Supplementary Material to Sparse Coding and Dictionary Learning with Linear Dynamical Systems*

Wenbing Huang[1], Fuchun Sun[1], Lele Cao[1], Deli Zhao[2], Huaping Liu[1] and Mehrtash Harandi[3]
[1] Department of Computer Science and Technology, Tsinghua University, State Key Lab. of Intelligent Technology and Systems, Tsinghua National Lab. for Information Science and Technology (TNList);
[3] Australian National University & NICTA, Australia;
[1]{huangwb12@mails, fcsun@mail, caoll12@mails, hpliu@mail}.tsinghua.edu.cn,
[2]zhaodeli@gmail.com, [3]Mehrtash.Harandi@nicta.com.au,

In this supplementary material, we present the proofs of Theorems (1-3), the algorithm for learning the transition matrix of LDSST, and the reconstruction error approach for classification in LDS-SC, LDSST-SC and covLDSST-SC. In addition, we describe the details of the benchmark datasets that are applied in our experiments. Our dictionary learning algorithm for anormaly detection is also explored in this supplementary material.

1. Proofs

Theorem 1. *Suppose $\mathbf{V}_1, \mathbf{V}_2, \cdots, \mathbf{V}_M \in \mathcal{S}(n, \infty)$, and $y_1, y_2, \cdots, y_M \in \mathbb{R}$, we have*

$$\| \sum_{i=1}^{M} y_i \Pi(\mathbf{V}_i) \|_F^2 \quad = \quad \sum_{i,j=1}^{M} y_i y_j \| \mathbf{V}_i^{\mathrm{T}} \mathbf{V}_j \|_F^2,$$

where $\mathbf{V}_i^{\mathrm{T}} \mathbf{V}_j = \mathbf{L}_i^{-1} \mathbf{O}_i^{\mathrm{T}} \mathbf{O}_j \mathbf{L}_j^{-\mathrm{T}}$. $\mathbf{O}_i^{\mathrm{T}} \mathbf{O}_j$ can be computed with the Lyapunov equation defined in Equation (2), $\mathbf{L}_i$ and $\mathbf{L}_j$ are Cholesky decomposition matrices for $\mathbf{O}_i^{\mathrm{T}} \mathbf{O}_i$ and $\mathbf{O}_j^{\mathrm{T}} \mathbf{O}_j$, respectively.

Proof. We denote the sub-matrix of the extended observability matrix $\mathbf{O}_i$ as $\mathbf{O}_i(t) = [\mathbf{C}_i^{\mathrm{T}}, (\mathbf{C}_i \mathbf{A}_i)^{\mathrm{T}}, \cdots, (\mathbf{C}_i \mathbf{A}_i^t)]^{\mathrm{T}}$ by taking the first t rows. We suppose that the Cholesky decomposition matrix for $\mathbf{O}_i$ is $\mathbf{L}_i$ and denote that $\mathbf{V}_i(t) = \mathbf{O}_i(t) \mathbf{L}_i^{-\mathrm{T}}$. Then, we derive

$$\begin{aligned}
\| \sum_{i=1}^{K} y_i \Pi(\mathbf{V}_i) \|_F^2 &= \lim_{t \to \infty} \| \sum_{i=1}^{M} y_i \mathbf{V}_i(t) \mathbf{V}_i(t)^{\mathrm{T}} \|_F^2 \\
&= \lim_{t \to \infty} \mathrm{Tr} \left(\sum_{i=1}^{M} y_i \mathbf{V}_i(t) \mathbf{V}_i(t)^{\mathrm{T}} \sum_{j=1}^{K} y_j \mathbf{V}_j(t) \mathbf{V}_j(t)^{\mathrm{T}} \right) \\
&= \lim_{t \to \infty} \sum_{i,j=1}^{M} y_i y_j \mathrm{Tr} \left(\mathbf{V}_i(t)^{\mathrm{T}} \mathbf{V}_j(t) \mathbf{V}_j(t)^{\mathrm{T}} \mathbf{V}_i(t) \right) \\
&= \sum_{i,j=1}^{M} y_i y_j \lim_{t \to \infty} \| \mathbf{V}_i(t)^{\mathrm{T}} \mathbf{V}_j(t) \|_F^2 \\
&= \sum_{i,j=1}^{M} y_i y_j \lim_{t \to \infty} \| \mathbf{L}_i^{-1} (\mathbf{O}_i(t)^{\mathrm{T}} \mathbf{O}_j(t)) \mathbf{L}_j^{-\mathrm{T}} \|_F^2 \\
&= \sum_{i,j=1}^{M} y_i y_j \| \mathbf{L}_i^{-1} \mathbf{O}_{ij} \mathbf{L}_j^{-\mathrm{T}} \|_F^2, \qquad (13)
\end{aligned}$$

*This work is jointly supported by National Natural Science Foundation of China under Grant No. 61327809, 61210013, 91420302 and 91520201.

where the limitation value $\mathbf{O}_{ij} = \lim_{t\to\infty} \mathbf{O}_i(t)^{\mathrm{T}}\mathbf{O}_j(t) = \mathbf{O}_i^{\mathrm{T}}\mathbf{O}_j$ can be computed by solving the Lyapunov equation similar to Equation (2). □

The Frobenius distance $\parallel \Pi(\mathbf{V}_1) - \Pi(\mathbf{V}_2) \parallel_F^2$ in Corollary (1) can be computed by setting $y_1 = 1$ and $y_2 = -1$ in Theorem (1).

As demonstrated in [7], the embedding $\Pi(\mathbf{V})$ from the finite Grassmannian $\mathcal{G}(n,d)$ to the space of the symmetric matrices is proven to be diffeomorphism (a one-to-one, continuous, and differentiable mapping with a continuous and differentiable inverse); The Frobenius distance between the two points $\mathbf{V}_1$ and $\mathbf{V}_2$ in the embedding space can be rewritten as

$$\parallel \mathbf{V}_1\mathbf{V}_1^{\mathrm{T}} - \mathbf{V}_2\mathbf{V}_2^{\mathrm{T}} \parallel_F^2 = 2\sum_{k=1}^{n} \sin^2 \alpha_k, \tag{14}$$

where α_k is the k-th principal angle of the subspaces between $\mathbf{V}_1$ and $\mathbf{V}_2$.

We denote the space of the finite observability subspaces as $\mathcal{S}(n,d)$ by taking the first d rows from the extended observability matrix. Clearly, $\mathcal{S}(n,d)$ is a closed subset of $\mathcal{G}(n,d)$. Hence $\mathcal{S}(n,d)$ maintain the relation in Equation (14); and the embedding $\Pi(\mathbf{V})$ from $\mathcal{S}(n,d)$ to the space of the symmetric matrices is diffeomorphism.

For our case, $\mathcal{S}(n,\infty) = \lim_{d\to\infty} \mathcal{S}(n,d)$. In Theorem (1), the Frobenius distance defined in the embedding $\Pi(\mathcal{S}(n,\infty))$ is proven to be convergent. Thus, we can obtain the relation between the Frobenius distance and the subspace angles in Corollary (1), and prove that the embedding $\Pi(\mathcal{S}(n,\infty))$ is diffeomorphism in Corollary (2), by extending the conclusions of $\mathcal{S}(n,d)$ with d approaching to the infinity.

