A Structured Sparse Subspace Learning Algorithm for Anomaly Detection in UAV Flight Data

Yongfu He[1] , *Student Member, IEEE*, Yu Peng[1] , *Member, IEEE*, Shaojun Wang[1] , *Member, IEEE*, Datong Liu[1] , *Senior Member, IEEE*, Philip H.W. Leong[2] , *Senior Member, IEEE*

***Abstract*—Health status monitoring of flight-critical sensors is crucial to the flight safety of unmanned aerial vehicles (UAVs). While many flight data anomaly detection algorithms have been proposed, most do not consider data source information and cannot identify which data sources contribute most to the anomaly, hindering proper fault mitigation. To address this challenge, a structured sparse subspace learning anomaly detection (SSSLAD) algorithm which reformulates anomaly detection as a structured sparse subspace learning problem is proposed. A structured norm is imposed on the projection coefficients matrix to achieve structured sparsity and help identify anomaly sources. Utilising an efficient optimization method based on Nesterov's method, and a subspace tracking approach considering temporal dependency, the computation is efficient. Experiments on real UAV flight data sets illustrate that the proposed SSSLAD algorithm can accurately and quickly detect and identify anomalous sources in flight data, outperforming state of art algorithms, both in terms of accuracy and speed.**

***Index Terms*—Anomaly detection, subspace learning, interpretability, structured sparse, unmanned aerial vehicle.**

I. Introduction

UNMANNED aerial vehicles (UAVs) are equipped with flight-critical sensors to monitor the surrounding environment. Sensor readings are interpreted as beliefs upon which the UAV decides how to act. Unfortunately, even with pre-flight certification, sensor faults can cause the controlling software to perceive the environment incorrectly, and in turn make decisions leading to task failure [1]-[4]. For example, some faults in the sensors of determining the aircraft's altitude, might lead to a stall and then a crash [5]-[8]. Consequently, there is an urgent need to continually monitor the health of flight-critical sensors [1]-[9]. Upon detecting an issue, appropriate mitigation actions can be triggered in a timely manner.

Faults and failures in flight-critical sensors are expressed as anomalies in the flight data. The challenge is to create an accurate anomaly detection algorithm that can identify abnormal behaviour [10]. Furthermore, for successfully healthy status monitoring, mere detection of anomalies is not sufficient. Algorithms should be able to provide additional interpretable information, such as the sources responsible for the anomaly. In addition, anomaly sources in flight data are required to be identified with minimal latency for usage in a control loop.

The properties of the data are crucial to the design of an anomaly detection algorithm [4],[6],[10]-[15],[19]-[23]. Flight data are received in a streaming fashion and multidimensional. In practice, the cost of manually identifying anomalies means that often, only a limited amount of labelled flight data are available. This motivates unsupervised operation in which labelled training data are not required. On the other hand, the health of a sensor cannot be established independently. Only by taking into account information (e.g. dependency) from other sources can a reliable result be obtained [1]-[3], [6],[7]. Existing time-series anomaly detectors can be roughly divided into two approaches: temporal and spatial [10],[11],[14]-[22],[24].

The temporal approach assumes that flight data streams adjacent in time are more likely to be similar. These would appear as linearly dependent columns in the flight data stream matrix [3],[15]. Many temporal anomaly detection algorithms have been proposed [10],[11],[14],[20],[22]. Especially for flight data anomaly detection, Eliahu Khalastchi et al.[3] defines a distribution which compares the Mahalanobis distance between new n-dimensional flight data to earlier data in terms of standard deviations. Outliers are identified as those having large Mahalanobis distance from previous data storing in a sliding window. He et al. [15] assume that subspace directions might extract most information of flight data distribution. And the presence of anomalous data will lead to the deviation of flight data subspace directions. Then anomalies are determined according to the angle variation in angle of the resulting direction. The temporal oriented algorithms reviewed above perform well for detecting overall change of real-time multidimensional data at adjacent timestamps. However, to take appropriate mitigation actions, rather than simply detecting overall change, it is also of significance to provide additional interpretable information (e.g. the sources that are most responsible for anomaly) in an anomaly detection algorithm.

Flight data also present spatial dependencies which mean similar evolutions often occur between specific flight parameters, making corresponding rows of flight data stream matrix correlated. Considering spatially dependent properties, data sources information can be preserved [14],[16]-[19],[24]. In this manner, a number of interpretable algorithms have been presented which identify the sources that contribute most to the anomaly, such as stochastic nearest neighbours based [16], graph based [17], joint sparsity based [18]. Unfortunately, most

Manuscript received April 04, 2017; revised June 20, 2017; revised July 31, 2017; accepted August 04, 2017. Date of publication xx xx, 2017; date of current version xx xx, 2017. This work was supported in part by the Australian Research Councils Linkage Projects funding scheme (project number LP130101034), National Natural Science Foundation of China (NSFC) Grants 61571160 and Zomojo Pty Ltd.

[1]Yongfu He, Yu Peng, Shaojun Wang, Datong Liu are with the Department of Automatic Test and Control, School of Electrical Engineering and Automation, Harbin Institute of Technology (HIT), Harbin 10080, China (e-mail: wangsj@hit.edu.cn).

[2]Philip H.W. Leong is with the School of Electrical and Information Engineering, The University Of Sydney, 2006, Australia.

Publisher Item Identifier S 0018–9456(97)09426–6.

algorithms are not designed for online UAV applications which have highly dynamic data stream and stringent real-time constraints. For instance, Tsuyoshi Ide et al. [16] propose a neighbourhood graph where each node corresponds to a time series, and each edge is weighted by the (dis)similarity between a pair of time series. The anomaly score of the i-th source is determined by the change in the k-neighbourhood graph around the i-th node. However, as UAV flight data have a complex distributions (e.g., multi-clustered structure), the k-neighbourhood graph will result in determining improper neighbours, thus anomaly sources cannot be correctly identified. Besides, the neighbourhood graph of each source must to be constructed at each time interval, making real-time implementations challenging. Therefore, those interpretable algorithms take much more computation time to get anomaly sources, making it less suitable for real-time UAV applications.

In summary, to enhance the interpretability of flight data anomaly detection, identifying the sources that are most responsible for anomaly is still a challenge. Besides, anomaly detection of flight data needs to be done in real-time, and latency is critical when used in a control loop. Taking spatio-temporal dependencies into account, multidimensional flight data can be approximated in a lower dimensional subspace. Thus, subspace learning based methods are favored their reduced computational requirements [10],[25]-[29]. One major disadvantage of traditional subspace learning methods is that the learned subspace projection matrix is a linear combination of all the original features [25]-[30]. This mixed nature of subspace makes it hard to identify the responsible anomaly sources.

In this paper, to provide additional interpretable information and identify the sources that are responsible for the observed flight data anomaly, a structured sparse subspace learning anomaly detection (SSSLAD) algorithm is proposed. The main contributions are as follows. (1) Utilizing spatial dependency among different flight data and a predefined structured sparsity-inducing norms, the SSSLAD preserves data source information and reformulates anomaly detection to a structured sparse subspace learning problem. (2) The predefined structured norm induces the projection coefficient matrix to belong to a pre-specified sparsity pattern, which improves mixed nature of subspace. Based on the structured sparsity subspace, anomaly sources are identified correctly. (3) An efficient optimization method based on Nesterov's method is proposed to accelerate convergence of the structured sparse subspace learning problem. And considering temporal dependency that subspaces in nearby time interval share similarity, subspace tracking approach is presented to reduce time consumption.

The remainder of this paper is organized as follows: In Section 2, we discuss the challenge of applying subspace learning to provide interpretable information in anomaly detection. Sparse subspace learning is also introduced. In Section 3, we introduce the formulation of structured sparse subspace learning anomaly detection algorithm and the related optimization method. We present our experimental study in Section 4 and conclude in Section 5.

Notations. Throughout the paper, we denote vectors with bold lower case letters, and matrices with bold upper case ones. Variables are in the italic.

II. Subspace learning based anomaly detection and Sparse subspace learning

A. Subspace learning based anomaly detection

The subspace learning problem [10],[25]-[29] is formally defined as follows. Let χ be a subset of the Euclidean unit ball in $\mathbb{R}^d$, and let P be some unknown distribution over χ . The goal is to learn a subspace projection $\Pi \in \mathbb{R}^{d\times d}$ using a combination of original attributes, such that the expected squared distance, $E_{\chi\in P}[||\chi - \chi\Pi||]$, is as small as possible.

