), \ldots, (\vec{x}_n, y_n)$
2. A y_i that is either 1 or -1.
Procedure:

1. Let the given training data set of n points be in the form of: $(\vec{x}_1, y_1), \ldots, (\vec{x}_n, y_n)$
2. Each x_i is a dimensional vector.
3. We want to find the maximum margin hyperplane that groups point of $\vec{x}_1$ to the group of points where y_i =1.
4. The points of $\vec{x}$ have to be satisfied by: $(\vec{w} * \vec{x} - b) = 0$ **Where $\vec{w}$ is the normal vector to the hyperplane.**

2.2 Quadratic SVM

If data sets are not linearly separable, Quadratic SVM is utilized to pick out an interval between two classes. To solve this problem the data is mapped on to a higher dimensional space and then uses a linear classifier in the higher dimensional space. For example, a linear separator can easily classify the data if we use a quadratic function to map the data into two dimensions. The general idea is to map the original feature space to a higher-dimensional feature space where the training set is separable. As the expansion increases in nth degrees it allows the data set to be trained in an efficient manner.

The algorithm of the quadratic SVM is illustrated as follows.

Algorithm of the Quadratic SVM
Input:
1. A training data set of the form: $(\vec{x}_1, y_1), \ldots, (\vec{x}_n, y_n)$
2. A kernel mapping: $K(x, y) = \langle \varphi(x), \varphi(y) \rangle$
Procedure:
1. Let the given training data set of n points be in the form of: $(\vec{x}_1, y_1), \ldots, (\vec{x}_n, y_n)$
2. The polynomial kernel is defined as: $K(x, y) = \langle \varphi(x), \varphi(y) \rangle$ where c > 0
3. For polynomials the kernel is defined by: $K(x, y) = (x^T * y + c)^d$ d = 2
4. Using the multinomial theorem the expansion becomes:

$$K(x,y) = (\sum_{i=1}^{n} x_i y_i + c)^2 = \sum_{i=1}^{n} (x_i^2 * y_i^2)$$
$$+ \sum_{i=2}^{n} \sum_{j=2}^{i-1} (\sqrt{2} * x_i * x_j)(\sqrt{2} * y_i * y_j)$$
$$+ \sum_{i=1}^{n} (\sqrt{2c} * x_i)(\sqrt{2c} * y_i) + c^2$$

2.3 Gaussian SVM

The Gaussian kernel only depends on the Euclidean distance between x and xi , and is based on the assumption that similar points are close one to each other in the feature space (in terms of Euclidean distance). The algorithm of the cubic SVM is illustrated as follows.

Algorithm of the Gaussian SVM
Input:
1. A training data set of the form: $(\vec{x}_1, y_1), \ldots, (\vec{x}_n, y_n)$
2. A kernel mapping: $K(x, y) = \langle \varphi(x), \varphi(y) \rangle$
Procedure:
1. Let the given training data set of n points be in the form of: $(\vec{x}_1, y_1), \ldots, (\vec{x}_n, y_n)$
2. The Gaussian kernel is defined as: $K(x, y) = e^{-\gamma \|x-y\|^2}$ for a given parameter $\gamma > 0$

3 Photo-Thermal Infrared (IR) Imaging Spectroscopy (PT-IRIS) Data Set

A photo-thermal infrared imaging spectroscopy (PT-IRIS) technique has recently been developed by the Naval Research Laboratory (NRL), Washington, DC with unprecedented spatial resolution at ~1 micron [8]. In this data set, the mixing Infrared IR absorption/emission features causes some complicated and overlapping samples, which leads to grand challenges to multi-class classification.

Specifically, infrared quantum cascade lasers are used to illuminate a surface potentially scattered with samples of interest. If the wavelength of the thermal emission of light is resonant with collection features of surface

samples, the sample heats by ~1oC. By varying the incident wavelength, any samples of interest could be imaged [9]. The feature of the PT-IRIS signal is the temperature increase normalized to the average power of the laser pulse at the end of the laser pulse, i.e. Tmax. Tmax as a function of excitation and collection wavelength are built into a feature vector [10], as shown in Fig. 1.

