Varies of Least Squares Approaches

Introduction

- Least squares unmixing methods are widely used to solve linear mixture problems.
- A weighted least squares method is introduced as a generalization. With different weight matrix is used, a certain detector or classifier will be resulted.
- Constrained weighted least squares approach is developed by combining sum-to-one and nonnegativity constraints.

Previous Studies

- Linear Mixture Model

$$\mathbf{r} = M\boldsymbol{\alpha} + \mathbf{n}$$

where

$\mathbf{r}$ is a spectral signature of pixel vector

M is a signature matrix

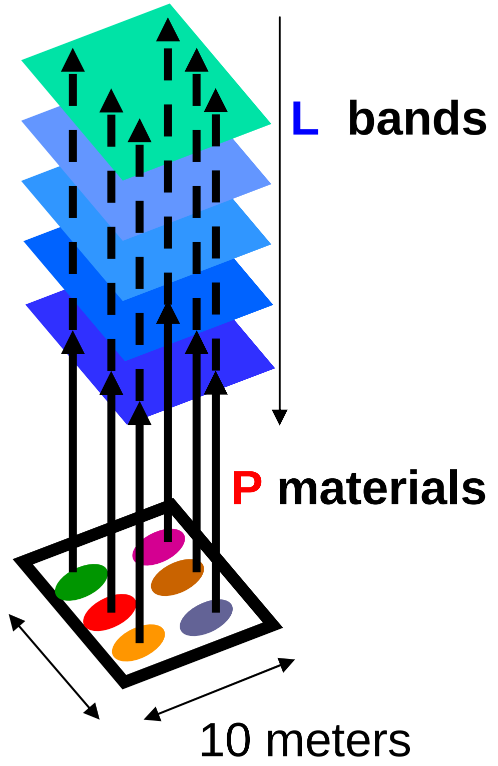
α is abundance column vector

$\mathbf{n}$ is an additive white Gaussian noise

Linear Spectral Mixture Analysis

$$r \quad = \quad M \quad \cdot \quad \alpha \quad + n$$

$$\begin{aligned}
r_1 &= m_{11}\alpha_1 + m_{21}\alpha_2 + \cdots + m_{P1}\alpha_P + n_1 \\
r_2 &= m_{12}\alpha_1 + m_{22}\alpha_2 + \cdots + m_{P2}\alpha_P + n_2 \\
&\vdots \\
r_L &= m_{1L}\alpha_1 + m_{2L}\alpha_2 + \cdots + m_{PL}\alpha_P + n_L
\end{aligned}$$



$L < P \Rightarrow$ infinitely number of solutions

$L = P \Rightarrow$ only one solution

$L > P \Rightarrow$ least square error solution

Least square error $\hat{\alpha}_{LS} = \left(M^T M\right)^{-1} M^T r$

Previous Studies (continued)

Unconstrained Least Square Unmixing

- Minimize least square error:

$$E(\boldsymbol{\alpha}) = \mathbf{n}^T\mathbf{n} = (\mathbf{r} - M\boldsymbol{\alpha})^T(\mathbf{r} - M\boldsymbol{\alpha})$$

- Solution:

$$\frac{\partial E(\boldsymbol{\alpha})}{\partial \boldsymbol{\alpha}} = 0 \qquad \hat{\boldsymbol{\alpha}}_{ULS} = (M^T M)^{-1} M^T \mathbf{r} = P_{ULS}\mathbf{r}$$

Weighted Least Square (WLS)

$$W\mathbf{r} = WM\boldsymbol{\alpha} + W\mathbf{n}$$

- Minimize least square error:

$$E(\boldsymbol{\alpha}) = \mathbf{n}^T\mathbf{n} = (W\mathbf{r} - WM\boldsymbol{\alpha})^T (W\mathbf{r} - WM\boldsymbol{\alpha})$$

- Solution:

$$\hat{\boldsymbol{\alpha}}_{WLS} = (M^T W^T WM)^{-1} M^T W^T W\mathbf{r}$$

Five Algorithms:

1. Least square solution $\rightarrow$ $W = I$ (identity matrix).
2. Noise whitened least square $\rightarrow$ $W = \Sigma_n^{-1/2}$ (Σ_n: noise covariance matrix).
3. Filter vector algorithm $\rightarrow$ $W^T W = I - \mathbf{uu}^T / L = S$
4. Target-constrained Interference-Minimized Filter $\rightarrow$ $W^T W = R^{-1}$ (*R*: Correlation matrix).
5. Constrained linear discriminant analysis $\rightarrow$ $W^T W = \Sigma^{-1}$ (Σ: covariance matrix).

Data Whitening

- Whitening:
  - Center the data:

$$\hat{\mathbf{X}} = \mathbf{X} - \boldsymbol{\mu} \cdot \mathbf{1}^T = [\mathbf{x}_1 - \boldsymbol{\mu}, \mathbf{x}_2 - \boldsymbol{\mu}, \cdots, \mathbf{x}_N - \boldsymbol{\mu}] \quad \text{where } \boldsymbol{\mu} = \frac{1}{N}\sum_{n=1}^{N} \mathbf{x}_n$$

  - Decorrelate the data:

$$\mathbf{A} = \mathbf{V}\boldsymbol{\Lambda}^{-1/2}$$

V is the eigenvector matrix of covariance matrix.