In Section 4, we derive the dictionary learning problem by adding the symmetric constraints to the transition matrices of the data and dictionary LDSs. Here we give the details. The tuple $(\mathbf{A}', \mathbf{C}')$ is regarded to be equivalent to the tuple $(\mathbf{A}, \mathbf{C})$ if and only if there exists an orthonormal square matrix $\mathbf{P}$ satisfying $\mathbf{A}' = \mathbf{P}^{-1}\mathbf{A}\mathbf{P}$ and $\mathbf{C}' = \mathbf{C}\mathbf{P}$. Clearly, the equivalent tuples derive the same target $\Gamma(r)$ defined in Equation (6). We have the following conclusion:

Lemma 1. *If the transition matrix $\bar{\mathbf{A}}_r$ is symmetric, then any equivalent transformation of the tuple $(\bar{\mathbf{A}}_r, \bar{\mathbf{C}}_r)$ is equivalent to the same standard form given by $(\bar{\mathbf{\Theta}}_r, \hat{\mathbf{C}}_r)$, where the matrix $\bar{\mathbf{\Theta}}_r$ is diagonal with the elements being the eigenvalues of the matrix $\bar{\mathbf{A}}_r$, the matrix $\hat{\mathbf{C}}_r = \bar{\mathbf{C}}_r\mathbf{P}_r^{-1}$, and $\mathbf{P}_r$ is the orthonormal square matrix.*

Proof. If the matrix $\bar{\mathbf{A}}_r$ is symmetric, there exists an orthonormal square matrix $\mathbf{P}_r$ satisfying

$$\begin{aligned}\bar{\mathbf{A}}_r &= \mathbf{P}_r^{-1}\begin{pmatrix} \bar{\boldsymbol{\theta}}_{r,1} & & \\ & \ddots & \\ & & \bar{\boldsymbol{\theta}}_{r,n}\end{pmatrix}\mathbf{P}_r \\ &= \mathbf{P}_r^{-1}\bar{\mathbf{\Theta}}_r\mathbf{P}_r,\end{aligned} \tag{15}$$

where $\bar{\boldsymbol{\theta}}_{r,1}, \cdots, \bar{\boldsymbol{\theta}}_{r,n}$ are the eigenvalues of matrix $\bar{\mathbf{A}}_r$.

The equivalent transformation of the tuple $(\bar{\mathbf{A}}_r, \bar{\mathbf{C}}_r)$ has the form $(\mathbf{P}^{-1}\bar{\mathbf{A}}_r\mathbf{P}, \bar{\mathbf{C}}_r\mathbf{P})$, where $\mathbf{P}$ is an orthonormal square matrix. By denoting the equivalence relation as the symbol "$\sim$", we obtain

$$\begin{aligned}(\mathbf{P}^{-1}\bar{\mathbf{A}}_r\mathbf{P}, \bar{\mathbf{C}}_r\mathbf{P}) &\sim (\bar{\mathbf{A}}_r, \bar{\mathbf{C}}_r) \\ &= (\mathbf{P}_r^{-1}\bar{\mathbf{\Theta}}_r\mathbf{P}_r, \bar{\mathbf{C}}_r) \\ &\sim (\bar{\mathbf{\Theta}}_r, \bar{\mathbf{C}}_r\mathbf{P}_r^{-1}) \\ &= (\bar{\mathbf{\Theta}}_r, \hat{\mathbf{C}}_r).\end{aligned} \tag{16}$$

Thus we conclude the proof. □

For consistency and unequivocalness, we ignore the difference between $\hat{\mathbf{C}}_r$ and $\bar{\mathbf{C}}_r$, and denote $\hat{\mathbf{C}}_r$ as $\bar{\mathbf{C}}_r$ in the following context. Lemma (1) shows that minimizing $\Gamma(r)$ can be equivalently performed on the space of the standard tuple $(\bar{\mathbf{\Theta}}_r, \bar{\mathbf{C}}_r)$. We denote that the eigenvalues of the matrix $\bar{\mathbf{A}}_j$ and $\mathbf{A}_i$ are $\bar{\boldsymbol{\theta}}_j = [\bar{\boldsymbol{\theta}}_{j,1}, \cdots, \bar{\boldsymbol{\theta}}_{j,n}]$ and $\boldsymbol{\theta}_i = [\boldsymbol{\theta}_{i,1}, \cdots, \boldsymbol{\theta}_{i,n}]$, respectively. The notation $[\bar{\mathbf{C}}_r]_k$ denotes the k-th column of matrix $\bar{\mathbf{C}}_r$.

Theorem 2. *If the transition matrices of dictionary atoms and the data systems are all symmetric, then Equation (7) is equivalent to*

$$
\begin{aligned}
\min_{\bar{\mathbf{C}}_r, \bar{\boldsymbol{\theta}}_r} \quad & \sum_{k=1}^{n} [\bar{\mathbf{C}}_r]_k^{\mathrm{T}} \mathbf{S}(r,k) [\bar{\mathbf{C}}_r]_k \\
\text{s.t.} \quad & \bar{\mathbf{C}}_r^{\mathrm{T}} \bar{\mathbf{C}}_r = \mathbf{I}_n; \quad -1 < \bar{\boldsymbol{\theta}}_{r,k} < 1,\ 1 \le k \le n.
\end{aligned} \tag{17}
$$

Here, $\mathbf{S}(r,k) = \sum_{i=1}^{N} \sum_{j=1, j\neq r}^{K} \mathbf{Z}_{r,i} \mathbf{Z}_{j,i} \bar{\mathbf{C}}_j \mathbf{E}(r,j,k) \bar{\mathbf{C}}_j^{\mathrm{T}} - \sum_{i=1}^{N} \mathbf{Z}_{r,i} \mathbf{C}_i \mathbf{F}(r,i,k) \mathbf{C}_i^{\mathrm{T}}$*;* $\mathbf{E}(r,j,k)$ *and* $\mathbf{F}(r,i,k)$ *are diagonal matrices:* $\mathbf{E}(r,j,k) = \mathrm{diag}([\frac{(1-\bar{\boldsymbol{\theta}}_{r,k}^2)(1-\bar{\boldsymbol{\theta}}_{j,1}^2)}{(1-\bar{\boldsymbol{\theta}}_{r,k}\bar{\boldsymbol{\theta}}_{j,1})^2}, \cdots, \frac{(1-\bar{\boldsymbol{\theta}}_{r,k}^2)(1-\bar{\boldsymbol{\theta}}_{j,n}^2)}{(1-\bar{\boldsymbol{\theta}}_{r,k}\bar{\boldsymbol{\theta}}_{j,n})^2}])$*, and* $\mathbf{F}(r,i,k) = \mathrm{diag}([\frac{(1-\bar{\boldsymbol{\theta}}_{r,k}^2)(1-\boldsymbol{\theta}_{i,1}^2)}{(1-\bar{\boldsymbol{\theta}}_{r,k}\boldsymbol{\theta}_{i,1})^2}, \cdots, \frac{(1-\bar{\boldsymbol{\theta}}_{r,k}^2)(1-\boldsymbol{\theta}_{i,n}^2)}{(1-\bar{\boldsymbol{\theta}}_{r,k}\boldsymbol{\theta}_{i,n})^2}])$.