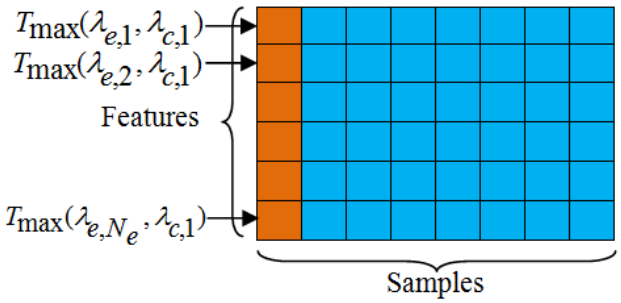

Fig. 1 Data matrix with feature vectors. Each square is a feature value, *Tmax*, which is a function of excitation and collection wavelength.

Simulated samples include 4 different particle diameters (5, 3, 2 and 7 mm) and 4 analytes (RDX, TNT, AN, Sucrose) on 4 substrates (white paint, steel, glass, polyethylene) using 38 excitation wavelengths and 33 collection wavelengths. Thus, with 38 excitation wavelengths and 33 collection wavelengths, they would generate 1254 features (predictor variables). Therefore, each column contains 1254 features and there are only 123 samples. *We* may demonstrate the signal matrix for all the 123 samples. This can be seen by display the data set in false color plot which will show visible or non-visible parts of the electromagnetic spectrum. The false color plot of the data set is shown in Fig. 2. The color of the data point is proportional to signal strength, i.e. red represents high, and blue represents low.

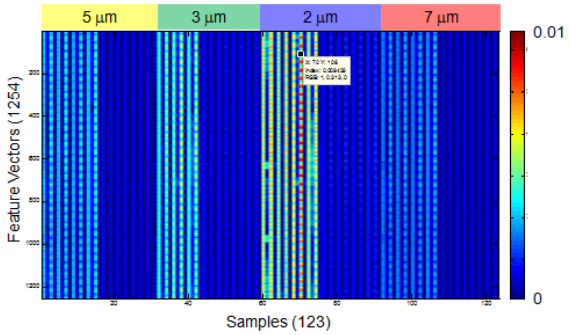

Fig. 2 False color plot of data set. The data set has 123 samples and 1254 features.

4 Simulation Analysis

4.1 Explore Data and Results in Response Plot

After a regression model is trained, the regression model results can be displayed by the response plot, i.e. the predicted response versus record number. Holdout or cross-validation is used, thus each prediction is obtained using a model that was trained without using the corresponding observation. Therefore, these predictions are the predictions on the held-out observations. 80% of the data is used to train the network and the remaining 20% data points are used as the testing data.

The response plot of linear SVM, quadratic SVM, and coarse Gaussian SVM are shown in Fig. 3, Fig. 4, and Fig. 5, respectively.

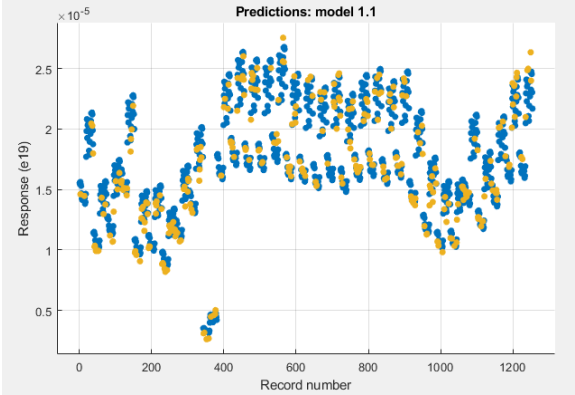

Fig. 3 The response plot of linear SVM.

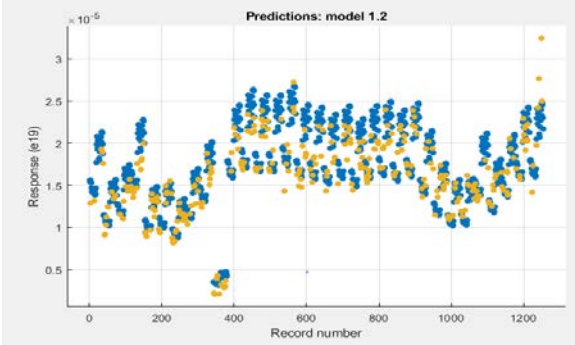

Fig. 4 The response plot of quadratic SVM.