Λ is the matrix with eigenvalue in the diagonal line

- New data:

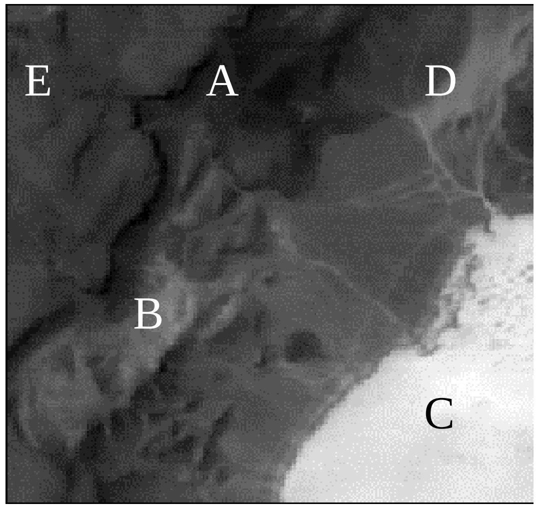
$$\mathbf{Y} = [\mathbf{y}_1 \mathbf{y}_2 \ldots \mathbf{y}_N] = \mathbf{A}^T (\mathbf{X} - \boldsymbol{\mu} \cdot \mathbf{1}^T)$$

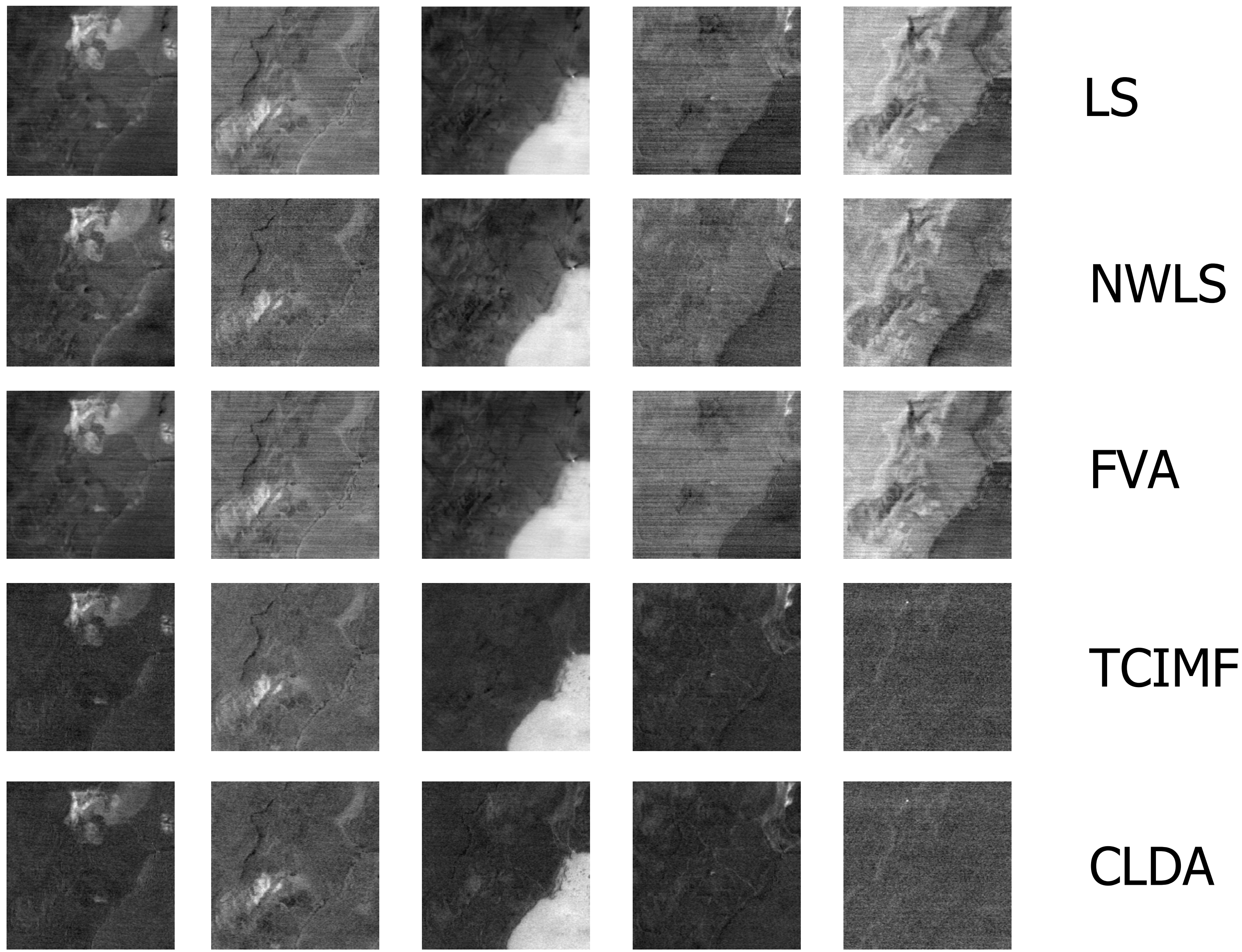
Experimental Results

- AVIRIS data:
  - Spatial resolution 20 m, 224 bands with spectral resolution 10 nm.
  - The AVIRIS scene of size 200×200 is from the Lunar Crater Volcanic Field (LCVF) in Northern Nye County, Nevada.

five signatures:

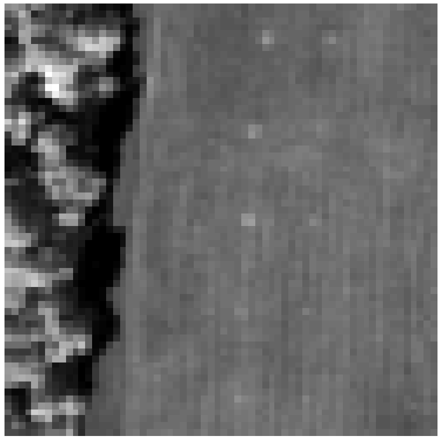
(A) cinder, (B) rhyolite
(C) playa, (D) vegetation
(E) shade