Subspace learning [10],[25]-[29] is a widely applied anomaly detection technique with applications in many domains [15],[18]-[24]. The learned subspace captures the variability of data. In such subspaces, the anomalous instances can be easily detected. In contrast with other methods, these techniques are suitable for multidimensional data sets and can work in an unsupervised setting.

The input data stream can be viewed as a continuously growing $n \times t$ matrix $\mathbf{X}_{n\times t} = [\mathbf{x}_1, \mathbf{x}_2, ...\mathbf{x}_t]$ in $\mathbb{R}^{n\times t}$, where n is the number of data sources, t is the measurement timestamp, and $\mathbf{x}_t$ is the measurement vector at t over all the data sources. At each timestep, the column vector $\mathbf{x}_t$ is appended to $\mathbf{X}_{n\times t}$. Temporal correlations appear in the data stream matrix $\mathbf{X}_{n\times t}$ across different time stamps, and spatial correlations appear across the different sources. The subspace where the projected data have the largest variation is favored for anomaly detection.

On one hand, considering the temporal dependencies between time $t-1$ and t, subspace learning based methods can operate on the column vectors $\mathbf{x}_t = [x_{1,t}, x_{2,t}, ...x_{n,t}]^T$ and the learned subspace captures the structure of the n-dimensional points [15],[20],[22]. Anomalies are indicated by a change of the direction of low dimension subspace. Similarly, on the other hand, in considering spatial dependencies, each row of data matrix $\mathbf{X}_{n\times t}$ can be treated as a point in $\mathbb{R}^t$ [16]-[18],[24]. In this approach, the subspace which learned by subspace learning as shown in eq. (1) can be divided into two parts: a low dimension subspace and high dimension subspace. In this case, the first subspace vector $\mathbf{u_1}$ in projection matrix $\mathbf{U}$ captures the strongest trend common to all data $\mathbf{X}_{n\times t}$, the second subspace vector $\mathbf{u_2}$ captures the next strongest, and so on.

$$\hat{\mathbf{U}} = \arg\min_{\mathbf{U}} \frac{1}{2}||\mathbf{X} - \mathbf{X}\mathbf{U}\mathbf{U}^T||_F^2 \quad (1)$$

where the goal is to minimise the residual between $\mathbf{X}$ and $\mathbf{X}\mathbf{U}\mathbf{U}^T$, $\mathbf{X}\mathbf{U}\mathbf{U}^T$ is the reconstructed data, and $\mathbf{U}$ is the subspace projection coefficients matrix, and $||\cdot||_F^2$ denotes the Frobenius norm. The solution is $\hat{\mathbf{U}} = [\mathbf{S}, \mathbf{G}]$. Each row of the projection coefficient matrix corresponds to a data source, while each column corresponds to a dimension of the subspace. The low dimension subspace is $\mathbf{S} = [\mathbf{u}_1, \mathbf{u}_2, ...\mathbf{u}_l]$, the high dimension subspace is $\mathbf{G} = [\mathbf{u}_{l+1}, \mathbf{u}_{l+2}, ...\mathbf{u}_n]$, n is the dimension of subspace and l is the dimension of low dimension subspace.

The low dimension subspace spans the component that is dominated by major trend in the data, and the high dimensional subspace captures the residual spikes, i.e. the abnormal patterns. An anomaly is detected when the magnitude of the projection onto the high dimension subspace exceeds a given threshold. Thus these techniques exploit correlation properties across different data sources to detect anomalies.

Recently, a subspace learning approach making use of spatial dependencies was proposed to identify the responsible anomaly sources. This is possible as data source information is preserved [18]. It assumes that the anomalous data have much more projection on the high dimension subspace and hence the subspace projection coefficients matrix could be used for anomaly source identification. However, one major disadvantage of traditional subspace learning methods is that the learned subspace projection matrix is a linear combination of all the original data sources. It is thus difficult to interpret the results [25]-[30]. To solve this mixed nature of subspace, sparse subspace learning methods were proposed.

B. Sparse subspace learning

A drawback of traditional subspace learning is that the learned subspaces are typically nonzero [25]-[29]. This is because the projection of the data on the subspace is a combination of data from all the sources, making it difficult to interpret the learned subspace and identify anomalous sources. Recently, sparse subspace learning methods have been proposed to address this issue.

Sparse subspaces with very few nonzero elements can be obtained by reformulating subspace learning as a regression-type optimization problem and imposing the lasso (elastic net) constraint ℓ_1 norm on the regression coefficients. However, sparse subspace learning is not directly applicable to anomaly identification problems in that sparsity (zero pattern) occurs randomly in the projection coefficient matrix (PCM). In fact, each row of the subspace PCM corresponds to a data source in the original data space. And, the data sources can be selected out if non-zero pattern is shown in certain row of subspace PCM. But the randomness in the subspace PCM leads to the selected data sources are independent and generally different for each dimension in the subspace. As a result, it is hard to select data sources by sparse subspace learning.

In order to select data sources with important features, those rows of the projection coefficients matrix (PCM) corresponding to unimportant features should shrink toward zero. Thus, each non-zero row of the resulting PCM corresponds to a data source in the original data space with important features. Row-sparsity (zero rows) [27] thus facilitates feature selection and can be achieved by solving

$$\hat{\mathbf{U}}=\arg\min_{\mathbf{U}} \frac{1}{2}||\mathbf{X}-\mathbf{X}\mathbf{U}\mathbf{U}^T||_F^2+\lambda||\mathbf{U}||_{2,1} \quad (2)$$

where $||\cdot||_{2,1}$ is the $\ell_{2,1}$ norm and λ is the regularization parameter. $||\mathbf{U}||_{2,1}$ denote a regularization term which penalizes $\mathbf{U}$ to achieve row-sparsity.

In some cases [31][32], the subspace is expected not only to be sparse but also has a certain structure, i.e. specific block nonzero patterns in the subspace. The structured sparsity-inducing norms Ω in eq. (3) sets entire horizontal and vertical half-spaces of the grid to zero, inducing rectangular nonzero patterns E (left of Figure 1, in black).

$$\Omega(\mathbf{w})=||\mathbf{d}^{\mathbf{G}}\circ\mathbf{w}||_{2,1} \quad (3)$$

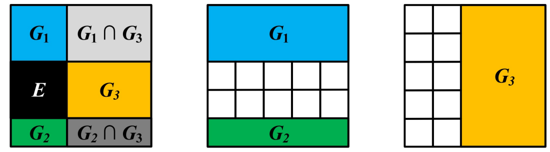


Fig. 1. Example of induced nonzero pattern E (left, in black) and three sparsity-inducing groups denoted by G_1, G_2, G_3 [31].

where $\mathbf{d}^{\mathbf{G}}=[d_1^G, ...d_j^G, ...d_p^G]$ is a $p\times p$ matrix, $G=\{G_1,G_2,G_3\}$ is the predefined subset shape, such that $d_j^G=0$ if $j\in G$ and $d_j^G>0$ otherwise, $\mathbf{w}$ is in 2-dimensional grid, and $\circ$ is the element wise product. The nonzero pattern E is the complement of the union of groups $(G_1\cup G_2\cup G_3)_c$ [31].

As shown in Figure 1, the structured sparsity-inducing norms regularization Ω controls not only the sparsity but also the structure of the supports of elements. Whereas, the sparsity regularized by ℓ_1 norm is yielded by treating each variable individually regardless of its position in the original data space.

Based on this property, structured sparse dictionary learning has been proposed which improved the performance of feature selection in the application of face recognition and bioinformatics [32]. Whereas, it focuses on controlling the structure of the dictionary $\mathbf{V}$, that cannot be directly applied for our purpose of anomaly source identification. In addition, a subspace approach with joint sparsity to identify anomaly source was proposed [18]. However, the joint sparsity approximation of subspace has to be computed repeated at each time interval and the fast optimization technique to solve the joint sparsity problem is also a major issue.

In fact, the automatic design of the sparsity-inducing norms is able to adapt to target sparsity patterns. This idea inspires us impose structure norms on the subspace projection coefficients matrix $\mathbf{U}$ and study the induced effect on the identification of anomaly sources.

Capitalizing on these results, we aim in this paper to go beyond sparse subspace learning and propose structured sparse subspace learning anomaly detection (SSSLAD) algorithm. SSSLAD will be introduced in section 3, in which the sparsity patterns of all subspace elements are structured and constrained to belong to a pre-specified set of shapes. Benefiting from a control of the structure across subspace elements, the performance of anomaly identification can be improved.

III. Structured Sparse Subspace Learning Anomaly Detection Algorithm

In this section, we describe a structured sparse subspace learning anomaly detection (SSSLAD) algorithm. Anomaly detection is reformulated to a structured sparse subspace learning problem using a structured $\ell_{2,1}$ norm on the projection coefficients matrix to achieve structured-sparsity to faciltate learning subspace and identifying anomaly sources simultaneously. Related optimization method and subspace tracking approach are presented to solve the problem and reduce execution time.