Proof. According to Lemma (1), we can replace the dictionary tuples $\{(\bar{\mathbf{A}}_j, \bar{\mathbf{C}}_j)\}_{j=1}^{K}$ and the data tuples $\{(\mathbf{A}_i, \mathbf{C}_i)\}_{i=1}^{N}$ in Equation (6) with the equivalent standard forms $\{(\bar{\boldsymbol{\Theta}}_j, \bar{\mathbf{C}}_j)\}_{j=1}^{K}$ and $\{(\boldsymbol{\Theta}_i, \mathbf{C}_i)\}_{i=1}^{N}$, respectively. Therefore, the objective function $\Gamma(r)$ is rewritten as

$$
\begin{aligned}
\Gamma(r) \quad = \quad & \sum_{i=1}^{N} \sum_{j=1, j\neq r}^{K} \mathbf{Z}_{r,i} \mathbf{Z}_{j,i} \parallel \bar{\mathbf{L}}_r^{-1} \sum_{t=0}^{\infty} \bar{\boldsymbol{\Theta}}_r^t \bar{\mathbf{C}}_r^{\mathrm{T}} \bar{\mathbf{C}}_j \bar{\boldsymbol{\Theta}}_j^t \bar{\mathbf{L}}_j^{-\mathrm{T}} \parallel_F^2 \\
& - \sum_{i=1}^{N} \mathbf{Z}_{r,i} \parallel \bar{\mathbf{L}}_r^{-1} \sum_{t=0}^{\infty} \bar{\boldsymbol{\Theta}}_r^t \bar{\mathbf{C}}_r^{\mathrm{T}} \mathbf{C}_i \boldsymbol{\Theta}_i^t \mathbf{L}_i^{-\mathrm{T}} \parallel_F^2,
\end{aligned} \tag{18}
$$

The minimization problem defined in Equation (7) is equivalently transformed to the following form

$$
\min_{\bar{\boldsymbol{\Theta}}_r, \bar{\mathbf{C}}_r} \ \Gamma(r), \qquad \text{s.t.} \ \bar{\mathbf{C}}_r^{\mathrm{T}} \bar{\mathbf{C}}_r = \mathbf{I}_n,\ -1 < \bar{\boldsymbol{\theta}}_{r,k} < 1,\ 1 \le k \le n. \tag{19}
$$

We denote the Cholesky matrices of the dictionary tuple $(\bar{\boldsymbol{\Theta}}_j, \bar{\mathbf{C}}_j)$ and the data tuple $(\boldsymbol{\Theta}_i, \mathbf{C}_i)$ as $\bar{\mathbf{L}}_j$ and $\mathbf{L}_i$, respectively. Since $\bar{\mathbf{L}}_j \bar{\mathbf{L}}_j^{\mathrm{T}} = \bar{\boldsymbol{\Theta}}_j^{\mathrm{T}} \bar{\boldsymbol{\Theta}}_j$, the Cholesky matrix can be derived as

$$
\bar{\mathbf{L}}_j \quad = \quad \begin{pmatrix} \frac{1}{(1-\bar{\boldsymbol{\theta}}_{j,1}^2)^{\frac{1}{2}}} & & \\ & \ddots & \\ & & \frac{1}{(1-\bar{\boldsymbol{\theta}}_{j,n}^2)^{\frac{1}{2}}} \end{pmatrix},
$$

and similar calculation can be applied to derive $\mathbf{L}_i$.

Then,

$$\begin{aligned}
& \| \bar{\mathbf{L}}_r^{-1} \sum_{t=0}^{\infty} \bar{\mathbf{\Theta}}_r^t \bar{\mathbf{C}}_r^{\mathrm{T}} \bar{\mathbf{C}}_j \bar{\mathbf{\Theta}}_j^t \bar{\mathbf{L}}_j^{-\mathrm{T}} \|_F^2 \\
= & \operatorname{Tr}\left(\bar{\mathbf{L}}_r^{-1} \left(\sum_{t_1=0}^{\infty} \bar{\mathbf{\Theta}}_r^{t_1} \bar{\mathbf{C}}_r^{\mathrm{T}} \bar{\mathbf{C}}_j \bar{\mathbf{\Theta}}_j^{t_1} \right) \bar{\mathbf{L}}_j^{-\mathrm{T}} \bar{\mathbf{L}}_j^{-1} \left(\sum_{t_2=0}^{\infty} \bar{\mathbf{\Theta}}_j^{t_2} \bar{\mathbf{C}}_j^{\mathrm{T}} \bar{\mathbf{C}}_r \bar{\mathbf{\Theta}}_r^{t_2} \right) \bar{\mathbf{L}}_r^{-\mathrm{T}} \right) \\
= & \sum_{t_1=0}^{\infty} \sum_{t_2=0}^{\infty} \operatorname{Tr}\left((\bar{\mathbf{L}}_r^{-1} \bar{\mathbf{\Theta}}_r^{t_1}) \bar{\mathbf{C}}_r^{\mathrm{T}} \bar{\mathbf{C}}_j \bar{\mathbf{\Theta}}_j^{t_1} (\bar{\mathbf{L}}_j^{-\mathrm{T}} \bar{\mathbf{L}}_j^{-1}) \bar{\mathbf{\Theta}}_j^{t_2} \bar{\mathbf{C}}_j^{\mathrm{T}} \bar{\mathbf{C}}_r (\bar{\mathbf{\Theta}}_r^{t_2} \bar{\mathbf{L}}_r^{-\mathrm{T}}) \right) \\
= & \sum_{t_1=0}^{\infty} \sum_{t_2=0}^{\infty} \sum_{k=1}^{n} (1-\bar{\boldsymbol{\theta}}_{r,k}^2)^{\frac{1}{2}} \bar{\boldsymbol{\theta}}_{r,k}^{t_1} [\bar{\mathbf{C}}_r]_k^{\mathrm{T}} \bar{\mathbf{C}}_j \bar{\mathbf{\Theta}}_j^{t_1} (\bar{\mathbf{L}}_j^{-\mathrm{T}} \bar{\mathbf{L}}_j^{-1}) \bar{\mathbf{\Theta}}_j^{t_2} \bar{\mathbf{C}}_j^{\mathrm{T}} [\bar{\mathbf{C}}_r]_k \bar{\boldsymbol{\theta}}_{r,k}^{t_2} (1-\bar{\boldsymbol{\theta}}_{r,k}^2)^{\frac{1}{2}} \\
= & \sum_{t_1=0}^{\infty} \sum_{t_2=0}^{\infty} \sum_{k=1}^{n} [\bar{\mathbf{C}}_r]_k^{\mathrm{T}} \bar{\mathbf{C}}_j (\bar{\boldsymbol{\theta}}_{r,k}^{t_1} \bar{\mathbf{\Theta}}_j^{t_1}) ((1-\bar{\boldsymbol{\theta}}_{r,k}^2) \bar{\mathbf{L}}_j^{-\mathrm{T}} \bar{\mathbf{L}}_j^{-1}) (\bar{\boldsymbol{\theta}}_{r,k}^{t_2} \bar{\mathbf{\Theta}}_j^{t_2}) \bar{\mathbf{C}}_j^{\mathrm{T}} [\bar{\mathbf{C}}_r]_k \\
= & \sum_{k=1}^{n} [\bar{\mathbf{C}}_r]_k^{\mathrm{T}} \bar{\mathbf{C}}_j \left(\sum_{t_1=0}^{\infty} \bar{\boldsymbol{\theta}}_{r,k}^{t_1} \bar{\mathbf{\Theta}}_j^{t_1} \right) ((1-\bar{\boldsymbol{\theta}}_{r,k}^2) \bar{\mathbf{L}}_j^{-\mathrm{T}} \bar{\mathbf{L}}_j^{-1}) \left(\sum_{t_2=0}^{\infty} \bar{\boldsymbol{\theta}}_{r,k}^{t_2} \bar{\mathbf{\Theta}}_j^{t_2} \right) \bar{\mathbf{C}}_j^{\mathrm{T}} [\bar{\mathbf{C}}_r]_k \\
= & \sum_{k=1}^{n} [\bar{\mathbf{C}}_r]_k^{\mathrm{T}} \bar{\mathbf{C}}_j \mathbf{E}(r,j,k) \bar{\mathbf{C}}_j^{\mathrm{T}} [\bar{\mathbf{C}}_r]_k, \qquad (20)
\end{aligned}$$

where $\mathbf{E}(r,j,k) = \operatorname{diag}([\frac{(1-\bar{\boldsymbol{\theta}}_{r,k}^2)(1-\bar{\boldsymbol{\theta}}_{j,1}^2)}{(1-\bar{\boldsymbol{\theta}}_{r,k}\bar{\boldsymbol{\theta}}_{j,1})^2}, \cdots, \frac{(1-\bar{\boldsymbol{\theta}}_{r,k}^2)(1-\bar{\boldsymbol{\theta}}_{j,n}^2)}{(1-\bar{\boldsymbol{\theta}}_{r,k}\bar{\boldsymbol{\theta}}_{j,n})^2}])$.