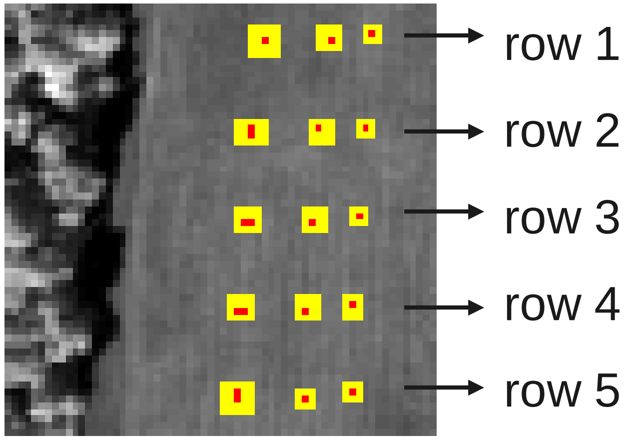


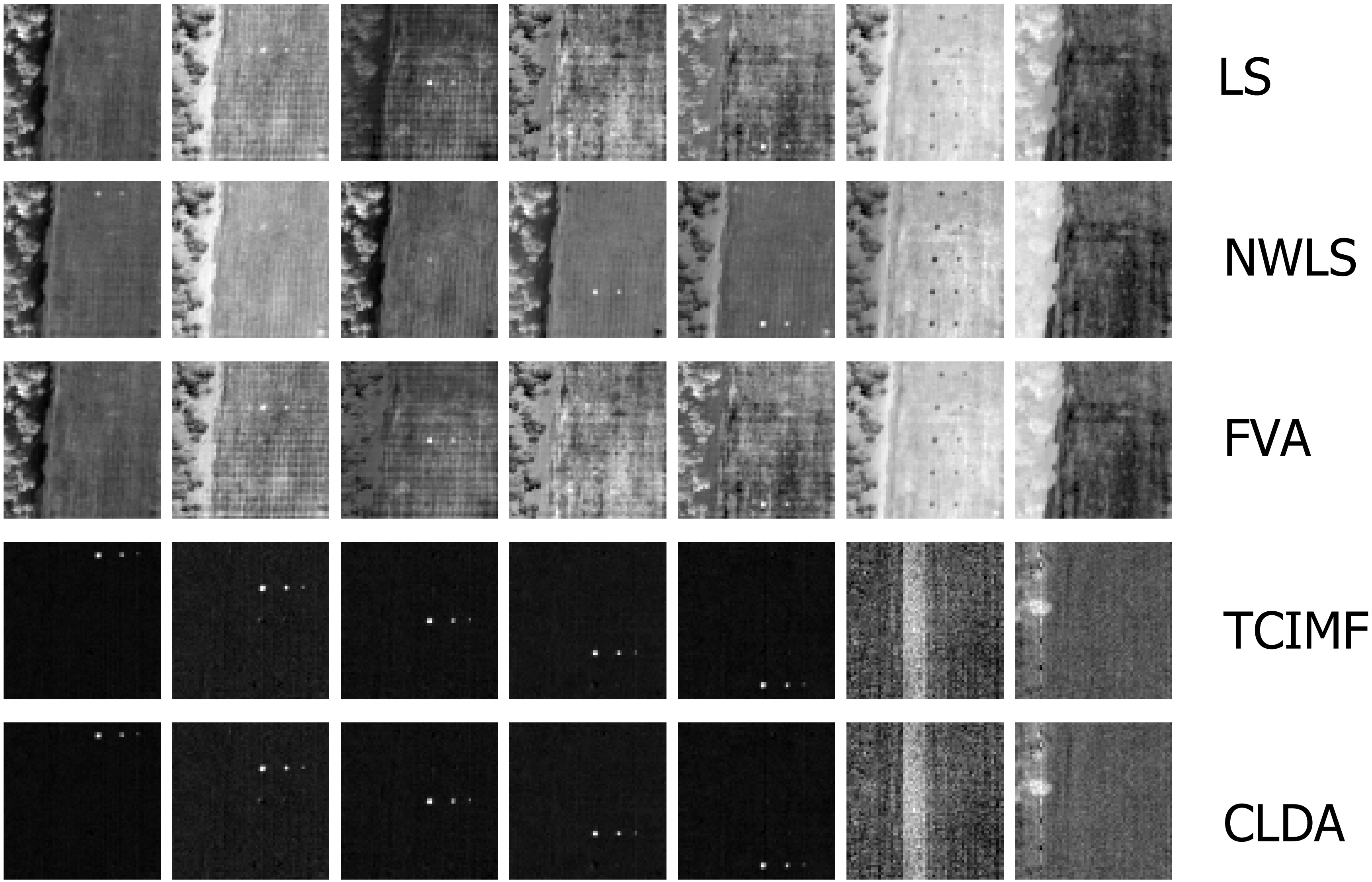

LS
NWLS
FVA
TCIMF
CLDA

- HYDICE data:
  - 210 bands from 0.4 – 2.5 μm.
  - Spatial resolution: 1.5 m.
  - Target of interests:

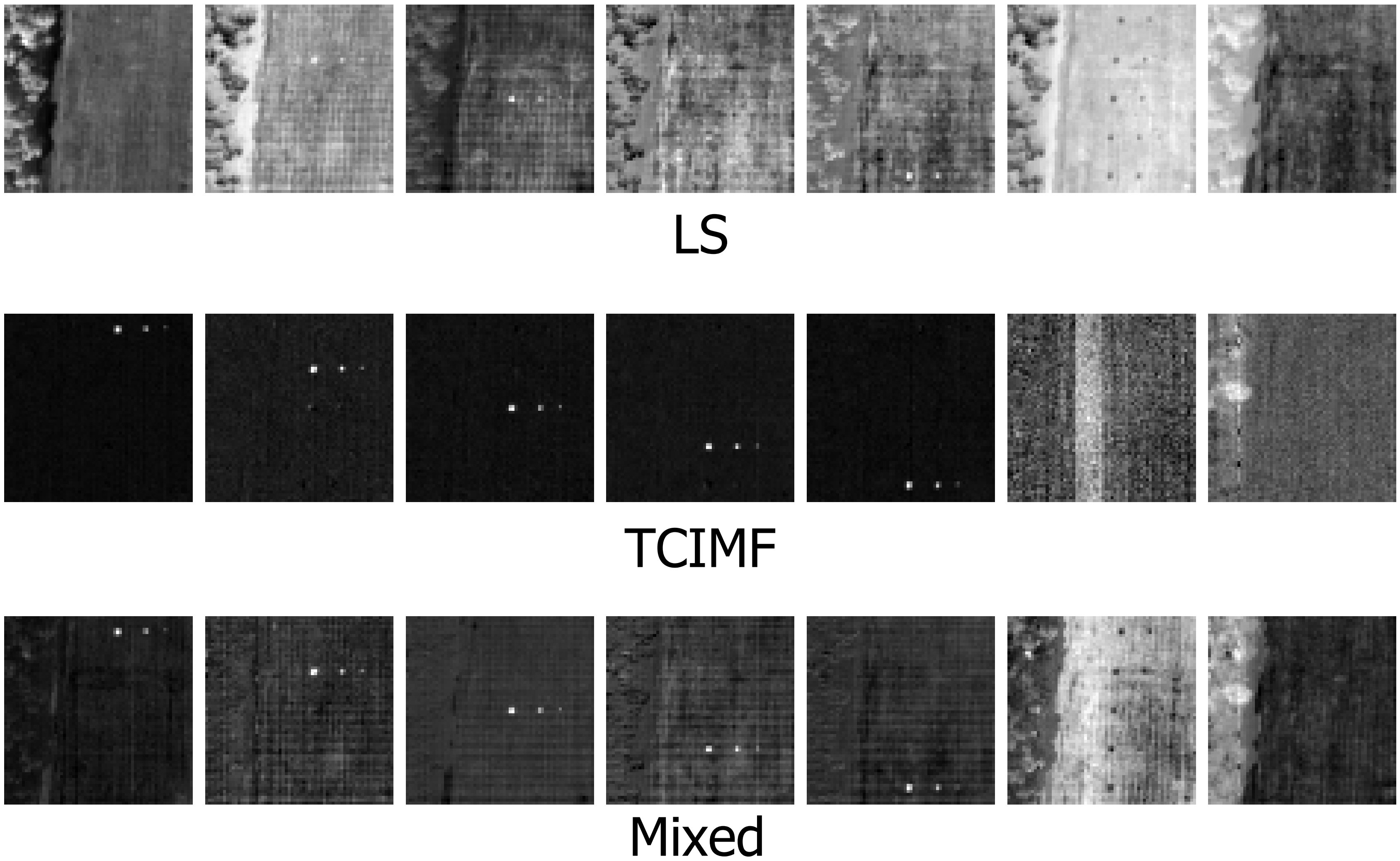
    Five types of panels, tree, grass







LS
NWLS
FVA
TCIMF
CLDA


LS
TCIMF
Mixed

Constraints

- A constrained linear unmixing requires that

$$\sum_{j=1}^{p} \alpha_j = 1 \quad \text{and} \quad \alpha_j \geq 0 \quad \text{for all} \quad 1 \leq j \leq p$$

- One method is to solve the constraints:

$$\sum_{j=1}^{p} \alpha_j = 1 \quad \text{and} \quad \sum_{j=1}^{p} \left| \alpha_j \right| = 1$$

Constrained WLS

- Two constrained:
  - Sum-to-one: $\sum_{j=1}^{p} \alpha_j = \mathbf{1}^T \boldsymbol{\alpha} = 1$
  - Nonnegativity: $\sum_{j=1}^{p} \left|\alpha_j\right| = sign(\boldsymbol{\alpha})^T \boldsymbol{\alpha} = 1$
- Lagrange Multiplier:

$$J = \frac{1}{2}(W\mathbf{r} - WM\boldsymbol{\alpha})(W\mathbf{r} - WM\boldsymbol{\alpha})^T - \lambda_1\left(\sum_{j=1}^{p} \alpha_j - 1\right) - \lambda_2\left(\sum_{j=1}^{p} \left|\alpha_j\right| - 1\right)$$

Sum-to-one Constrained Least Squares (SCLS) Unmixing

- Constraint:

$$\min_{\alpha \in \Delta_f} \left\{ (\mathbf{r} - M\alpha)(\mathbf{r} - M\alpha)^T \right\} \text{ subject to } \sum_{j=1}^{p} \alpha_j = 1$$

- Rewrite the equation:

$$\begin{bmatrix} \mathbf{r} \\ \lambda \end{bmatrix} = \begin{bmatrix} M \\ \lambda \mathbf{1}^T \end{bmatrix} \boldsymbol{\alpha} + \mathbf{n}'$$

and minimize the error:

$$E(\boldsymbol{\alpha}) = \mathbf{n}'^T \, \mathbf{n}' = (\mathbf{r}' - M'\boldsymbol{\alpha})^T (\mathbf{r}' - M'\boldsymbol{\alpha})$$

$$= (\mathbf{r} - M\boldsymbol{\alpha})^T (\mathbf{r} - M\boldsymbol{\alpha}) + \lambda^2 (1 - \mathbf{1}^T \boldsymbol{\alpha})^2$$

- Constraint:

$$\min_{\alpha \in \Delta_f} \left\{ (\mathbf{r} - M\alpha)(\mathbf{r} - M\alpha)^T \right\} \text{ subject to } \sum_{j=1}^{p} \alpha_j = 1 \text{ , } \sum_{j=1}^{p} \left| \alpha_j \right| = 1$$

- Rewrite the equation:

$$\begin{bmatrix} \mathbf{r} \\ \lambda_1 \\ \lambda_2 \end{bmatrix} = \begin{bmatrix} M \\ \lambda_1 \mathbf{1}^T \\ \lambda_2 \text{sign}(\boldsymbol{\alpha})^T \end{bmatrix} \boldsymbol{\alpha} + \mathbf{n}''$$

and minimize the error:

$$\begin{aligned} E(\boldsymbol{\alpha}) &= \mathbf{n}''^T \, \mathbf{n}'' = (\mathbf{r}'' - M''\boldsymbol{\alpha})^T (\mathbf{r}'' - M''\boldsymbol{\alpha}) \\ &= (\mathbf{r} - M\boldsymbol{\alpha})^T (\mathbf{r} - M\boldsymbol{\alpha}) + \lambda_1^2 (1 - \mathbf{1}^T \boldsymbol{\alpha})^2 + \lambda_2^2 (1 - \text{sign}(\boldsymbol{\alpha})^T \boldsymbol{\alpha})^2 \end{aligned}$$