A. Framework of Model

Fig. 2 illustrates the framework. A sliding window is used to observe the streaming flight data which has five sources in

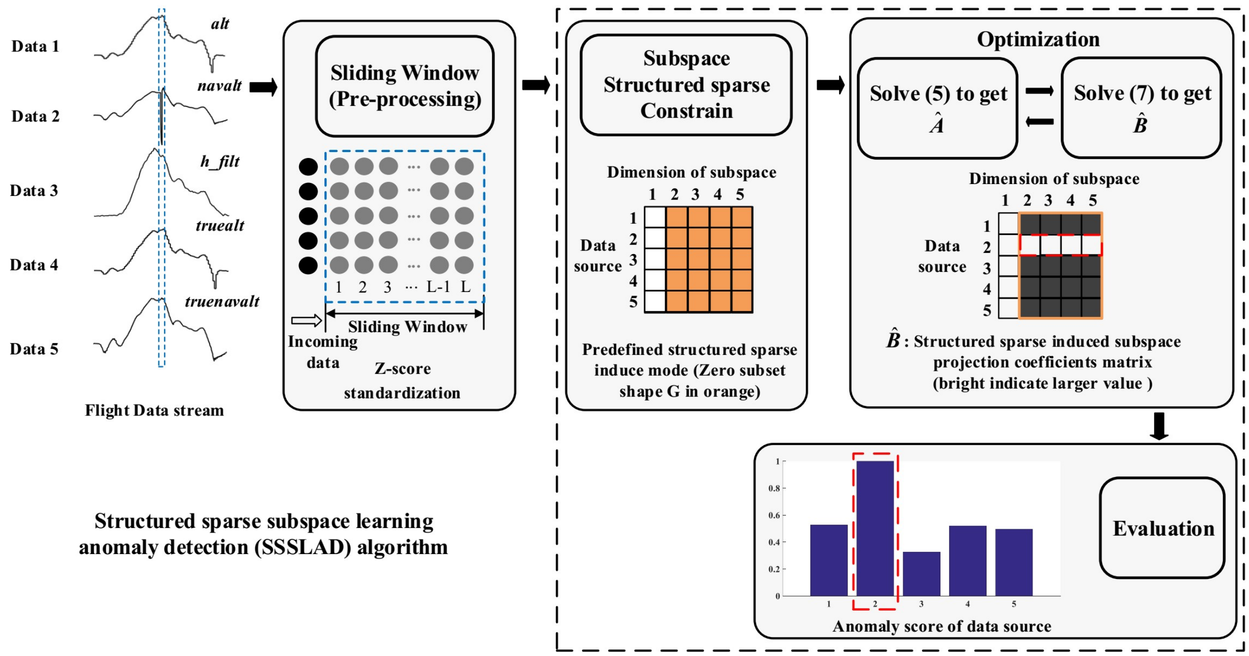


Fig. 2. Framework of structured sparse subspace learning anomaly detection (SSSLAD) algorithm.

this example. Preprocessing methods such as Z-score are used to ensure rows have a zero mean, ensuring that subspace dimensions capture true variance. $\mathbf{X}$ denotes the mean-centered flight data stream matrix in a sliding window.

The first step involves constraining the subspace with predefined structured sparsity (for instance, the orange subset G in Fig. 2). In the second step, the resulting structured sparse optimization problem is solved. A structured sparse induced subspace projection coefficients matrix (SSISPCM) is calculated in this step. The first dimension of SSISPCM accounts for the general trend of data in the sliding window. Higher dimension (2th-5th dimension in this example) captures abnormal behaviours. The benefit of the achieved structured sparsity of the subspace, is that some rows of the high dimensional SSISPCM are approximately all zero, which corresponds to the normal data sources. Other rows in the higher dimension of SSISPCM with larger values correspond to the anomaly data sources. Based on the higher dimension of SSISPCM, the last step is to calculate the anomaly scores of each data source. A larger score indicates larger possibility of the corresponding data source is abnormal.

B. Construction of structured sparse subspace learning

It is observed that flight data has a certain structure. Some parameters share common characteristics that can be embedded into a subspace. Given a flight data matrix $\mathbf{X}_{n\times t} = [\mathbf{x}_1, \mathbf{x}_2, ...\mathbf{x}_t]$ in $\mathbb{R}^{n\times t}$, we aim to learn a projection matrix, projecting the input flight data into an n-dimensional subspace. While the learned subspaces projection matrix by the traditional subspace learning approach is a linear combination of all the original data sources. This mixed nature of subspace makes it often difficult to interpret the learned result, and to identify anomalous sources.

In order to identify anomalous sources and improve the mixed nature of subspace , we want a specific set of structured sparse patterns to be in subspace, such as nonzero patterns in low dimension subspace and zero patterns in higher dimension subspace. We define an a priori structured sparse constraint on the subspace. Under this constraint, a lower dimension of subspace is as usual. While higher dimension of subspace is enforced that different subspace coefficients share exact same zero patterns. As a result, the anomalous behaviors of data are significant in higher dimension of subspace. Based on the structured sparsity subspace, we can localize anomalies sources .

To construct a subspace with this expected sparse patterns, we develop a new structured sparsity-inducing regularization scheme and a structured sparse subspace learning anomaly detection (SSSLAD) algorithm as shown in eq. (4).

$$\begin{aligned} &(\hat{\mathbf{A}},\hat{\mathbf{B}}) = \arg\min_{\mathbf{A},\mathbf{B}} \frac{1}{2}||\mathbf{X}-\mathbf{X}\mathbf{B}\mathbf{A}^T||_F^2 + \lambda||\mathbf{d}^{\mathbf{G}} \circ \mathbf{B}||_{2,1} \\ &\text{s.}t. \;\; \mathbf{A}^T\mathbf{A} = \mathbf{I}_{p\times p} \end{aligned} \tag{4}$$

where $\mathbf{X}\mathbf{B}\mathbf{A}^T$ is the reconstructed version of original data $\mathbf{X}$ based on $\mathbf{B}$ and $\mathbf{A}$, $\mathbf{B}$ is the subspace projection coefficients matrix, Ω=$||\mathbf{d}^{\mathbf{G}} \circ \mathbf{B}||_{2,1}$ is the structured sparsity-inducing regularization scheme, $\mathbf{d}^{\mathbf{G}}$= $[d_1^G,...d_l^G,...d_n^G]$ is a $n\times n$ matrix, G is the predefined zero subset shape in subspace, l controls the dimension of low dimension subspace, such that $d_l^G = \{0,...,0\}_n^T$ if $l \in G$ and $d_l^G = \{1,...,1\}_n^T$ otherwise, $\circ$ is the element wise product operator, and the regularization parameter λ controls the extent the structured sparse induced subspace projection coefficients matrix (SSISPCM) $\mathbf{B}$ is regularized. s.t. denotes *subject to*. I is unit diagonal matrix. The resulting solution, $\hat{\mathbf{A}}$ and $\hat{\mathbf{B}}$, have structured-sparsity.

In the example of Figure. 2, the predefined zero subset shape G is in orange. The structured sparse constraint will try to continuously shrink the coefficients in predefined zero subset G toward zero. However, coefficients corresponds to abnormal sources will still keep a larger value because anomaly data have much more projection on the subspace. This achieved structured sparse pattern will help to identify the anomaly source. As a result, our work formalizes flight data anomaly identification via a structured sparse regularization framework. However, an efficient convex optimization technique is required to find a solution [32]-[36].

C. Optimization method and subspace tracking

We present our optimization method to solve eq. (4) based on Nemirovski's Line Search Scheme. This is inspired by [26], although eq. (4) is not jointly convex in $\mathbf{A}$ and $\mathbf{B}$, but rather convex for $\mathbf{A}$ and $\mathbf{B}$ individually. Thus, the method solves $\mathbf{A}$ and $\mathbf{B}$ iteratively to achieve a local optimum.