Similarly,

$$\| \bar{\mathbf{L}}_r^{-1} \sum_{t=0}^{\infty} \bar{\mathbf{\Theta}}_r^t \bar{\mathbf{C}}_r^{\mathrm{T}} \mathbf{C}_i \mathbf{\Theta}_i^t \mathbf{L}_i^{-\mathrm{T}} \|_F^2 \quad = \quad \sum_{k=1}^{n} [\bar{\mathbf{C}}_r]_k^{\mathrm{T}} \mathbf{C}_i \mathbf{F}(r,i,k) \mathbf{C}_i^{\mathrm{T}} [\bar{\mathbf{C}}_r]_k, \qquad (21)$$

where $\mathbf{F}(r,i,k) = \operatorname{diag}([\frac{(1-\bar{\boldsymbol{\theta}}_{r,k}^2)(1-\boldsymbol{\theta}_{i,1}^2)}{(1-\bar{\boldsymbol{\theta}}_{r,k}\boldsymbol{\theta}_{i,1})^2}, \cdots, \frac{(1-\bar{\boldsymbol{\theta}}_{r,k}^2)(1-\boldsymbol{\theta}_{i,n}^2)}{(1-\bar{\boldsymbol{\theta}}_{r,k}\boldsymbol{\theta}_{i,n})^2}])$.

By substituting Equation (20) and Equation (21) into Equation (18), we obtain the objective function in Theorem (2). □

Theorem 3. *We denote $[\bar{\mathbf{C}}_r]_{-k} \in \mathbb{R}^{m \times (n-1)}$ as the sub-matrix of $\bar{\mathbf{C}}_r$ by removing the column $[\bar{\mathbf{C}}_r]_k$, i.e. $[\bar{\mathbf{C}}_r]_{-k} = [[\bar{\mathbf{C}}_r]_1, \cdots, [\bar{\mathbf{C}}_r]_{k-1}, [\bar{\mathbf{C}}_r]_{k+1}, \cdots, [\bar{\mathbf{C}}_r]_n]$, and define $\mathbf{W} = [\mathbf{w}_1, \cdots, \mathbf{w}_{m-n+1}] \in \mathbb{R}^{m \times (m-n+1)}$ as the orthonormal basis of the orthonormal complement of $[\bar{\mathbf{C}}_r]_{-k}$. If $\mathbf{u} \in \mathbb{R}^{(m-n+1) \times 1}$ is the eigenvector of $\mathbf{W}^{\mathrm{T}}\mathbf{S}(r,k)\mathbf{W}$ corresponding to the smallest eigenvalue, then $\mathbf{W}\mathbf{u}$ is the optimal solution of $[\bar{\mathbf{C}}_r]_k$ for Equation (9).*

Proof. Since $[\bar{\mathbf{C}}_r]_k^{\mathrm{T}} \bar{\mathbf{C}}_{r,o} = 0$ for all $1 \leq o \leq n, o \neq k$, then $[\bar{\mathbf{C}}_r]_k$ lies in the orthonormal complement of the space spanned by the columns of $[\bar{\mathbf{C}}_r]_{-k}$. Thus, there exists a vector $\mathbf{u} \in \mathbb{R}^{(m-n+1) \times 1}$ satisfying $[\bar{\mathbf{C}}_r]_k = \mathbf{W}\mathbf{u}$ and $\mathbf{u}^{\mathrm{T}}\mathbf{u} = 1$. The objective function in Equation (9) becomes $\mathbf{u}^{\mathrm{T}}(\mathbf{W}^{\mathrm{T}}\mathbf{S}(r,k)\mathbf{W})\mathbf{u}$. Obviously, the optimal $\mathbf{u}$ for minimizing this function is the eigenvector of the matrix $\mathbf{W}^{\mathrm{T}}\mathbf{S}(r,k)\mathbf{W}$ corresponding to the smallest eigenvalue. □

2. The Algorithm for Learning the Symmetric Transition Matrix of LDSST

As for LDSs, given the observed sequence, several methods [13, 12] have been proposed to learn the optimal solutions of the system parameters, while the method presented in [3] is widely used. In this approach, the measurement matrix and the hidden states can be estimated as $\mathbf{C} = \mathbf{U}$ and $\mathbf{X} = \mathbf{S}\mathbf{V}^{\mathrm{T}}$ by performing the singular value decomposition of the centered matrix $\mathbf{Y}' = [\mathbf{y}_1 - \bar{\mathbf{y}}, \cdots, \mathbf{y}_T - \bar{\mathbf{y}}] = \mathbf{U}\mathbf{S}\mathbf{V}^{\mathrm{T}}$. Using the least-square method on the hidden state, the transition matrix $\mathbf{A}$ is then computed as $\mathbf{A} = \mathbf{X}_1\mathbf{X}_0^{\dagger}$, where $\mathbf{X}_0 = [\mathbf{x}_1, \cdots, \mathbf{x}_{T-1}]$, $\mathbf{X}_1 = [\mathbf{x}_2, \cdots, \mathbf{x}_T]$, and † denotes the pseudo-inverse. The state covariance is given by $\mathbf{Q} = \mathbf{B}\mathbf{B}^{\mathrm{T}} = \frac{1}{T-1}\sum_{t=1}^{T-1} \mathbf{v}'_t(\mathbf{v}'_t)^{\mathrm{T}}$ where $\mathbf{v}'_t = \mathbf{B}\mathbf{v}_t = \mathbf{x}_{t+1} - \mathbf{A}\mathbf{x}_t$. By computing the singular value decomposition $\mathbf{Q} = \mathbf{U}'\mathbf{S}'\mathbf{V}'^{\mathrm{T}}$, we obtain $\mathbf{B} = \mathbf{U}'\mathbf{S}'^{1/2}$.