Nonnegative:

$$\min_{\alpha \in \Delta_f} \left\{ (\mathbf{r} - M\alpha)(\mathbf{r} - M\alpha)^T \right\} \quad \text{subject to} \quad \alpha_j \geq 0$$

- Lagrange Multiplier:

$$J(\boldsymbol{\alpha}) = (\mathbf{r}' - M'\boldsymbol{\alpha})^T (\mathbf{r}' - M'\boldsymbol{\alpha}) + \lambda(\boldsymbol{\alpha} - \mathbf{c})$$

NCLS Algorithm

1. Initialization:

   Set the passive set $P^{(k)} = \{1, 2, \cdots, p\}$ and active set $R^{(0)} = \varnothing$. Set $k = 0$.

2. Compute $\hat{\alpha}_{LS}$ using (3.12). Let $\hat{\alpha}^{(k)}_{\text{NCLS}} = \hat{\alpha}_{LS}$

3. At the k-th iteration, if all components in $\hat{\alpha}^{(k)}_{\text{NCLS}}$ are positive, the algorithm is terminated; otherwise, continue.

4. Let $k \leftarrow k + 1$.

5. Move all indices in $P^{(k-1)}$ that correspond to negative components of $\hat{\alpha}^{(k-1)}_{\text{NCLS}}$ to $R^{(k-1)}$.

   Let the resulting index sets be denoted by $P^{(k)}$ and $R^{(k)}$ respectively. Create a new index set $S^{(k)}$ and set it equal to $R^{(k)}$.

6. Let $\hat{\alpha}_{R^{(k)}}$ denote the vector consisting of all components $\hat{\alpha}_{LS}$ in $R^{(k)}$.

7. Form a steering matrix $\Phi^{(k)}_{\alpha}$ by deleting all rows and columns in the matrix $\hat{\alpha}_{R^{(k)}}$ specified by $P^{(k)}$.

8. Calculate $\lambda^{(k)} = \left(\Phi^{(k)}_{\alpha}\right)^{-1} \hat{\alpha}_{R^{(k)}}$. If all components in $\lambda^{(k)}_{\max}$ are negative, go to step 13; otherwise, continue.

9. Calculate $\lambda^{(k)}_{\max} = \max_j \lambda^{(k)}_j$ and move its index in $R^{(k)}$ to $P^{(k)}$

10. Form another matrix $\Psi^{(k)}_{\lambda}$ by deleting every column of $\left(\mathbf{M}^T\mathbf{M}\right)^{-1}$ specified by $P^{(k)}$.

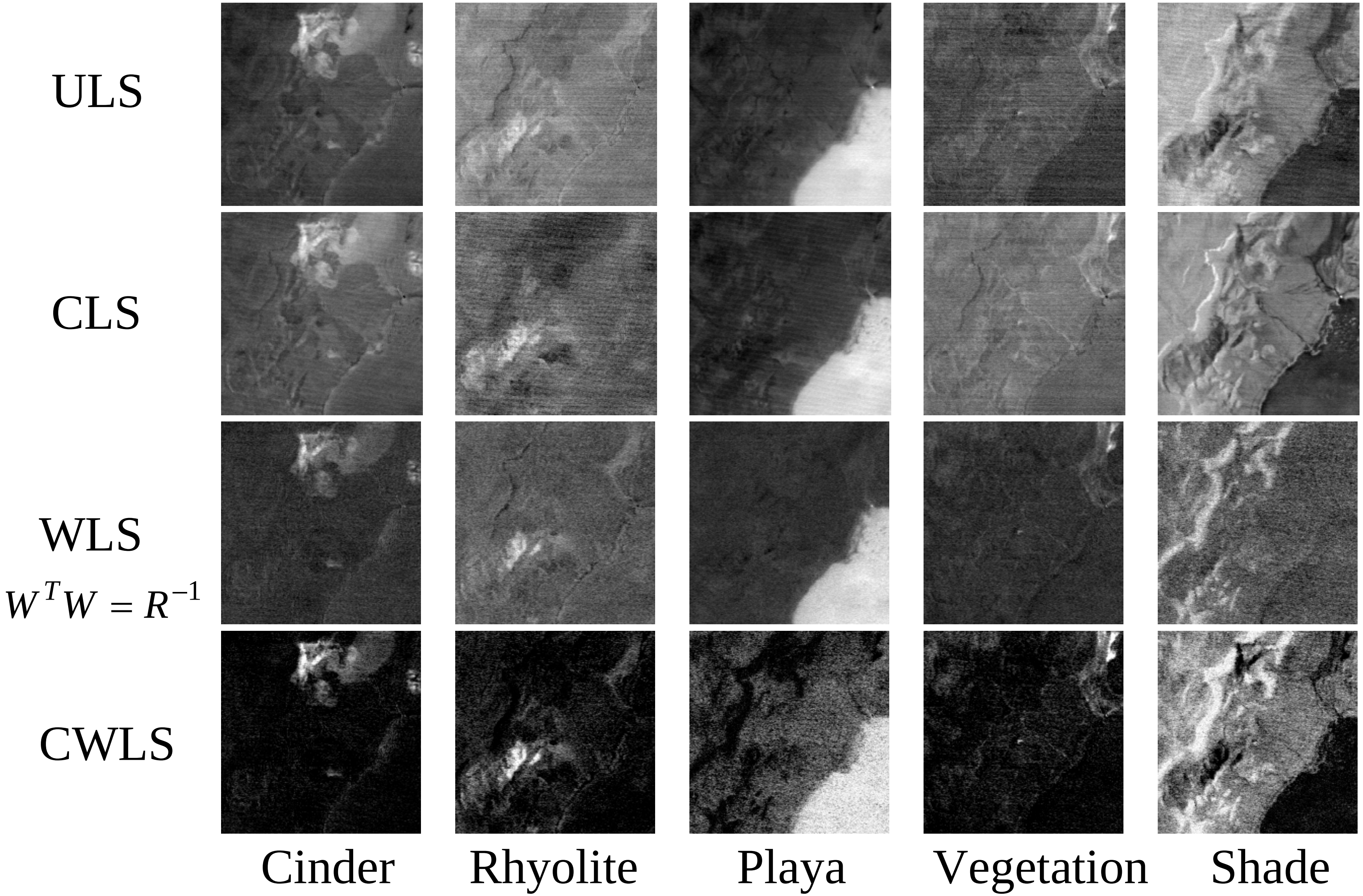
11. Set $\hat{\alpha}_{S^{(k)}} = \hat{\alpha}_{LS} - \Psi^{(k)}_{\lambda}\lambda^{(k)}$.