$\mathbf{A}$ given $\mathbf{B}$: If $\mathbf{B}$ is fixed, we obtain the optimal $\mathbf{A}$ analytically. Ignoring the regularization part, eq. (4) is simplified to minimize $||\mathbf{X}-\mathbf{XBA}^T||_F^2$,s.t. $\mathbf{A}^T\mathbf{A}=\mathbf{I}_{p\times p}$. The solution is obtained by a reduced rank form of the Procrustes rotation. We compute the SVD:

$$\begin{aligned}(\mathbf{X}^T\mathbf{X})\mathbf{B}&=\mathbf{UDV}^T\\ \hat{\mathbf{A}}&=\mathbf{UV}^T\end{aligned} \tag{5}$$

$\mathbf{B}$ given $\mathbf{A}$: If $\mathbf{A}$ is fixed, the optimization problem becomes

$$\hat{\mathbf{B}}=\arg\min_{\mathbf{B}}\frac{1}{2}||\mathbf{X}-\mathbf{XBA}^T||_F^2+\lambda||\mathbf{d}^{\mathbf{G}}\circ\mathbf{B}||_{2,1} \tag{6}$$

As mentioned above, one appealing feature of the $\ell_{2,1}$ norm regularization is that it encourages multiple predictors to share similar sparsity patterns. However, the resulting optimization problem is challenging to solve due to the non-smoothness of the $\ell_{2,1}$ norm regularization [33]-[35]. Lower complexity bound for smooth convex optimization is significantly better than that of non-smooth convex optimization. [35] shows that the non-smoothness of the $\ell_{2,1}$ norm can be reformulated as equivalent smooth convex optimization problems, and Nesterov's method can be used to solve the problem because it is an optimal first-order black-box method for smooth convex optimization.

Due to the superior convergence rate of the smooth convex optimization over the non-smooth one, we propose to reformulate the non-smooth $\ell_{2,1}$ norm regularized problem as its equivalent constrained smooth convex optimization problem. Inspired by [35], we introduce an additional variable $\mathbf{t}=[t_1,..t_i..,t_n]^T$, where t_i acts as the upper-bound of $||d_i{}^G\circ\mathbf{B}_i||$. Equation (6) can be rewritten as

$$\begin{aligned}&\hat{\mathbf{B}}=\arg\min_{(\mathbf{t},\mathbf{B})\in D}\frac{1}{2}||\mathbf{X}-\mathbf{XBA}^T||_F^2+\rho\sum_{i=1}^{n}t_i\\ &\mathbf{t}=[t_1,..t_i..,t_n]^T\end{aligned} \tag{7}$$

Where $D=\{(\mathbf{t},\mathbf{B})|||d_i{}^G\circ\mathbf{B}_i||\le t_i,\forall_i=1,2,...,n\}$ is closed and convex.

We propose to employ the Nesterov's method [33] for solving eq. (7). The reason is that the Nesterov's method has a much faster convergence rate than the traditional methods such as subgradient descent and gradient descent [33]. Nemirovski's Line Search Scheme for the solution of $\hat{\mathbf{B}}$ in the sliding window is described in Table I.

A key building block in Nemirovski's Line Search Scheme is the Euclidean projection. Referring to step 4 in Table I, the approximate solution $[\mathbf{B}_{k+1},\mathbf{t}_{k+1}]$ is computed as a "gradient" step of $[\mathbf{B}_{k+},\mathbf{t}_k]$ by Euclidean projection. The Euclidean projection $\pi_D(\mathbf{v},\mathbf{U})$ of a given point $(\mathbf{v},\mathbf{U})$ onto the set D is defined in eq. (8) [34]-[35].

$$\pi_D(\mathbf{v},\mathbf{U})=\arg\min_{(\mathbf{t},\mathbf{B})\in D}\frac{1}{2}||\mathbf{B}-\mathbf{U}||_F^2+\frac{1}{2}||\mathbf{t}-\mathbf{v}||^2 \tag{8}$$

TABLE I
NEMIROVSKI'S LINE SEARCH SCHEME FOR THE SOLUTION OF $\hat{\mathbf{B}}$ IN THE SLIDING WINDOW

Algorithm 1
Input: l, $\mathbf{X}$, $\mathbf{A}$, ρ
Output: $\hat{\mathbf{B}}$
1: for k = 0 to . . . do
2: β_k=$(\alpha_{k-2}-1)/\alpha_{k-1}$, $\mathbf{S}_k=\mathbf{B}_k+\beta_k(\mathbf{B}_k-\mathbf{B}_{k-1})$,
$g'(\mathbf{S}_k)=\mathbf{X}^{\mathbf{T}}\mathbf{X}\mathbf{S}_k-\mathbf{X}^{\mathbf{T}}\mathbf{XA}$
3: while 1 do
4: $[\mathbf{B}_{k+1},\mathbf{t}_{k+1}]=\pi_D(\mathbf{S}_k-g'(\mathbf{S}_k)/L_k,\mathbf{t}_k-\rho/L_k,l)$
5: if $g(\mathbf{B}_{k+1})\le g(\mathbf{S}_k)+\langle g'(\mathbf{S}_k),\mathbf{B}_{k+1}-\mathbf{S}_k\rangle$
$+L_{\mathrm{k}}(||\mathbf{B}_{k+1}-\mathbf{S}_k||^2+||\mathbf{t}_{k+1}-\mathbf{t}_k||^2)/2$
6: then go to Step 9
7: else L_{k}= $2L_{\mathrm{k}}$
8: end while
9: set α_k= $(1+\sqrt{1+4\alpha_{\mathrm{k}-1}^2})/2$
10: If convergence criterion of objective function in eq.(7) is satisfied then $\mathbf{B}_k$=$\mathbf{B}_{k+1}$ and terminate the algorithm
11: end if
12: end for

Finally, the structured sparse induced subspace projection coefficients matrix (SSISPCM) is $\hat{\mathbf{B}}=[\mathbf{S},\hat{\mathbf{G}}]$, where $\hat{\mathbf{G}}=[\hat{\mathbf{u}}_{l+1},\hat{\mathbf{u}}_{l+2},...\hat{\mathbf{u}}_n]$.

Considering temporal dependency, we can store the value of SSISPCM $\mathbf{B}_{j-1}$ at $j-1^{th}$ sliding window to initialize the SSISPCM $\mathbf{B}_j$ at j^{th} sliding window before optimization. This is because the solution corresponding to $\mathbf{B}_{j-1}$ lies in the feasible domain of $\mathbf{B}_j$.

As a result, we keep tracking the value of $\mathbf{B}$ along the time direction incrementally updating the subspace. This accelerates the convergence of optimization and reduce time consumption.

D. Anomaly source scoring and overall steps of SSSLAD algorithm

To measure the degree of anomalies for each source, we define the following anomaly source score

$$\varsigma_i=\frac{\sum_{j=l+1}^{n}|g_{i,j}|}{n-l} \tag{9}$$

where $g_{i,j}$ is the element in $\hat{\mathbf{G}}$. l is the dimension of low dimension subspace. n is the dimension of subspace. ς_i is the anomaly source score for data source i.

The overall steps of our SSSLAD algorithm is illustrated in Table II.

TABLE II
STEPS OF SSSLAD ALGORITHM

Algorithm 2
Input: flight data stream $\mathbf{I}$
Output: anomaly scores of data source ς_i
1: Get observed matrix $\mathbf{X}$ by sliding window and pre-process.
2: Choose regulation parameter ρ and low dimension subspace parameter l to constrain the subspace.
3: Obtain $\hat{\mathbf{A}}$ and $\hat{\mathbf{B}}$ iteratively. Obtain $\hat{\mathbf{A}}$ by solving eq. (5) , Obtain $\hat{\mathbf{B}}$ by solving eq. (7) using Algorithm 1. The output of this step is subspace projection coefficients matrix $\hat{\mathbf{B}} = [\mathbf{S}, \hat{\mathbf{G}}]$.
4: Compute abnormal score ς_i for each source by the definition in eq. (9).

IV. RESULTS

We have conducted extensive experiments with realworld flight data sets to evaluate the performance of SSSLAD on anomaly identification in terms of both accuracy and time consumption. Three state-of-the-art anomaly identification methods: Sparse subspace learning (SSL) [26], Joint Sparse subspace learning (JSSL) [18], and K-nearest neighborhood graph (KNN-G) [16] are implemented for comparison. We implement all four methods with Matlab 2015b and perform all experiments on a laptop computer equipped with an Intel core i7-4710HQ@2.50-GHz CPU and 8 GB of memory.

A. Data Sets and Model Evaluation Metrics

We use two real-world flight data sets from UAV Laboratories at the University of Minnesota [37]-[38]. In Table III, we list the detailed information of flight data sets.

In the experiments, we only use part of flight data from take-off to landing. In real applications, flight data are in stream fashion, so our approach process data by sliding window (SW). The size and step of sliding window are 100 and 10 in the experiment. For Thor Flight 107 data set and Thor Flight 111 data set, the total number of the sliding window are 261 and 433 respectively. Anomaly is in the parameter of navalt. This kind of anomaly is in the form of contextual anomaly over related time stamps because navalt shows the different trend compared with the other 4 altitude related parameters. For each data set, we single out several abnormal windows with anomalies. The index and total number of abnormal windows (AW) are also shown in Table III below.