In our dictionary learning algorithm presented in Section 4, we constrain the transition matrices of the data systems to be symmetric, meaning that we should use the LDS model with Symmetric Transition matrix (LDSST) to model the spatio-temporal data. For LDSSTs, learning the measurement matrix $\mathbf{C}$ is identical with that of LDSs, while learning the transition matrix $\mathbf{A}$ is different due to the imposed symmetric constraint. Recalling that, $\mathbf{A}$ is learned by solving the least-square problem in the hidden space of LDS, we can derive $\mathbf{A}$ in LDSST as follows

$$\min_{\mathbf{A}} \parallel \mathbf{X}_1 - \mathbf{A}\mathbf{X}_0 \parallel_F^2, \qquad \text{s.t. } \mathbf{A}^{\mathrm{T}} = \mathbf{A}, \tag{22}$$

where $\mathbf{X}_0 = [\mathbf{x}_1, \cdots, \mathbf{x}_{T-1}]$, $\mathbf{X}_1 = [\mathbf{x}_2, \cdots, \mathbf{x}_T]$. Since $\mathbf{A}$ is symmetric, there exists an orthonormal square matrix $\mathbf{P}$ satisfying $\mathbf{A} = \mathbf{P}\mathbf{\Theta}\mathbf{P}^{\mathrm{T}}$, where the matrix $\mathbf{\Theta}$ is diagonal with the elements being the eigenvalues of the matrix $\mathbf{A}$. Then Equation (22) is reduced to

$$\min_{\mathbf{P},\mathbf{\Theta}} \parallel \mathbf{X}_1 - \mathbf{P}\mathbf{\Theta}\mathbf{P}^{\mathrm{T}}\mathbf{X}_0 \parallel_F^2, \qquad \text{s.t. } \mathbf{P}^{\mathrm{T}}\mathbf{P} = \mathbf{I}_n. \tag{23}$$

We first reduce the objective function in Equation (23) as follows

$$\begin{aligned}
& \parallel \mathbf{X}_1 - \mathbf{P}\mathbf{\Theta}\mathbf{P}^{\mathrm{T}}\mathbf{X}_0 \parallel_F^2 \\
= \; & \mathrm{Tr}\left((\mathbf{X}_1 - \mathbf{P}\mathbf{\Theta}\mathbf{P}^{\mathrm{T}}\mathbf{X}_0)(\mathbf{X}_1^{\mathrm{T}} - \mathbf{X}_0^{\mathrm{T}}\mathbf{P}\mathbf{\Theta}\mathbf{P}^{\mathrm{T}})\right) \\
= \; & \mathrm{Tr}\left(\mathbf{X}_1\mathbf{X}_1^{\mathrm{T}} - \mathbf{P}\mathbf{\Theta}\mathbf{P}^{\mathrm{T}}\mathbf{X}_0\mathbf{X}_1^{\mathrm{T}} - \mathbf{X}_1\mathbf{X}_0^{\mathrm{T}}\mathbf{P}\mathbf{\Theta}\mathbf{P}^{\mathrm{T}} + \mathbf{P}\mathbf{\Theta}\mathbf{P}^{\mathrm{T}}\mathbf{X}_0\mathbf{X}_0^{\mathrm{T}}\mathbf{P}\mathbf{\Theta}\mathbf{P}^{\mathrm{T}}\right) \\
= \; & \mathrm{Tr}(\mathbf{X}_1\mathbf{X}_1^{\mathrm{T}}) - 2\mathrm{Tr}(\mathbf{P}\mathbf{\Theta}\mathbf{P}^{\mathrm{T}}\mathbf{X}_0\mathbf{X}_1^{\mathrm{T}}) + \mathrm{Tr}(\mathbf{P}\mathbf{\Theta}\mathbf{P}^{\mathrm{T}}\mathbf{X}_0\mathbf{X}_0^{\mathrm{T}}\mathbf{P}\mathbf{\Theta}\mathbf{P})^{\mathrm{T}} \\
= \; & \mathrm{Tr}(\mathbf{X}_1\mathbf{X}_1^{\mathrm{T}}) - 2\mathrm{Tr}(\mathbf{\Theta}\mathbf{P}^{\mathrm{T}}\mathbf{X}_0\mathbf{X}_1^{\mathrm{T}}\mathbf{P}) + \mathrm{Tr}(\mathbf{\Theta}\mathbf{P}^{\mathrm{T}}\mathbf{X}_0\mathbf{X}_0^{\mathrm{T}}\mathbf{P}\mathbf{\Theta}) \\
= \; & \mathrm{Tr}(\mathbf{X}_1\mathbf{X}_1^{\mathrm{T}}) - 2\sum_{k=1}^{n}([\mathbf{P}]_k^{\mathrm{T}}\mathbf{X}_1\mathbf{X}_0^{\mathrm{T}}[\mathbf{P}]_k)\boldsymbol{\theta}_k + \sum_{k=1}^{n}([\mathbf{P}]_k^{\mathrm{T}}\mathbf{X}_0\mathbf{X}_0^{\mathrm{T}}[\mathbf{P}]_k)\boldsymbol{\theta}_k^2,
\end{aligned} \tag{24}$$

where $\boldsymbol{\theta}_k$ is the k-th diagonal element of the matrix $\mathbf{\Theta}$.

Ignoring the term $\mathrm{Tr}(\mathbf{X}_1\mathbf{X}_1^{\mathrm{T}})$ that is irrelevant to both $\mathbf{P}$ and $\mathbf{\Theta}$, Equation (23) is further reduced to

$$\begin{aligned}
\min_{\mathbf{P},\mathbf{\Theta}} \quad & \sum_{k=1}^{n}([\mathbf{P}]_k^{\mathrm{T}}\mathbf{X}_0\mathbf{X}_0^{\mathrm{T}}[\mathbf{P}]_k)\boldsymbol{\theta}_k^2 - 2\sum_{k=1}^{n}([\mathbf{P}]_k^{\mathrm{T}}\mathbf{X}_1\mathbf{X}_0^{\mathrm{T}}[\mathbf{P}]_k)\boldsymbol{\theta}_k, \\
\text{s.t.} \quad & \mathbf{P}^{\mathrm{T}}\mathbf{P} = \mathbf{I}_n.
\end{aligned} \tag{25}$$

We can break the minimization problem in Equation (25) into n sub-problems by updating the pair $([\mathbf{P}]_k, \boldsymbol{\theta}_k)$ once at a time with the values of other pairs $\{([\mathbf{P}]_o, \boldsymbol{\theta}_o)\}_{o=1,o\neq k}^{n}$ fixed. However, once the values of other columns $\{[\mathbf{P}]_o\}_{o=1,o\neq k}^{n}$ are given , the value of the column $[\mathbf{P}]_k$ is determined due to the orthonormal constraint of the matrix $\mathbf{P}$. It means that no update of $[\mathbf{P}]_k$ can be performed if we break the problem in Equation (25) into sub-problems in this way.

Instead, we can find the locally-optimal solution of the minimization problem in the recursive process: we first update the pair $([\mathbf{P}]_1, \boldsymbol{\theta}_1)$ by relaxing the orthonormal constraint of the matrix $\mathbf{P}$; once we obtain the optimal values of the next pair $([\mathbf{P}]_1, \boldsymbol{\theta}_1)$, we keep updating the values of the pair $([\mathbf{P}]_2, \boldsymbol{\theta}_2)$ by imposing the constraint that the column $[\mathbf{P}]_2$ needs to lie in the orthonormal complement of the updated column, *i.e* $[\mathbf{P}]_1$. This process does not halt until we obtain the values of the final pair $([\mathbf{P}]_n, \boldsymbol{\theta}_n)$.

Given the updated pairs $\{([\mathbf{P}]_o, \boldsymbol{\theta}_o)\}_{o=1}^{k-1}$ where $2 \leq k \leq n$, finding the optimal $([\mathbf{P}]_k, \boldsymbol{\theta}_k)$ in Equation (25) formulates the sub-problem

$$\begin{aligned}
\min_{[\mathbf{P}]_k,\boldsymbol{\theta}_k} \quad & ([\mathbf{P}]_k^{\mathrm{T}}\mathbf{X}_0\mathbf{X}_0^{\mathrm{T}}[\mathbf{P}]_k)\boldsymbol{\theta}_k^2 - ([\mathbf{P}]_k^{\mathrm{T}}\mathbf{X}_1\mathbf{X}_0^{\mathrm{T}}[\mathbf{P}]_k)\boldsymbol{\theta}_k, \\
\text{s.t.} \quad & [\mathbf{P}]_k^{\mathrm{T}}[\mathbf{P}]_k = 1, \; [\mathbf{P}]_k^{\mathrm{T}}[\mathbf{P}]_o = 0, \; 1 \leq o \leq k-1.
\end{aligned} \tag{26}$$

An efficient approach for solving this problem is to alternately update $\mathbf{P}_k$ and $\boldsymbol{\theta}_k$. Since the objective function is a quadratic function with respect to $\boldsymbol{\theta}_k$, the optimal $\boldsymbol{\theta}_k$ is obtained by the following equation with $\mathbf{P}_k$ fixed

$$\boldsymbol{\theta}_k \quad = \quad \frac{[\mathbf{P}]_k^{\mathrm{T}}\mathbf{X}_1\mathbf{X}_0^{\mathrm{T}}[\mathbf{P}]_k}{2[\mathbf{P}]_k^{\mathrm{T}}\mathbf{X}_0\mathbf{X}_0^{\mathrm{T}}[\mathbf{P}]_k}. \tag{27}$$

Reversely, the optimal $\mathbf{P}_k$ can be easily derived in a similar way as the proof of Theorem (3), when the value of $\boldsymbol{\theta}_k$ is given. Specifically,

$$[\mathbf{P}]_k = \mathbf{W}_k \cdot SE(\mathbf{W}_k^{\mathrm{T}}(\boldsymbol{\theta}_k^2\mathbf{X}_0\mathbf{X}_0^{\mathrm{T}} - \boldsymbol{\theta}_k\mathbf{X}_1\mathbf{X}_0^{\mathrm{T}})\mathbf{W}_k), \tag{28}$$

Algorithm 2 Learning the symmetric transition of LDSST

```
Input: $\mathbf{X}_0$, $\mathbf{X}_1$
Initialize the transition matrix $\mathbf{A} = \mathbf{X}_1\mathbf{X}_0^{\dagger}$
Perform the singular value decomposition $\mathbf{A} = \mathbf{U}\mathbf{S}\mathbf{V}^{\mathrm{T}}$
Initialize $\mathbf{P}$ and $\mathbf{\Theta}$ as $\mathbf{P} = \mathbf{U}$ and $\mathbf{\Theta} = \mathbf{S}$
Set the number of the iterations for optimizing each pair as $I = nIters$
for $k = 1$ to $n$ do
  if $k = 1$ then
    $\mathbf{W}_k = \mathbf{I}$
  else
    Assign the value of the matrix $\mathbf{W}_k$ with the orthonormal complement of $[[\mathbf{P}]_1, \cdots, [\mathbf{P}]_{k-1}]$
  end if
  for $i = 1$ to $I$ do
      $\boldsymbol{\theta}_k = \frac{[\mathbf{P}]_k^{\mathrm{T}}\mathbf{X}_1\mathbf{X}_0^{\mathrm{T}}[\mathbf{P}]_k}{2[\mathbf{P}]_k^{\mathrm{T}}\mathbf{X}_0\mathbf{X}_0^{\mathrm{T}}[\mathbf{P}]_k}$
      $[\mathbf{P}]_k = \mathbf{W}_k \cdot SE(\mathbf{W}_k^{\mathrm{T}}(\boldsymbol{\theta}_k^2\mathbf{X}_0\mathbf{X}_0^{\mathrm{T}} - \boldsymbol{\theta}_k\mathbf{X}_1\mathbf{X}_0^{\mathrm{T}})\mathbf{W}_k)$
  end for
end for
$\mathbf{A} = \mathbf{P}\mathbf{\Theta}\mathbf{P}^{\mathrm{T}}$
```

where $SE(\bullet)$ denotes the function to find the smallest eigenvector of the input matrix; and $\mathbf{W}_k$ consists of the orthonormal basis of the orthonormal complement of the updated vectors $\{[\mathbf{P}]_o\}_{o=1}^{k-1}$. When $k = 1$, we can update $[\mathbf{P}]_1$ with Equation (28) by setting $\mathbf{W}_1 = \mathbf{I}$.

For reader's convenience, we list the procedures in Algorithm (2). Since updating each pair scales $O(I(T-1)n^2)$, the computational complexity of this algorithm is $O(I(T-1)n^3)$. It is comparable with the complexity of learning parameters in LDS, which scales $O((T-1)n^2 + n^3)$.

Usually, the transition matrix $\mathbf{A}$ in LDSST (also in LDS) learned from the data sequences is not assured to be stable. A practical but not deliberate method to guarantee $\mathbf{A}$ to be stable is to divide $\mathbf{A}$ with a factor, *i.e.* $\mathbf{A}' = \rho\mathbf{A}$, where $\rho > 1$. Practically, $\rho \in \{1.1, 1.2, 1.3\}$. We can also add the stable constraint to Equation (23) to formulate a new delicate problem, which is interesting but beyond the scope of the paper.

3. The Reconstruction Error Approach

The learned codes from sparse coding defined Equation (4) can be adopted as features for classification by computing the reconstruction error of the dictionary atoms for each class, if the dictionary atoms are labeled. Denoting the dictionary atoms labeled as class c to be $\{\mathbf{D}_k^{(c)}\}_{k=1}^{K_c}$, where K_c is the total number of the dictionary atoms, the reconstruction error of $\mathbf{V}$ is defined as

$$e_c(\mathbf{V}) \quad = \quad \| \mathbf{V}\mathbf{V}^{\mathrm{T}} - \sum_{j=1}^{K_c} \mathbf{z}_k^{(c)}\mathbf{D}_k^{(c)}(\mathbf{D}_k^{(c)})^{\mathrm{T}} \|_F^2, \tag{29}$$

where $\mathbf{z}_k^{(c)}$ is the coefficient associated with atom $\mathbf{D}_k^{(c)}$. The label of $\mathbf{V}$ is then determined as the class that yields the minimal reconstruction error.

Equation (29) can be easily extended for the models that considering the state covariance by adding the reconstructed errors of the covariance terms, *i.e.*

$$e_c(\mathbf{V}) \quad = \quad \beta \| \mathbf{V}\mathbf{V}^{\mathrm{T}} - \sum_{j=1}^{K_c} \mathbf{z}_k^{(c)}\mathbf{D}_k^{(c)}(\mathbf{D}_k^{(c)})^{\mathrm{T}} \|_F^2 + (1-\beta) \| \mathbf{\Omega} - \sum_{j=1}^{K_c} \mathbf{z}_k^{(c)}\bar{\mathbf{\Omega}}_k^{(c)} \|_F^2, \tag{30}$$

where $\mathbf{\Omega}$ and $\bar{\mathbf{\Omega}}_k^{(c)}$ denote the one-step covariances of the data and the $k^{(c)}$-th dictionary, respectively; β is the weight parameter.

Table 1. The specification of the benchmark datasets.

Datasets	#Sequences	Spatial size	#Frames	#Classes
Cambridge	900	320 × 240	37-119	9
UCSD	254	48 × 48	42-52	3
CK+	327	640 × 480	10-60	7
DynTex++	3600	50 × 50	50	36
SD	100	27 × 18	325-526	10
SPR	97	8 × 16	503-549	10
BDH	100	8 × 9	203-486	2

4. Benchmark Datasets

In this section, we give the detailed information of the benchmark datasets: *Cambridge*, *UCSD*, *CK+*, *DynTex++*, *SD*, *SPR*, and *BDH*. For reader's convenience, we provide the specification of the datasets in Table 1.

4.1. Cambridge

The *Cambridge* hand gesture dataset [8] consists of 900 images sequences of 9 gesture classes generated by 3 primitive hand shapes and 3 primitive motions. Each class contains 100 image sequences performed by 2 subjects, with 10 arbitrary camera motions and under 5 illumination conditions. Samples of the images are demonstrated in Figure 1. The spatial size of each original image is 320 × 240. Similar to [6], we resize all images to be 20 × 20, and adopt the first 80 images of each class for testing while the remaining for training.

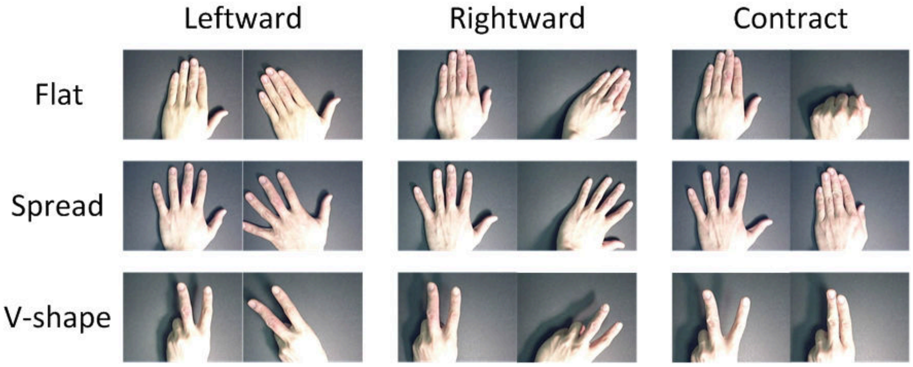

Figure 1. Examples of *Cambridge*. The image sequences are performed by 3 primitive hand shapes with 3 primitive motions.

4.2. UCSD

The experiment of scene analysis is performed on *UCSD* traffic dataset [2], which consists of 254 video sequences of highway traffic with a variety of traffic patterns in various weather conditions. Each video is recoded with a resolution of 320 × 240 pixels for a duration between 42 and 52 frames. The clipped version that has been resized to the scale of 48 × 48 is applied in this experiment. The dataset is labeled into three classes with respect to the severity of traffic congestion in each video. The total numbers of the sequences of heavy traffic, medium traffic and light traffic are 44, 45 and 165, respectively. Four random divisions of this dataset have been performed by the authors in [2]. In each division, 75% of the sequences are used for training and the rest 25% for testing.

4.3. CK+

The extended Cohn-Kanade (*CK+*) database is a development for research in automatic facial image analysis and synthesis [9]. There are 593 sequences of 7 basic emotions (Anger, Contempt, Disgust, Fear, Happiness, Sadness and Surprise) across

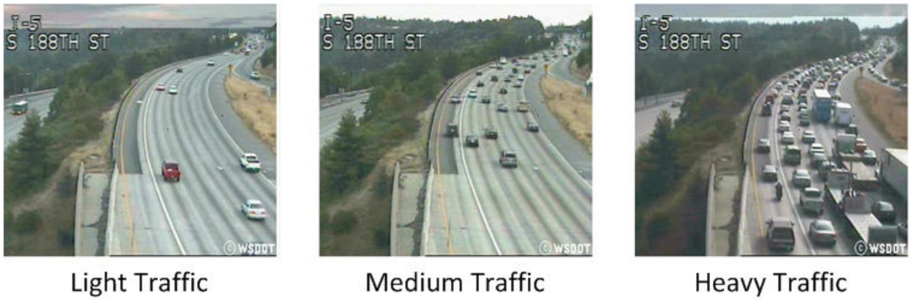

Figure 2. Representative examples of the three classes in *UCSD* traffic dataset.

123 subjects. Each sequence begins with a neutral expression and proceeds to a peak expression. All sequences are AAM tracked with 68-point landmarks for each image. Suggested by [4], we only use the 327 image sequences that have emotion patterns for experiments. Besides, we utilize the extracted 68-point landmarks of each image as the input features.

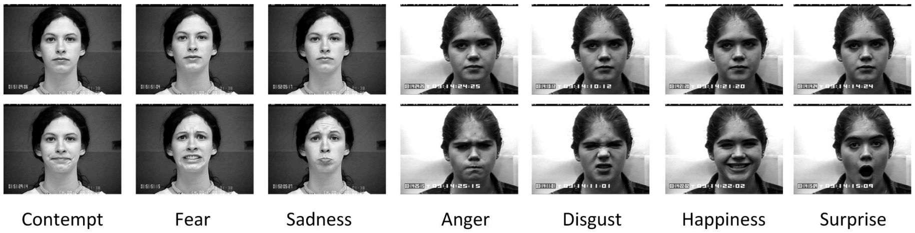

Figure 3. Examples of *CK+*. There are 7 basic emotions: Contempt, Fear, Sadness, Anger, Disgust, Happiness and Surprise.

4.4. DynTex++

Dynamic textures are video sequences of complex scenes that exhibit certain stationary properties in the time domain, such as water on the surface of a lake, a flag fluttering in the wind, swarms of birds, humans in crowds, etc. The constant change in the geometry of the these objects poses a challenge for applying traditional vision algorithms to these video sequences. The dataset *DynTex++* [5] is a variant of the original *DynTex* [11]. It contains 3600 videos of 36 classes each of which is comprised of 100 videos with a fixed size of $50 \times 50 \times 50$. In this paper, we apply the same test protocol as [5], namely, half of the videos are applied as the training set and the other half as the testing set.

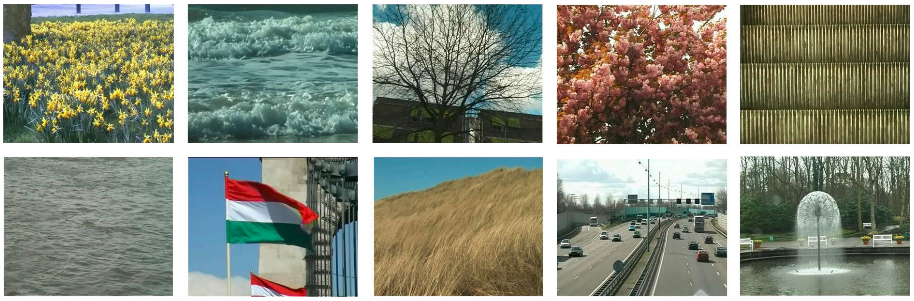

Figure 4. Representative classes of *DynTex*: Flowers, Sea, Naked trees, Foliage, Escalator, Calm water, Flags, Grass, Traffic, and Fountains.

4.5. Tactile Datasets

Recognizing the object that the robot grasps via the tactile series is an active research area in robotics [10]. The tactile series obtained from the force sensors can be used to determine some properties of the object like shape and softness. For our experiments, the recognition tasks are evaluated on three datasets: *SD* [14], *SPR* [10] and *BDH* [10]. *SD* contains 100 tactile series of 10 household objects grasped by the 3-finger Schunk Dextrous Hand (SDH). The *SPR* dataset composed of 97 sequences has the same object classes as *SD*, but is generated with the 3-finger Schunk Dextrous Hand (SDH). *BDH* consists of 100 tactile sequences generated by controlling the BH8-280 Hand to grasp 5 different bottles with water or without water, as illustrated in Figure 5. The task is to predict whether the bottle is empty or is filled with water based on the tactile sequences.

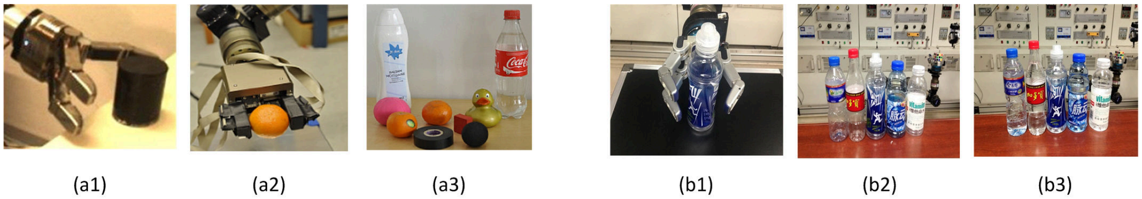

Figure 5. The used robot hands and the grasping objects in *SD*, *SPR* and *BDH*. (a1) The 3-finger Schunk Dextrous Hand (SDH); (a2) The 2-finger Schunk Parallel Hand (SPH); (a3) Objects in *SD* and *SPR*: Rubber ball, Balsam bottle, Rubber duck, Empty bottle, Full bottle, Bad orange, Fresh orange, Joggling ball, Tape, and Wood block. (b1) The BH8-280 Hand; (b2) Bottles without water; (b3) Bottles with water.

5. Online dictionary learning for anormaly detection

We also explore our dictionary learning algorithm for anormaly detection of dynamical scene. To this end, we perform experiments on the subset of the coastal surveillance dataset *PETS2005* [1], *i.e.* *ZOD4*, which consists of 5100 images collected by thermal cameras with a resolution of 640×480. These images record the dynamical waves of the sea and the movement of a very small shipping container. There are some people on shore moving back and forth. *ZOD4* is originally employed for object tracking. Here we apply it for anormaly detection of the dynamical scene, such as the case when the people move and obscure the view. We take 300 images numbered from 0200 to 0500 for analysis. These images are further resized to 160×120. In this task, we adopt covLDSST-DL in an online-learning manner. Every 20 consecutive images without overlap formulate a video. We split the current video into 100 sub-blocks with a spatial size of 16×12 and then learn the dictionary of the sub-blocks. When the next video comes, we reconstruct it with the learned dictionary. The reconstructed error defined in Equation (12), *i.e.* $\beta L_{mean} + (1-\beta)L_{cov}$, is utilized for evaluation. A sudden large error implies an obvious anormaly condition. Then, we retrain the dictionary with the coming video until the last video arrives. Figure 6 shows that covLDSST-DL can successfully detect the anormaly condition when the people emerge in the scene.

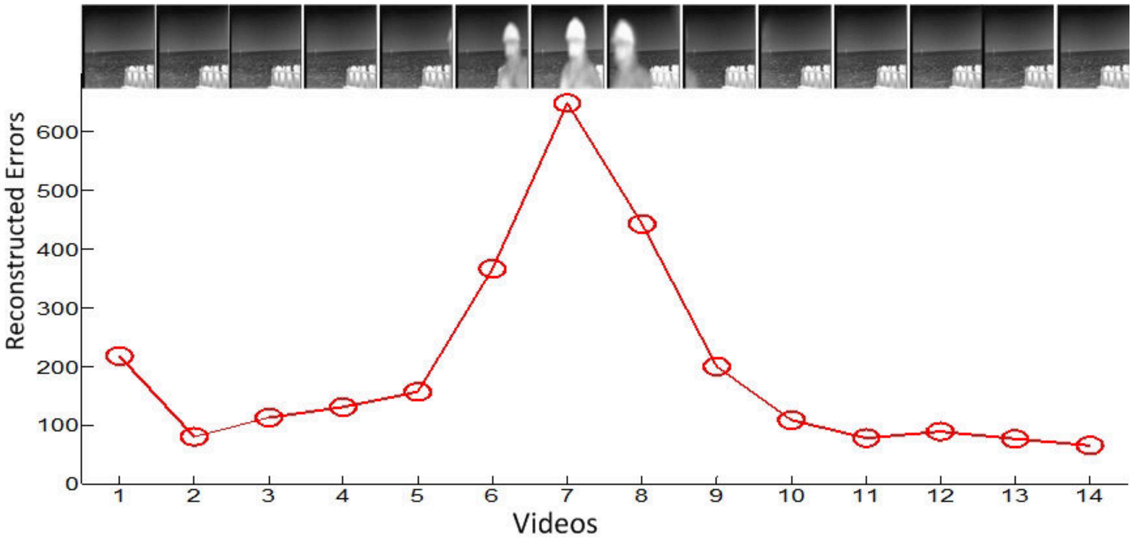

Figure 6. The errors of reconstructing each video with the online-learned dictionary on *ZOD4*. $(n, \beta, n_v) = (5, 0.8, 4)$.

References

[1] T. Boult. Coastal surveillance datasets. vision and security lab, u. colorado at colorado springs, 2005. 9

[2] A. B. Chan and N. Vasconcelos. Probabilistic kernels for the classification of auto-regressive visual processes. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 846–851. IEEE, 2005. 7

[3] G. Doretto, A. Chiuso, Y. N. Wu, and S. Soatto. Dynamic textures. *International Journal of Computer Vision (IJCV)*, 51(2):91–109, 2003. 4

[4] X. Fan and T. Tjahjadi. A spatial-temporal framework based on histogram of gradients and optical flow for facial expression recognition in video sequences. *Pattern Recognition*, 2015. 8

[5] B. Ghanem and N. Ahuja. Maximum margin distance learning for dynamic texture recognition. In *European Conference on Computer Vision (ECCV)*, pages 223–236. Springer, 2010. 8

[6] M. Harandi and M. Salzmann. Riemannian coding and dictionary learning: Kernels to the rescue. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015. 7

[7] M. Harandi, C. Sanderson, C. Shen, and B. Lovell. Dictionary learning and sparse coding on Grassmann manifolds: An extrinsic solution. In *IEEE International Conference on Computer Vision (ICCV)*, pages 3120–3127. IEEE, 2013. 2

[8] T.-K. Kim and R. Cipolla. Canonical correlation analysis of video volume tensors for action categorization and detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(8):1415–1428, 2009. 7

[9] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews. The extended Cohn-Kanade Dataset (CK+): A complete dataset for action unit and emotion-specified expression. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 94–101. IEEE, 2010. 7

[10] M. Madry, L. Bo, D. Kragic, and D. Fox. St-hmp: Unsupervised spatio-temporal feature learning for tactile data. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 2262–2269. IEEE, 2014. 8

[11] R. Péteri, S. Fazekas, and M. J. Huiskes. DynTex : a Comprehensive Database of Dynamic Textures. *Pattern Recognition Letters*, doi: 10.1016/j.patrec.2010.05.009. http://projects.cwi.nl/dyntex/. 8

[12] R. H. Shumway and D. S. Stoffer. An approach to time series smoothing and forecasting using the EM algorithm. *Journal of time series analysis*, 3(4):253–264, 1982. 4

[13] P. Van Overschee and B. De Moor. N4SID: Subspace algorithms for the identification of combined deterministic-stochastic systems. *Automatica*, 30(1):75–93, 1994. 4

[14] J. Yang, H. Liu, F. Sun, and M. Gao. Tactile sequence classification using joint kernel sparse coding. In *International Joint Conference on Neural Networks (IJCNN)*, 2015. 8