12. If any component of $\hat{\alpha}_{S^{(k)}}$ in $S^{(k)}$ is negative, then move it from $P^{(k)}$ to $R^{(k)}$ and go to step 6.

13. Form another matrix $\Psi^{(k)}_{\lambda}$ by deleting every column of $\left(\mathbf{M}^T\mathbf{M}\right)^{-1}$ specified by $P^{(k)}$.

14. Set $\hat{\alpha}^{(k)}_{\text{NCLS}} = \hat{\alpha}_{\text{LS}} - \Psi^{(k)}_{\lambda}\lambda^{(k)}$. Go to step 3.

Experimental Results



Unsupervised WLS

- Two stage processes:
  - ➢ Target Detection Process (TDP):

    Automatic search for potential targets.
  - ➢ Target Classification Process (TCP):

    Classify those targets using Least-Squares approaches.

Unsupervised WLS

Step 1 Search Endmember

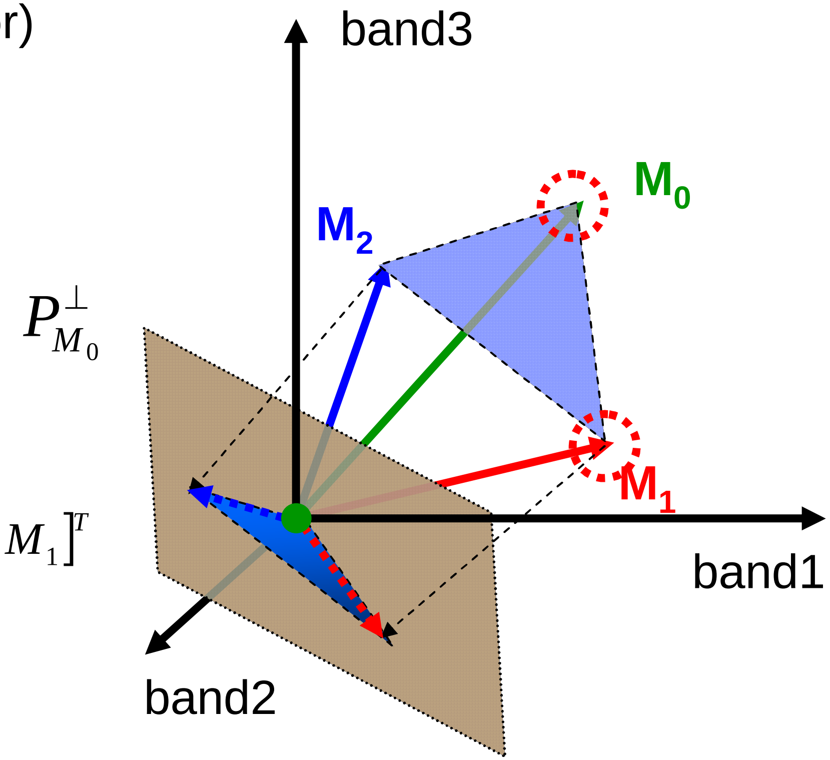
Step 2 Unmixing (Least square error)

Choose largest as M_0

$$P_{M_0}^{\perp} = I - M_0 \cdot (M_0^T \cdot M_0)^{-1} \cdot M_0^T$$

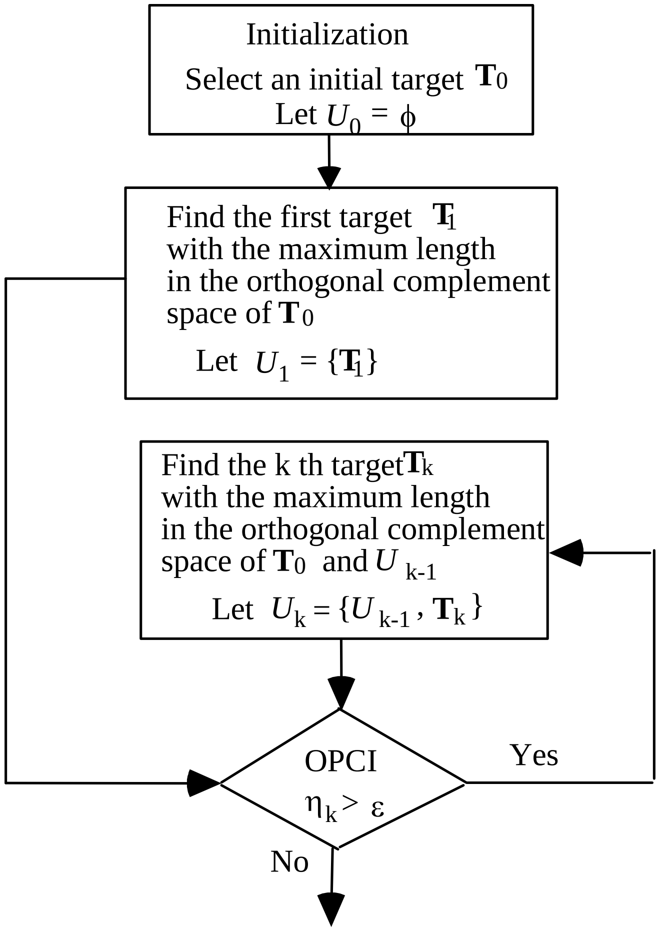
Choose largest as M_1

$$P_{M_0 M_1}^{\perp} = I - [M_0 M_1] \cdot ([M_0 M_1]^T \cdot [M_0 M_1])^{-1} \cdot [M_0 M_1]^T$$