We use the standard Receiver Operating Characteristic (ROC) curves and Area under ROC curve (AUC) to evaluate the anomaly identification performance. The ROC curve is a standard technique for summarizing anomaly detection performance over a range of trade-offs between true positive rate (TPR) and false positive rate (FPR). AUC measures the accuracy and an AUC which is close to 1 is optimal while scores near 0.5 indicate a random decision boundary. We also measure the time to identification (TTI) to evaluate execution speed.

TABLE III
CHARACTERISTICS OF FLIGHT DATA SETS

Data sets	Thor Flight 107	Thor Flight 111
Intervals (s)	0.02	0.02
Parameters	118	118
Length	14585	10328
Index	[10000,12700]	[3900,8328]
Size(SW)	100	100
Step(SW)	10	10
Number(SW)	261	433
Indices(AW)	(119,139)	(133,151), (298,316)
Number(AW)	21	38

B. Performance

There are two tunable parameters in SSL, JSSL and our proposed SSSLAD: ρ controls the sparsity, and l controls the dimension of low dimension subspace. In SSSLAD, l also controls the predefined zero subset shape G in subspace as shown in eq. (4). Firstly, we set $\rho = 8$ and $l = 1$ in the experiment. Then we compare the performance of SSSLAD with different ρ. For KNN method, we need to select the number of neighbor $n = 1$.

1) The example of anomaly source identification in a sliding window: Figure. 3 shows the 130th sliding window in the detection process of altitude data of the Thor 107 dataset. Altitude parameters have five sources which are alt, navalt, h-filt, truealt and truenavealt respectively. The data have an upward trend before going downward. The parameter of navalt from source 2 is the anomaly source because navalt shows a different trend compared with the other 4 parameters. The anomaly source score of our SSSLAD for individual sources in the 130th sliding window is shown in Figure. 4 (The score for each source is calculated by eq. (9)) . The larger the score, the greater chance it could be the anomaly source. Thus, making use of spatially dependency among different data sources in stream flight data measurements, our SSSLAD detects the anomaly source 2 (navalt) correctly.

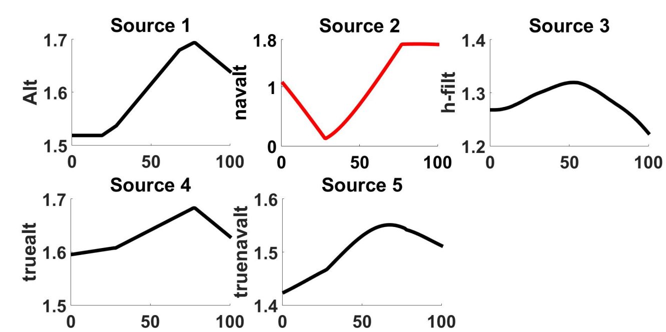


Fig. 3. Data in the 130th sliding window (altitude data of Thor 107).

2) ROC and AUC evaluation: To further evaluate our SSSLAD is adequate for anomaly identification, we calculate ROC curve and AUC. As shown in Figure. 5 and Figure. 6, for ROC curve in dataset of Thor 107 and Thor 111, we observe that the ROC curves of the SSSLAD generally lie above those of the SSL, JSSL, and KNN-G approach. We also find the

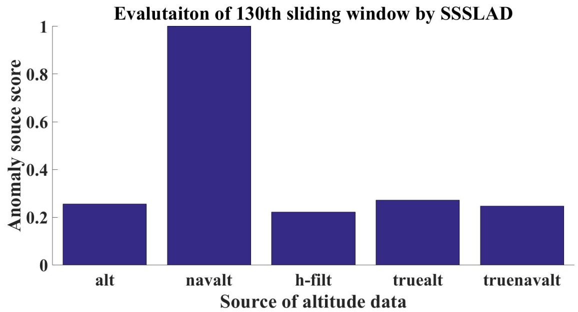


Fig. 4. Anomaly source score of the 130th sliding window (altitude data of Thor 107) by SSSLAD.

AUC value of our SSSLAD are 0.9881 and 0.9836, respectively for dataset of Thor 107 and Thor 111, which are much higher compared with the results of other approach (AUC around [0.7139,0.9395] for Thor 107, AUC around [0.8143,0.9535] for Thor 111). As a result, SSSLAD clearly outperforms the other three approach in terms of ROC and AUC. Thus, our SSSLAD shows better performance for flight data anomaly source identification.

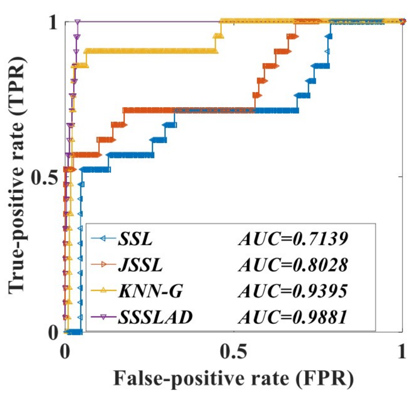


Fig. 5. Comparison of ROC curve and AUC values for Thor 107 altitude data.

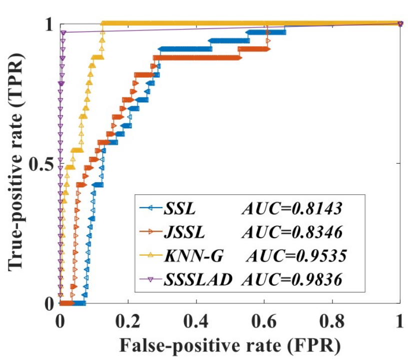


Fig. 6. Comparison of ROC curve and AUC values for Thor 111 altitude data.

We also show the details of false positive rate (FPR) at different true positive rate (TPR) in Table IV and Table V for these two dataset. Especially, for Thor 107 and Thor 111 data set, SSSLAD achieves the TPR of 97% at the cost of FPR of 3.8% and 1.0%, respectively. Our SSSLAD shows low false positive rate to identify anomaly source 2.

TABLE IV
COMPARISON OF FALSE POSITIVE RATE FOR THOR FLIGHT 107

TPR	FPR			
	SSL	JSSL	KNN-G	SSSLAD
85%	73.8%	59.1%	2.5%	2.9%
90%	77.9%	62.5%	6.3%	3.3%
97%	78.7%	66.2%	46.3%	3.8%

TABLE V
COMPARISON OF FALSE POSITIVE RATE FOR THOR FLIGHT 111

TPR	FPR			
	SSL	JSSL	KNN-G	SSSLAD
85%	29.0%	27.8%	9.8%	0.5%
90%	29.5%	52.7%	12.2%	0.7%
97%	66.1%	61.0%	13.0%	1.0%

3) Comparison of projection coefficients matrix (PCM) : Compared with SSL and JSSL, SSSLAD improves the mixed nature of data subspace by the structured sparse constraint in the case of streaming flight data. The sparse subspace learned by SSL does not fit directly into anomaly identification problems in that sparse subspace enforces sparsity randomly in the subspaces. To illustrate this, we normalize and compare the learned subspace projection coefficients matrix (PCM) of SSSLAD, JSSL and SSL for each sliding window. The size of PCM is 5×5 as altitude data is with five sources. Each row of the PCM corresponds to a data source, while each column corresponds to a dimension of the subspace. The first dimension of PCM with nonzero entries corresponds the general trend of the data in sliding window. While higher dimensions of PCM (2th-5th dimension in Fig. 7 and Fig. 8) capture abnormal behaviours of the data in sliding window. Based on the statics of higher dimensions of PCM by eq. (9), anomaly scores of each data source can be calculated. A larger score indicates higher possibility of the corresponding data source being abnormal.

We show the PCM in Figure. 7 and Figure. 8, with a brighter element indicating a larger value. We observe that for some anomaly sliding windows (for example, the 130th sliding window of Thor 107 flight altitude data in Fig. 7, the 307th sliding window of Thor 111 flight altitude data in Fig. 8), both SSSLAD and JSSL perform better than SSL and achieve the expected sparse subspace (the row of high dimensions PCM corresponds to the normal data source is dark, while the one corresponds to the anomaly data source is much brighter) that help identify the anomaly source 2. However, for some normal sliding windows (for example, the 30th sliding window of Thor 107 flight altitude data in Fig. 7 and the 11th sliding window of Thor 111 flight altitude data in Fig. 8), both JSSL and SSL fail to achieve the right subspace sparse patterns that the achieved subspace is hard to interpret. In the same sliding windows, high dimension subspace projection coefficients matrix learned by SSSLAD approximately shrink toward nearly all zero. Thus, using the predefined structured norm $\Omega = ||\mathbf{d}^{\mathbf{G}} \circ \mathbf{B}||_{2,1}$ on the projection coefficients matrix $\mathbf{B}$ (as shown in eq. (4)) to induce a specific set of structured-sparsity patterns in the subspace, SSSLAD controls not only the sparsity but also helps to identify

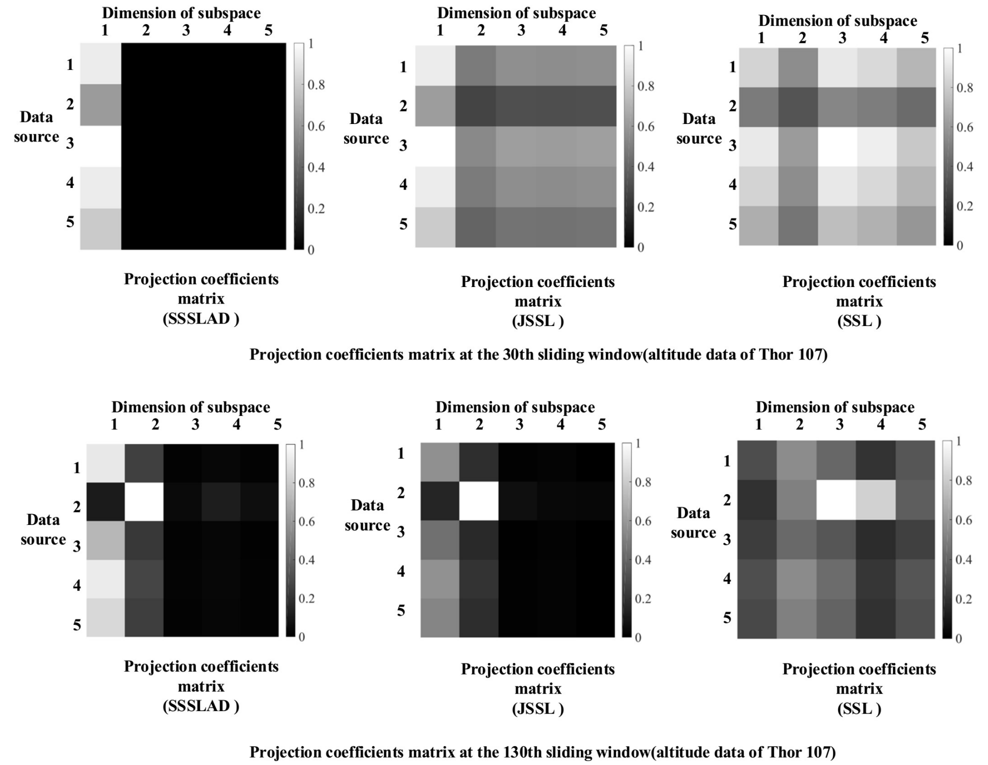


Fig. 7. Projection coefficients matrix learned by SSSLAD, JSSL and SSL at the different sliding windows (altitude data of Thor 107).

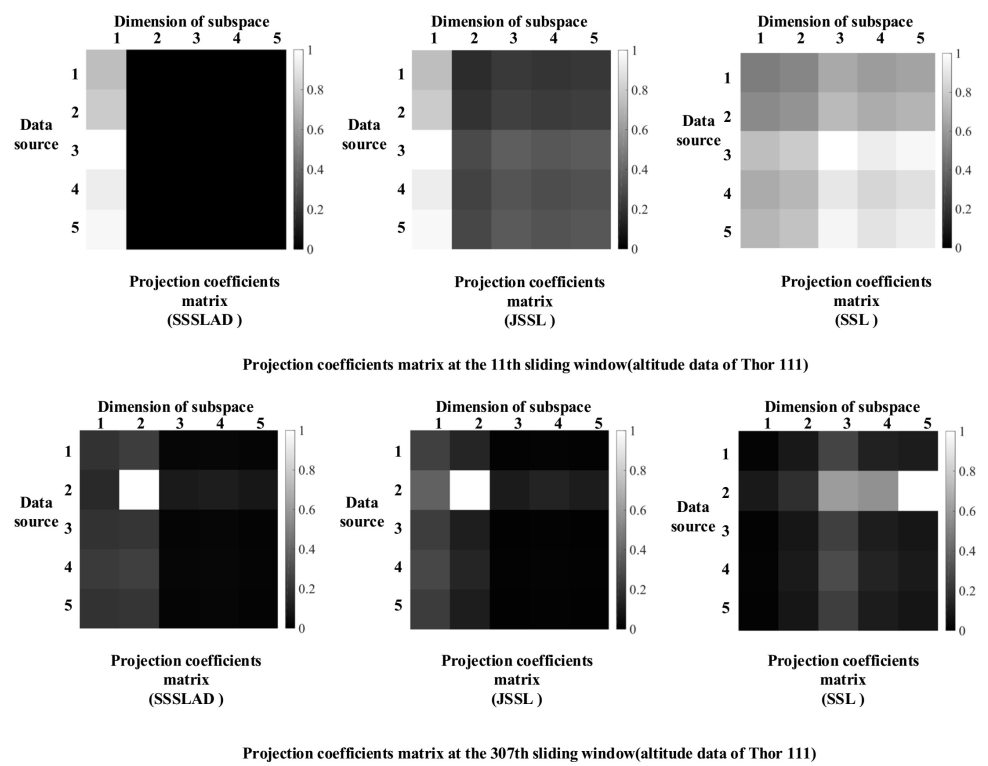


Fig. 8. Projection coefficients matrix learned by SSSLAD, JSSL and SSL at the different sliding windows (altitude data of Thor 111).

anomalous data sources. This is the reason why our SSSLAD performed better compared with JSSL and SSL.

SSSLAD outperforms the KNN-G approach because SSSLAD identifies the anomaly source by the structured sparse constraint of data subspace rather than constructing neighbourhood graph on the original data space for each source. Overall, our proposed SSSLAD achieves better performance in identifying all anomaly sources.

4) Time to identification evaluation: In addition, we evaluate the time execution of theses 4 approaches to identify the anomaly source as we should detect them in flight data as soon as possible in the UAV flight control loop. As shown in Table VI and Table VII, the total number of sliding windows for data set of Thor 107 and Thor 111 are 261 and 433, respectively. For each window, the time execution of our SSSLAD is 1.05ms and 0.94ms for Thor 107 and Thor 111 data, respectively. Compared with the time execution of SSL, JSSL and KNN-G, our SSSLAD decreases the time to identification (TTI) by 45.2%, 54.6%, and 60.1% respectively for Thor 107. For Thor 111, our SSSLAD decreases the time to identification (TTI) by 44.1%, 52.2%, 65.1% respectively. The reason is that by reformulating eq. (4) as equivalent smooth convex optimization problems in eq. (7) and making use of the proposed optimal first-order black-box optimization technology based on Nesterov's method (as shown in Table I), we not only solve the subspace learning problem, but also accelerate convergence. Moreover, considering temporal dependencies where the subspace in nearby time intervals share similarity, we keep tracking **B** in time by incrementally updating the subspace, which accelerates convergence of the optimization and reduce execution time. This is advantageous for real-time UAV flight data processing.

TABLE VI
COMPARISON OF TIME TO IDENTIFICATION FOR THOR FLIGHT 107 (261 WINDOWS)

List	Time (ms)			
	SSL	JSSL	KNN-G	SSSLAD
Total	498	601	684	273
Per window	1.91	2.30	2.62	1.05

TABLE VII
COMPARISON OF TIME TO IDENTIFICATION FOR THOR FLIGHT 111 (433 WINDOWS)

List	Time (ms)			
	SSL	JSSL	KNN-G	SSSLAD
Total	725	847	1160	405
Per window	1.67	1.96	2.68	0.94

5) Parameter evaluation: Next, we evaluate the AUC and Time to identification (TTI) for different regularization parameter ρ. ρ controls the sparsity. λ in eq. (4) is relate to ρ. As shown in Figure. 9, our SSSLAD keeps large and stable AUC value when choosing different regularization parameter ρ around [1,32] on Thor 107 and Thor 111 data. Besides, the time consumed at each sliding window also stays stable at different regularization parameter ρ as shown in Figure. 10. Therefore, the performance of our SSSLAD is stable and not sensitive to the different regularization parameter ρ around the range of [1,32].

6) Convergence: Finally, we evaluate the convergence of our SSSLAD in obtaining subspace projection coefficients matrix $\hat{\mathbf{B}}$ (as shown in Table I). We initiate the value of **A** and **B** as unit diagonal matrix and zero matrix respectively in the experiment. As shown in Figure. 11, for normal flight data (30th sliding window of Thor 107 and 11th sliding window of Thor 111 in this example), the value of objective function in eq. (7) decreases to nearly zero after 4 iterations. For anomalous data (130th sliding window of Thor 107 and 307th sliding window of Thor 111 in this example), the value of the objective function in eq. (7) decreases to a steady value after 5 iterations. SSSLAD thus achieves fast convergence and reduces processing time requirements.

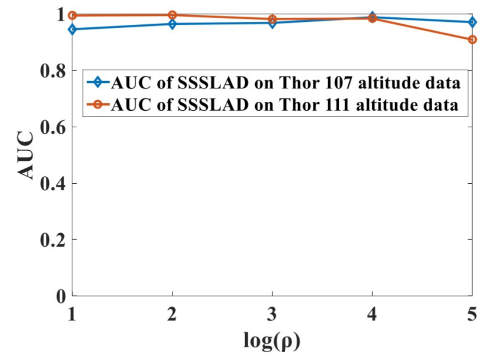


Fig. 9. AUC comparison of SSSLAD at different sparsity regularizing parameter ρ on Thor 107 and Thor 111 altitude data.

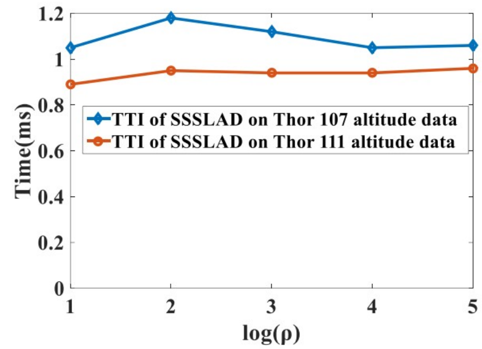


Fig. 10. TTI comparison of SSSLAD at different sparsity regularizing parameter ρ on Thor 107 and Thor 111 altitude data.

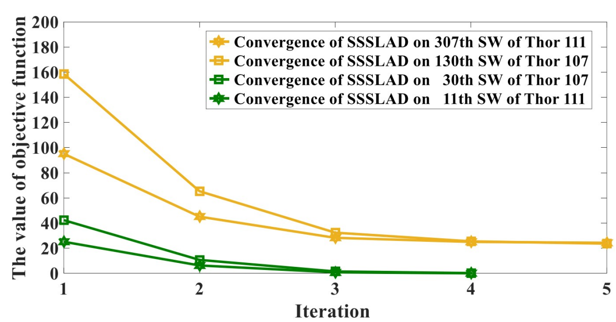


Fig. 11. Convergence of SSSLAD for different sliding windows of Thor 107 and Thor 111 altitude data.

V. CONCLUSIONS

In this work, we propose a structured sparse subspace learning anomaly detection (SSSLAD) considering spatially-temporal oriented dependency. The technique can identify anomalous sources in flight data accurately and in a timely, online manner.

Using spatially dependency and predefined structured sparsity inducing norms, the SSSLAD reformulates anomaly detection to a structured sparse subspace learning problem and preserves data sources information. A structural norm is imposed on the projection coefficients matrix to achieve structured-sparsity. Benefiting from the control of the structure across subspace projection coefficients matrix, the performance of anomaly sources identification is improved. The original non-smooth convex optimization is reformulated as equivalent smooth convex optimization problems based on Nesterov's method to accelerate convergence. Considering temporal dependency, subspace tracking approach is presented to reduce time consumption because the subspace in nearby time interval share similarity.

The experiments on two real flight data sets validate that the proposed SSSLAD can identify anomaly sources correctly and efficiently. Compared with other approach, SSSLAD can achieve good performance in terms of accuracy and speed. The study has significant supports to provide interpretability for flight data online anomaly detection.

There are three avenues for potential extensions and further work. Firstly, the trend of ever-increasing amounts of flight data create significant challenges for real-time processing. We will explore techniques to improve the scalability of this approach in the context of big data. Secondly, we will further evaluate the identification performance of SSSLAD on flight data with multi-anomaly sources. Finally, the proposed SSSLAD algorithm is evaluated on a personal laptop and does not consider power consumption of real UAV onboard application. Therefore, hardware acceleration techniques such as acceleration on field-programmable gate arrays (FPGAs) will be used to accelerate the SSSLAD with parallelization strategies and reduce power consumption to meet real UAV onboard application requirements.

Acknowledgements

This work was supported in part by the Australian Research Councils Linkage Projects funding scheme (project number LP130101034), National Natural Science Foundation of China (NSFC) Grants 61571160 and Zomojo Pty Ltd.

The authors would like to thank the editor and anonymous reviewers for their careful review and insightful comments. Moreover, the authors would like to acknowledge Prof. Peter Seiler for kindly providing the dataset and helpful suggestions.

References

[1] J. M. Schumann, K. Y. Rozier, T. Reinbacher, et al, "Towards real-time, on-board, hardware-supported sensor and software health management for unmanned aerial systems," *I. J. PHM.*, vol. 6, no. 1, pp. 1-27, Jun. 2015.

[2] S Hansen, M Blanke, "Diagnosis of airspeed measurement faults for unmanned aerial vehicles," *IEEE Trans. Aero. Elec. Sys.*, vol. 50, no. 1, pp. 224-239, Jan. 2014.

[3] E. Khalastchi, M. Kalech, G. A. Kaminka, et al, "Online data-driven anomaly detection in autonomous robots," *Knowl. Inf. Syst.*, vol. 43, no. 3, pp. 657-688, Jun. 2015.

[4] C. Chen, B. Zhang, G. Vachtsevanos, "Prediction of machine health condition using neuro-fuzzy and Bayesian algorithms," *IEEE Trans. Instrum. Meas.*, vol. 61, no. 2, pp. 297-306, Feb. 2012.

[5] X. Qi, D. Theilliol, J. Qi, et al, "A literature review on fault diagnosis methods for manned and unmanned helicopters," in *International Conference on Unmanned Aircraft Systems (ICUAS).*, May 2013, pp. 1114-1118.

[6] P. Freeman, R. Pandita, N. Srivastava, et al, "Model-based and data-driven fault detection performance for a small UAV," *IEEE-ASME Trans. Mech.*, vol. 18, no. 4, pp.1300-1309, Aug. 2013.

[7] K. F. Aljanaideh, D. S. Bernstein, "Aircraft sensor health monitoring based on transmissibility operators," *J Guid. Control. Dynam.*, vol. 38, no. 8, pp.1492-1495, Aug. 2015.

[8] P. Freeman, P. Seiler, G. J. Balas, "Air data system fault modeling and detection," *Control. Eng. Pract.*, vol. 21, no. 10, pp.1290-1301, Oct. 2013.

[9] H. E. Sevil, A. Dogan, "Fault Diagnosis in Air Data Sensors for Receiver Aircraft in Aerial Refueling," *J Guid. Control. Dynam.*, vol. 38, no. 10, pp.1959-1975, Oct. 2015.

[10] V. Chandola, A. Banerjee, V. Kumar, "Anomaly detection: A survey," *ACM Comput. Surv.*, vol. 41, no. 3, pp. 15:1-15:58, 2009.

[11] C. Liguori, A. Paolillo, A. Ruggiero, et al, "Outlier Detection for the Evaluation of the Measurement Uncertainty of Environmental Acoustic Noise," *IEEE Trans. Instrum. Meas.*, vol. 65, no. 2, pp. 234-242, Feb. 2016.

[12] D. H. Won, J. Ahn, E. Lee, et al, "GNSS Carrier Phase Anomaly Detection and Validation for Precise Land Vehicle Positioning," *IEEE Trans. Instrum. Meas.*, vol. 64, no. 9, pp. 2389-2398, Sep. 2015.

[13] S. Kumar, T. W. S. Chow, M. Pecht, "Approach to fault identification for electronic products using Mahalanobis distance," *IEEE Trans. Instrum. Meas.*, vol. 59, no. 8, pp. 2055-2064, Aug. 2010.

[14] Q. Zhang, T. Chu, "Structure Regularized Traffic Monitoring for Traffic Matrix Estimation and Anomaly Detection by Link-Load Measurements," *IEEE Trans. Instrum. Meas.*, vol. 65, no. 12, pp. 2797-2807, Dec. 2016.

[15] Y. F. He, S. J. Wang, W. J. Wang, "UAV anomaly detection based on oversampling projection approximation basis pursuit,"*Chinese Journal of Scientific instrument.*, vol. 37, no. 7, pp. 1468-1476, Jul. 2016.

[16] T. Ide, S. Papadimitriou, M. Vlachos, "Computing correlation anomaly scores using stochastic nearest neighbors," in *Seventh IEEE International Conference on Data Mining(ICDM).*, Oct. 2007, pp. 523-528.

[17] L. Akoglu , H. Tong, D. Koutra, "Graph based anomaly detection and description: a survey," *Data Min. Knowl. Disc.*, vol. 29, no. 3, pp. 626-688, May 2015.

[18] R. Jiang, H. Fei , J. Huan, "A family of joint sparse PCA algorithms for anomaly localization in network data streams,"*IEEE Trans Knowl. Data En.*, vol. 25, no. 11, pp. 2421-2433, Nov. 2013.

[19] M. Mardani, G. Mateos, G. B. Giannakis, "Dynamic anomalography: Tracking network anomalies via sparsity and low rank," *IEEE J-STSP.*, vol. 7, no. 1, pp. 50-66, Feb. 2013.

[20] Y. J. Lee, Y. R. Yeh, Y. C. F. Wang, "Anomaly detection via online oversampling principal component analysis," *IEEE Trans. Knowl. Data En.*, vol. 25, no. 7, pp. 1460-1470, Jul. 2013.

[21] F. Palmieri, U. Fiore, A. Castiglione, "A distributed approach to network anomaly detection based on independent component analysis," *Concurr. CompPract. E.*, vol. 26, no. 5, pp. 1113-1129, Apr. 2014.

[22] P. H. Dos Santos Teixeira, R. L. MilidiÃž, "Data stream anomaly detection through principal subspace trackings," in *Proceedings of the 2010 ACM Symposium on Applied Computing (SAC).*, Mar. 2010, pp. 1609-1616.

[23] J. E. Fowler, Q. Du, "Anomaly detection and reconstruction from random projections,"*IEEE Trans. Image Process.*, vol. 21, no. 1, pp. 184-195, Jan. 2012.

[24] Q. Ding, E. D. Kolaczyk, "A compressed PCA subspace method for anomaly detection in high-dimensional data," *IEEE Trans. Inform. Theory.*, vol. 59, no. 11, pp. 7419-7433, Nov. 2013.

[25] D. Cai, X. He, J. Han, "Spectral regression: A unified approach for sparse subspace learning," in *Seventh IEEE International Conference on Data Mining (ICDM).*, Oct. 2007, pp. 73-82.

[26] H. Zou, T. Hastie, R. Tibshirani, "Sparse principal component analysis," *J Comput. Graph. Stat.*, vol. 15, no. 2, pp. 265-286, 2006.

[27] Q. Gu, Z. Li, J. Han, "Joint feature selection and subspace learning," in *Proceedings-International Joint Conference on Artificial Intelligence (IJCAI).*, Jul. 2011, pp. 1294-1299.

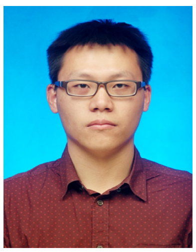
[28] M. Mardani, G. Mateos, G. B. Giannakis, "Subspace learning and imputation for streaming big data matrices and tensors," *IEEE Trans. Singal Proces.*, vol. 63, no. 10, pp. 2663-2677, May. 2015.

[29] A. Gonen, D. Rosenbaum, Y. C. Eldar, et al, "Subspace learning with partial information,"*J Mach. Learn. Res.*, vol. 17, no. 52, pp. 1-21, 2016.

[30] R. Jenatton, G. Obozinski, F. R. Bach, "Structured Sparse Principal Component Analysis," in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics (AISTATS).*, May 2010, pp. 366-373.

[31] R. Jenatton, J. Y. Audibert, F. Bach, "Structured variable selection with sparsity-inducing norms," *J Mach. Learn. Res.*, vol. 12, no. Oct, pp. 2777-2824, 2011.

[32] F. Bach, R. Jenatton, J. Mairal, et al, "Structured sparsity through convex optimization," *Stat. Sci.*, vol. , no. , pp. 450-468, 2012.
[33] J. Liu, J. Chen, J. Ye, "Large-scale sparse logistic regression," in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD).*, Jun. 2009, pp. 547-556.
[34] J. Liu, J. Ye, "Efficient Euclidean projections in linear time," in *Proceedings of the 26th Annual International Conference on Machine Learning (ICML).*, Jun. 2009, pp. 657-664.
[35] J. Liu, S. Ji, J. Ye, "Multi-task feature learning via efficient L2,1-norm minimization," in *Proceedings of the twenty-fifth conference on uncertainty in artificial intelligence (UAI).*, Jun. 2009, pp. 339-348.
[36] R. E. Precup, R. C. David, E. M. Petriu, et al, "Novel adaptive charged system search algorithm for optimal tuning of fuzzy controllers," *Expert Syst. Appl.*, vol. 41, no. 4, pp. 1168-1175, Mar. 2014.
[37] B. Taylor, "Thor Flight 107," Retrieved from the University of Minnesota Digital Conservancy, http://hdl.handle.net/11299/174231, 2014.
[38] B. Taylor, "Thor Flight 111," Retrieved from the University of Minnesota Digital Conservancy, http://hdl.handle.net/11299/174236, 2014.



Yongfu He (S'17) received his B.Sc. and M.Sc. degree from Central South University (CSU) in 2010 and 2013, respectively.

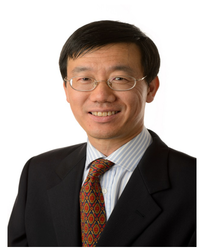
He is currently pursuing the Ph.D. degree with the School of Electrical Engineering and Automation, Harbin Institute of Technology (HIT), Harbin, China.

His research interests include machine learning and data mining, reconfigurable computing and anomaly detection methods for safety-critical systems.

Yu Peng (M'10) received the B.S. degree in Measurement Technology and Instrumentation, Harbin Institute of Technology (HIT), Harbin, China, in 1996, and M.Sc. and Ph.D. degrees in Instrumentation Science and Technology, HIT, in 1998 and 2004, respectively. He is a full Professor with the Department of Automatic Test and Control, School of Electrical Engineering and Automation, HIT. He is also the vice dean of School of Electrical Engineering and Automation, HIT. His research interests include automatic test technologies, virtual instruments, system health management, reconfigurable computing, etc.

Shaojun Wang (M'14) received his Ph.D. degree in 2012 from Harbin Institute of Technology (HIT). He was an Academic Visitor with the Imperial College London, London, U.K., from 2015-2016. He is currently an Assistant Professor with the School of Electrical Engineering and Automation, Harbin Institute of Technology (HIT), Harbin, China. His research interests include FPGA based Computing (Architecture, Dynamic Reconfiguration, Vector Processor), Machine Learning (Time Series Analysis, Image Processing), Instrumentation and Measurement (Automatic Test Technique, Data Acquisition).

Datong Liu (M'11-SM'16) received his B.Sc. degree in Automatic Test and Control from Harbin Institute of Technology (HIT), Harbin, China, in 2003. During 2001 to 2003, he also minored in the Department of Computer Science and Technology at HIT. He received the M.Sc. and Ph.D. degrees in Instrumentation Science and Technology from HIT in 2005 and 2010, respectively. He is currently an associate professor in the Department of Automatic Test and Control, School of Electrical Engineering and Automation, HIT. During 2013 to 2014, he was a visiting scholar at the University of Arizona (UofA), Tucson, AZ, USA. His research interests include automatic test, machine learning for test data processing, data-driven prognostics and health management, lithium-ion battery management, complex system health management, etc.



Philip H. W. Leong (SM'02) received the B.Sc., B.E. and Ph.D. degrees from the University of Sydney. In 1993 he was a consultant to ST Microelectronics in Milan, Italy working on advanced flash memory-based integrated circuit design. From 1997-2009 he was with the Chinese University of Hong Kong. He is currently Professor of Computer Systems in the School of Electrical and Information Engineering at the University of Sydney, Visiting Professor at Imperial College, Visiting Professor at Harbin Institute of Technology, and Chief Technology Advisor to ClusterTech. Prof. Leong was co-founder and program co-chair of the International Conference on Field Programmable Technology (FPT); program co-chair of the International Conference on Field Programmable Logic and Applications (FPL); Senior Associate Editor for ACM Transactions on Reconfigurable Technology and Systems (TRETS) and Associate Editor for IEEE Transactions on Computer Aided Design (TCAD). His current research interests include Reconfigurable computing, parallel processing, signal processing, computer architecture, computer arithmetic and biologically inspired computing.