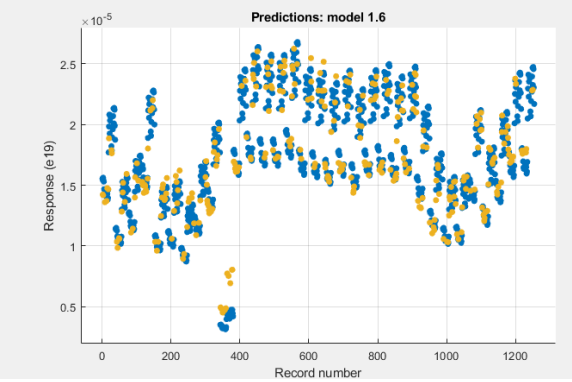

Fig. 5 The response plot of coarse Gaussian SVM.

4.2 Plot Predicted vs. Actual Response

The Predicted vs. Actual plot is used to check model performance after training a model. Use this plot to understand how well the regression model makes predictions for different response values.

When the plot is open, the predicted response of our model is plotted against the actual, true response. A perfect regression model has a predicted response equal to the true response, so all the points lie on a diagonal line. The vertical distance from the line to any point is the error of the prediction for that point. A good model has small errors, and so the predictions are scattered near the line. Usually a good model has points scattered roughly symmetrically around the diagonal line. If we can see any clear patterns in the plot, it is likely that we can improve the model.

The predicted vs. actual plot of linear SVM, quadratic SVM, and coarse Gaussian SVM are shown in Fig. 6, Fig. 7, and Fig. 8, respectively.

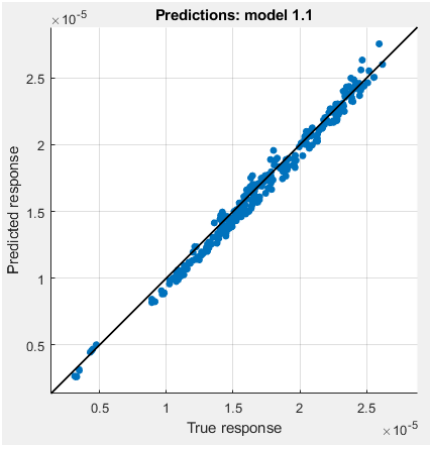

Fig. 6 The Predicted vs. Actual plot of linear SVM.

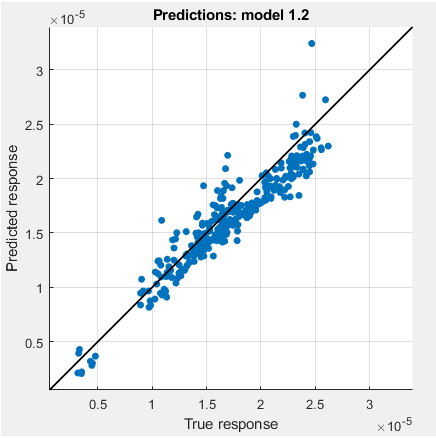

Fig. 7 The Predicted vs. Actual plot of quadratic SVM.

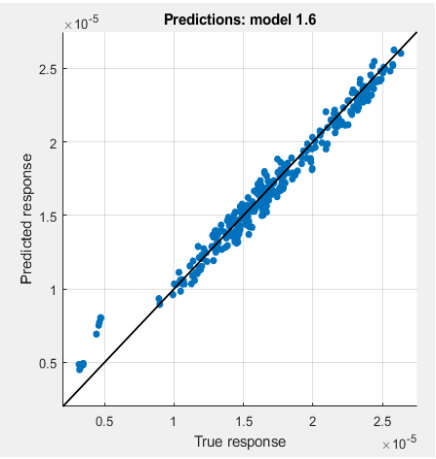

Fig. 8 The Predicted vs. Actual plot of coarse Gaussian SVM.

4.3 Evaluate Model Using Residuals Plot

We further evaluate the model performance by using the residuals plot after training a model. The residuals plot displays the difference between the predicted and true responses.

Usually a good model has residuals scattered roughly symmetrically around 0. If we can see any clear patterns in the residuals, it is likely that we can improve the model. We eapecially look for the following patterns:

- Residuals are not symmetrically distributed around 0.
- Residuals change significantly in size from left to right in the plot.
- Outliers occur, that is, residuals that are much larger than the rest of the residuals.

- Clear, nonlinear pattern appears in the residuals.

The residual plots of linear SVM, quadratic SVM, and coarse Gaussian SVM are shown in Fig. 9, Fig. 10, and Fig. 11, respectively.

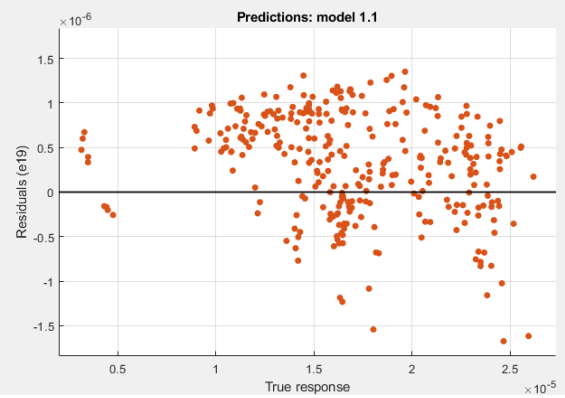

Fig. 9 The residuals plot of linear SVM.

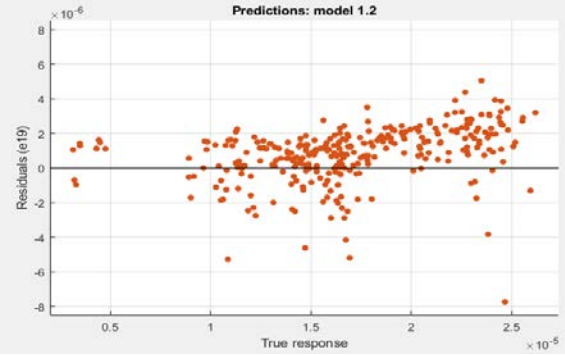

Fig. 10 The residuals plot of quadratic SVM.

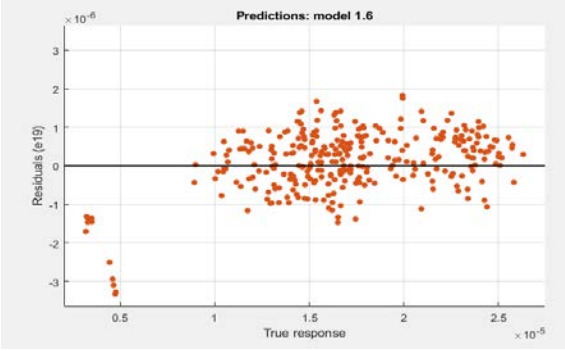

Fig. 11 The residuals plot of coarse Gaussian SVM.

4.4 Model Statistics

The model parameters are very useful and important to evaluate the performance of different models. They are defined as follows.

- RMSE (Root mean square error). The RMSE is always positive and its units match the units of the response. Look for smaller values of the RMSE.
- R-Squared. Coefficient of determination. R-squared is always smaller than 1 and usually larger than 0. It compares the trained model with the model where the response is constant and equals the mean of the training response. If the model is worse than this constant model, then R-Squared is negative. Look for an R-Squared close to 1.
- MSE (Mean squared error). The MSE is the square of the RMSE. Look for smaller values of the MSE.
- MAE (Mean absolute error). The MAE is always positive and similar to the RMSE, but less sensitive to outliers. Look for smaller values of the MAE.

For each SVM algorithm, after the network has been well trained, we evaluate the performance of each featured subset. The comprehensive comparison is shown in Table 1.

TABLE 1 COMPASIRON OF DIFFERENT SVM MODELS

	RSME	R-Sq	MSE	MAE	Train Time (sec)
Linear	$6.61*10^{-7}$	0.98	$6.61*10^{-7}$	$6.61*10^{-7}$	0.4687
Quadratic	$1.79*10^{-6}$	0.86	$3.23*10^{-12}$	$1.46*10^{-6}$	0.1182
Cubic	$3.65*10^{-6}$	0.42	$1.32*10^{-11}$	$3.10*10^{-6}$	0.1149
Fine Gaussian	$3.77*10^{-6}$	0.37	$1.43*10^{-11}$	$3.02*10^{-6}$	0.1217
Median Gaussian	$6.38*10^{-6}$	0.98	$4.07*10^{-13}$	$4.71*10^{-7}$	0.1059
Coarse Gaussian	$7.83*10^{-7}$	0.97	$6.13*10^{-13}$	$6.05*10^{-7}$	0.1444

5 Conclusions

We propose a kernel based SVM algorithm with variable models to adapt to the high-dimensional but relatively small samples for remote explosive detection on photo-thermal infrared imaging spectroscopy (PT-IRIS) classification. For each SVM algorithm it reveals classification accuracy and minimum feature number objectives. After the network has been well trained, we evaluate the performance of each featured subset. The response plot，predicted vs. actual plot, and residuals plot of the linear, quadratic, and coarse Gaussian SVM are demonstrated. A comprehensive comparison of linear SVM, quadratic SVM, cubic SVM, fine Gaussian SVM, median Gaussian SVM, coarse Gaussian SVM is performed in terms of root mean square error, R-squared, mean squared error, and mean absolute error. The excellent experimental results demonstrated that the kernel based SVM models provide a promising solution to high-dimensional data sets with limited training samples.

ACKNOWLEDGMENT

This work was supported by the National Science Foundation (NSF) grants: HRD #1505509, DUE #1654474, and HRD #1533479.

References:

[1] L. Zhang, Q. Zhang, B. Du, X. Huang, Y. Y. Tang and D. Tao, "Simultaneous Spectral-Spatial Feature Selection and Extraction for Hyperspectral Images," in IEEE Transactions on Cybernetics, vol. 48, no. 1, pp. 16-28, Jan. 2018.

[2] M. Liu, C. Xu, Y. Luo, C. Xu, Y. Wen and D. Tao, "Cost-Sensitive Feature Selection by Optimizing F-Measures," in IEEE Transactions on Image Processing, vol. 27, no. 3, pp. 1323-1335, March 2018.

[3] X. Wen, L. Shao, W. Fang and Y. Xue, "Efficient Feature Selection and Classification for Vehicle Detection," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 25, no. 3, pp. 508-517, March 2015.

[4] C. Cao, R. L. Tutwiler and S. Slobounov, "Automatic Classification of Athletes With Residual Functional Deficits Following Concussion by Means of EEG Signal Using Support Vector Machine," in IEEE Transactions on Neural Systems and Rehabilitation Engineering, vol. 16, no. 4, pp. 327-335, Aug. 2008.

[5] T. Zhang, J. Liu, S. Liu, C. Xu and H. Lu, "Boosted Exemplar Learning for Action Recognition and Annotation," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 21, no. 7, pp. 853-866, July 2011.

[6] Vapnik, V. The Nature of Statistical Learning Theory. Springer, New York, 1995.

[7] Furstenberg, R., Kendziora, C. A., Stepnowski, J., Stepnowski, S. V., Rake, M., Papantonakis, M. R., Nguyen. V., Hubler, G. K., and McGill, R. A., “Stand-Off Detection of Trace Explosives via Resonant Infrared Photothermal Imaging”, Appl. Phys. Lett., vol. 93, no. 22, 2008.

[9] Furstenberg, R., Kendziora, C. A., Stepnowski, J., Stepnowski, S. V., Rake, M., Papantonakis, M. R., Nguyen. V., Hubler, G. K., and McGill, R. A., “Stand-Off Detection of Trace Explosives via Resonant Infrared

Photothermal Imaging", Appl. Phys. Lett., vol. 93, no. 22, 2008.
[10] C. A. Kendziora, R. Furstenberg, M. Papantonakis, "Infrared Photothermal Imaging of Trace Explosives on Relevant Substrates," Proceedings of SPIE, vol. 8709, 2013.