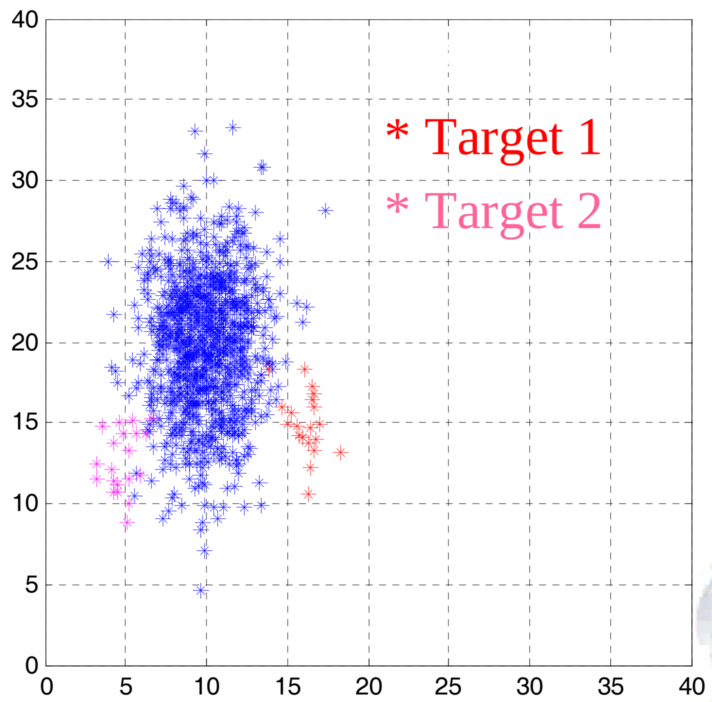


Least square error $\hat{\alpha}_{LS} = \left(M^T M\right)^{-1} M^T r$

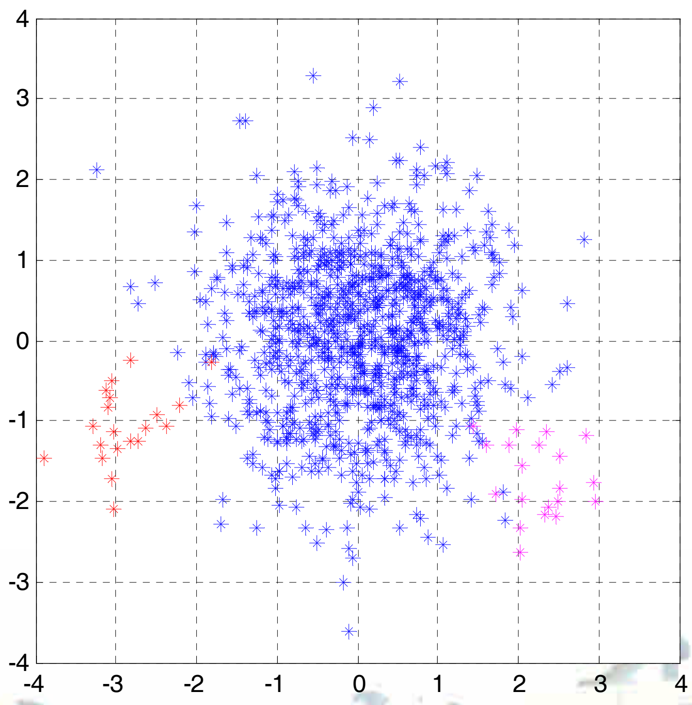
Target Detection Process



Data whitening (continued)

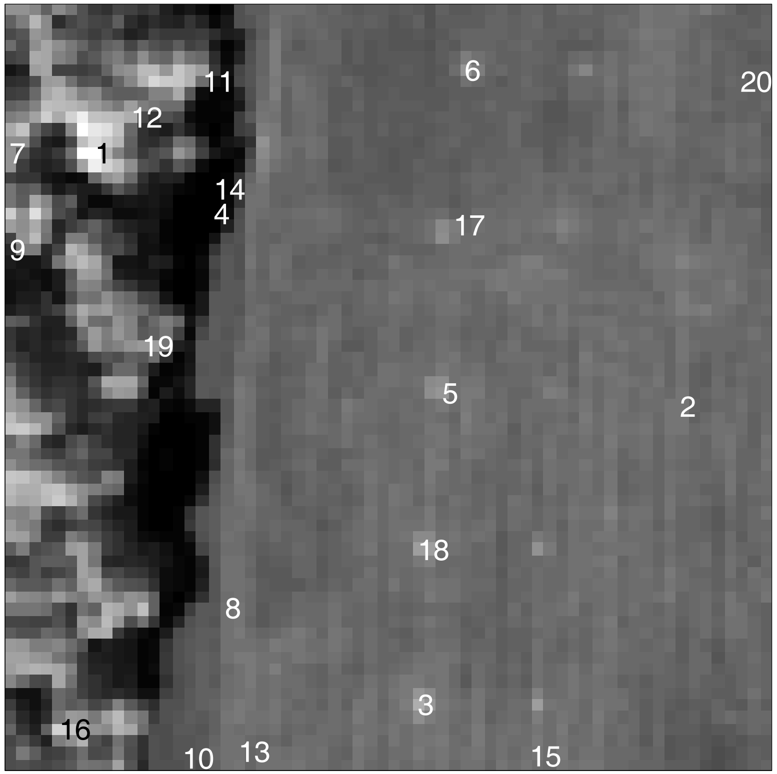


Original data

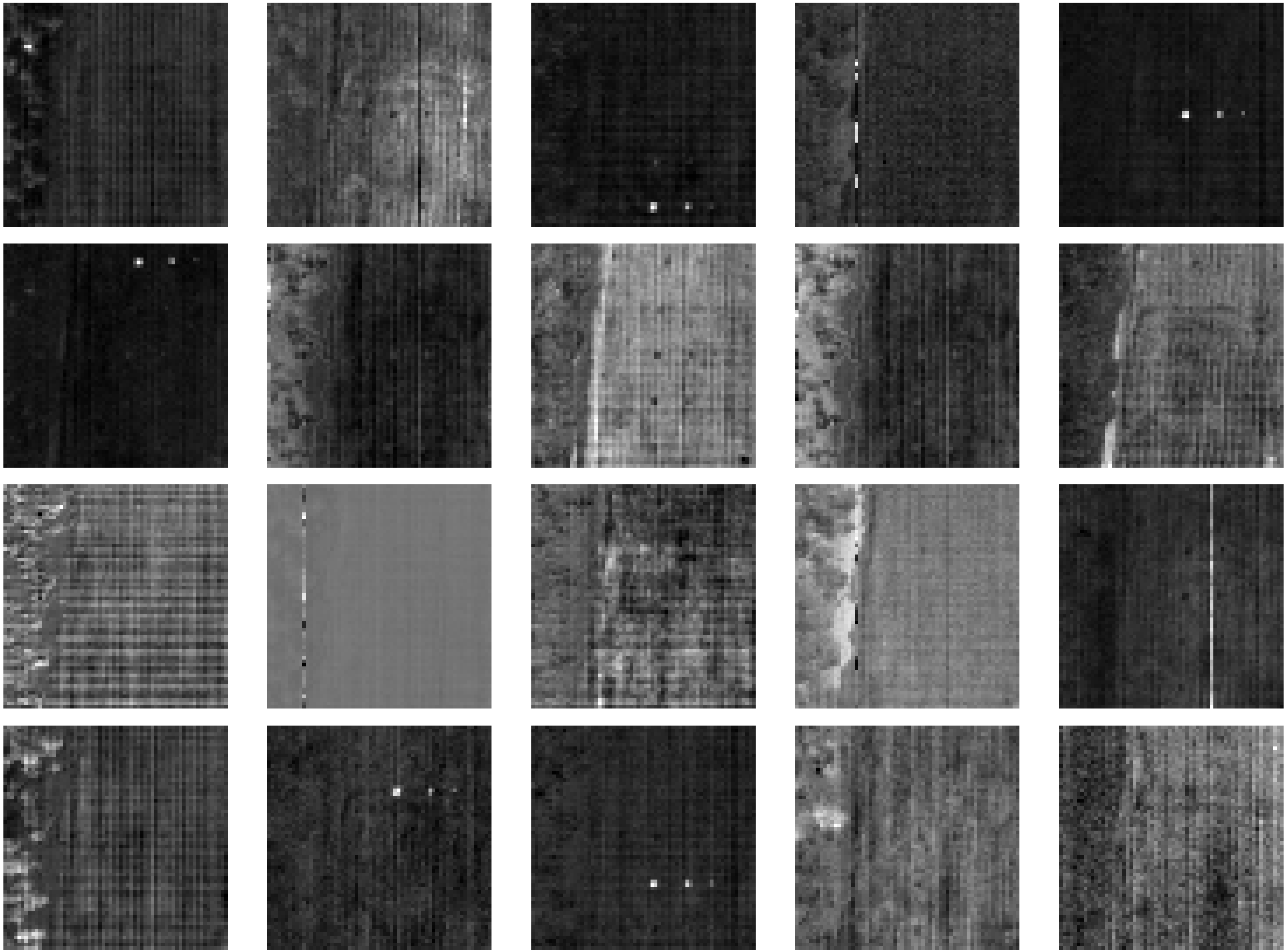


Whitened data

- Unsupervised LS:

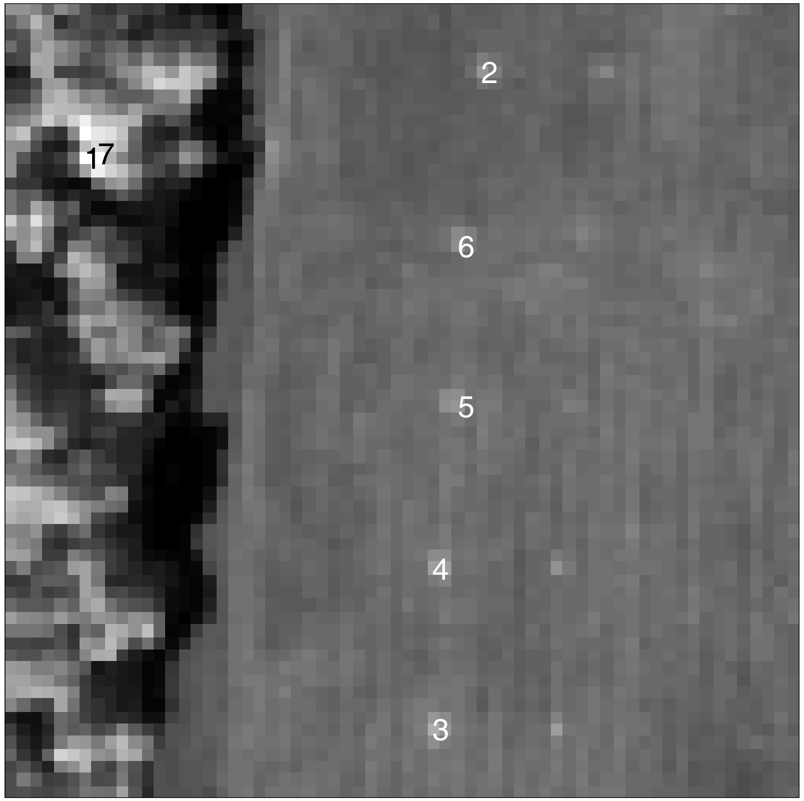


TDP finds background pixels as potential targets



- Unsupervised Weighted LS:

Background Whitened Target Detection Algorithm



- Background Whitened Target Detection Algorithm: