Hyperspectral Remote Sensing

Spectral Unmixing & Target Detection

University of Tehran

College of Engineering

Faculty of Surveying and Geospatial engineering

Remote Sensing

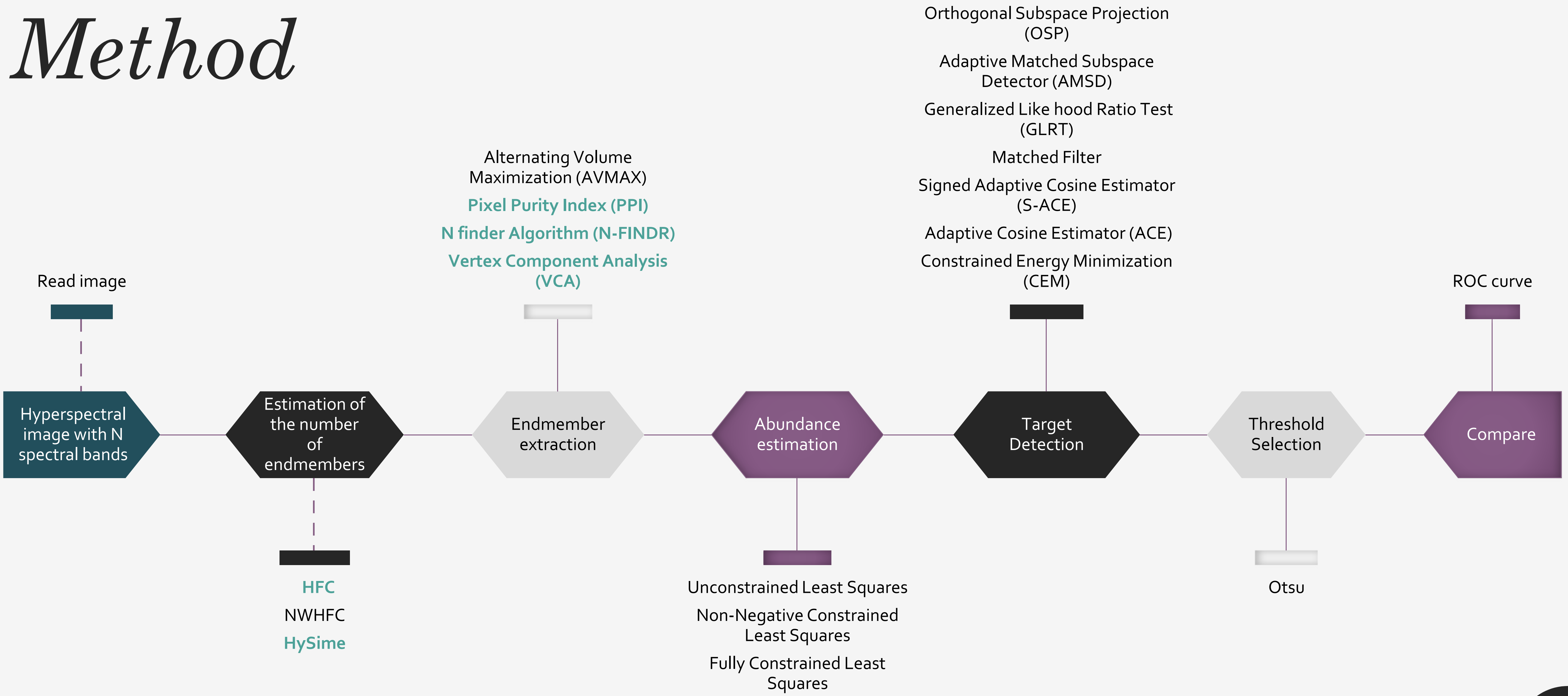
Dr. Hasanlou

Faezeh Zamiri

810399040

aghdam.zamiri.fa@ut.ac.ir

Method



- *General scheme of Spectral Unmixing And Target Detection*

Data

Data that I used in this project.

Link to Download dataset:

- https://rslab.ut.ac.ir/documents/81960329/105075217/Data.part1.rar
- https://rslab.ut.ac.ir/documents/81960329/105075217/Data.part2.rar
- https://rslab.ut.ac.ir/documents/81960329/105075217/Targets.rar

Read image

The Cooke City Dataset

(280*800) *126

Endmembers Estimation

Classic methods for subspace estimation is *Virtual dimensionality* ***HFC*** & ***NWHFC*** & *Hyperspectral subspace identification minimum error* (***HySime***)

1

HFC

2

NWHFC

3

HySime

Endmember extraction

Classic methods for endmember extraction

Vertex Component Analysis

(VCA)

Pixel Purity Index

(PPI)

N finder Algorithm

(N-FINDR)

Abundance estimation

Classic methods for abundance estimation:

Unconstrained Least Squares

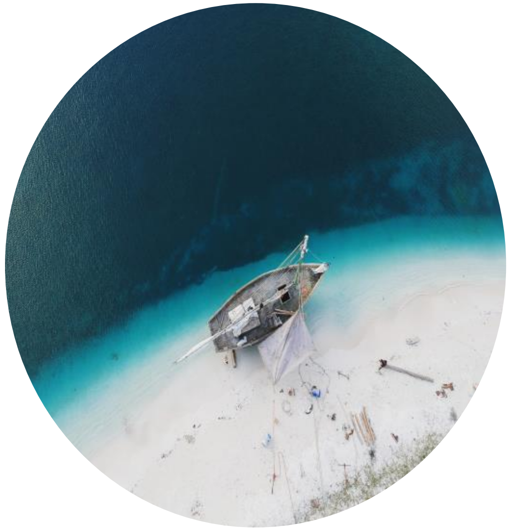
(Ucls)

Non-Negative Constrained Least Squares

(Nnls)

Fully Constrained Least Squares

(Fcls)



HFC

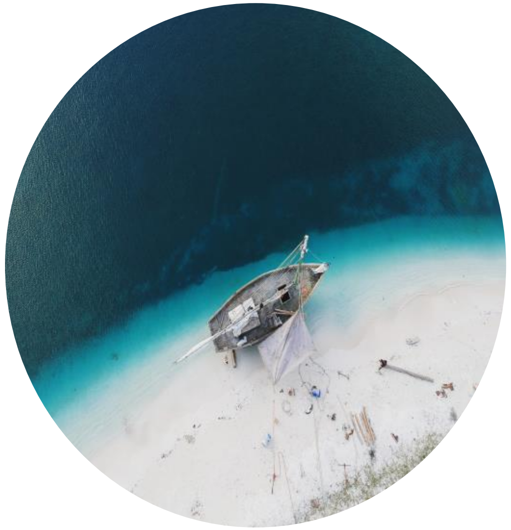
*Harsanyi-Farrand-Chang(**HFC**)*

$$\mathbf{R}_{L\times L} = \sum_{i=1}^{N} \mathbf{r}_i \mathbf{r}_i^T \qquad \{\hat{\lambda}_1 \geq \hat{\lambda}_2 \geq \cdots \geq \hat{\lambda}_L\}$$

$$\mathbf{K}_{L\times L} = \sum_{i=1}^{N} (\mathbf{r}_i - \boldsymbol{\mu})(\mathbf{r}_i - \boldsymbol{\mu})^T \qquad \{\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_L\}$$

$$\hat{\lambda}_l > \lambda_l \text{ for } l = 1, \ldots, \mathrm{VD}$$
$$\hat{\lambda}_l = \lambda_l \text{ for } l = \mathrm{VD}+1, \ldots, L$$

The HFC method was first envisioned in Harsanyi et al. (1994a) to detect spectral signatures present in AVIRIS data. It was then used to find the now-popular terminology, VD which is defined as the number of spectral distinct signatures and later published in Chang and Du (2004). It calculates the difference between eigenvlaues in sample correlation matrix and sample covariance matrix and makes use of Neyman–Pearson detector to determine the value of VD.



HySime

Endmembers Estimation

Hyperspectral subspace identification minimum error

The idea of HySime is to find the first k eigenvectors that contain the most of data information, i.e. , to find the k such that the mean square error (MSE) between the original data and its projection on to the eigenvector subspace is minimum.

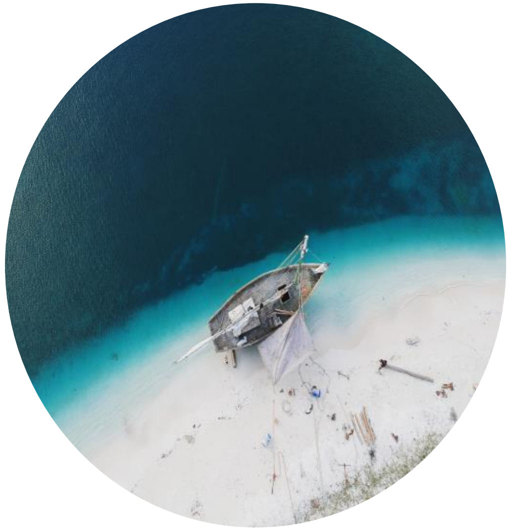
Subspace k is ranked in terms of data variance ,but noise variance is not unitary indifferent directions and the contribution from signals maybe smaller than from noise.

HySime addresses this issue using subspace projection techniques, thus bringing an additional feature with regards to VD :the modelling of noise before the estimation.

Result of Endmember Estimation

Method	Noise	The Cooke City Dataset
HFC	$P_f = 10^{-8}$	33
NWHFC	$P_f = 10^{-8}$	37
HySime	$P_f = 10^{-8}$	19

we know HySime is better than NWHFC and NWHFC is better than HFC , so that I used result of increase HySime for next step.



PPI

Pixel Purity Index

It is developed based on the concept of the convex geometry and the criterion of orthogonal projection.

It first generates a set of K random unit vectors, called skewers, to cover all possible projection directors and then orthogonally projects all data sample vectors on these skewers to find the maximal and minimal orthogonal projections of each skewer.

For each data sample vector, it counts the number of skewers on which its orthogonal projections yield either maximal or minimal projections. This count is referred to as the PPI count, which will be used to determine whether or not a particular data sample vector is an endmember.

Result of

PPI

HYPERPPI Performs the pixel purity index (PPI) algorithm

Performs the pixel purity index algorithm for endmember finding.

Usage

[U] = hyperPpi(M, q, numSkewers)

Inputs

M - 2d matrix of HSI data (p x N) ..... **Cooke City (126*224000)**

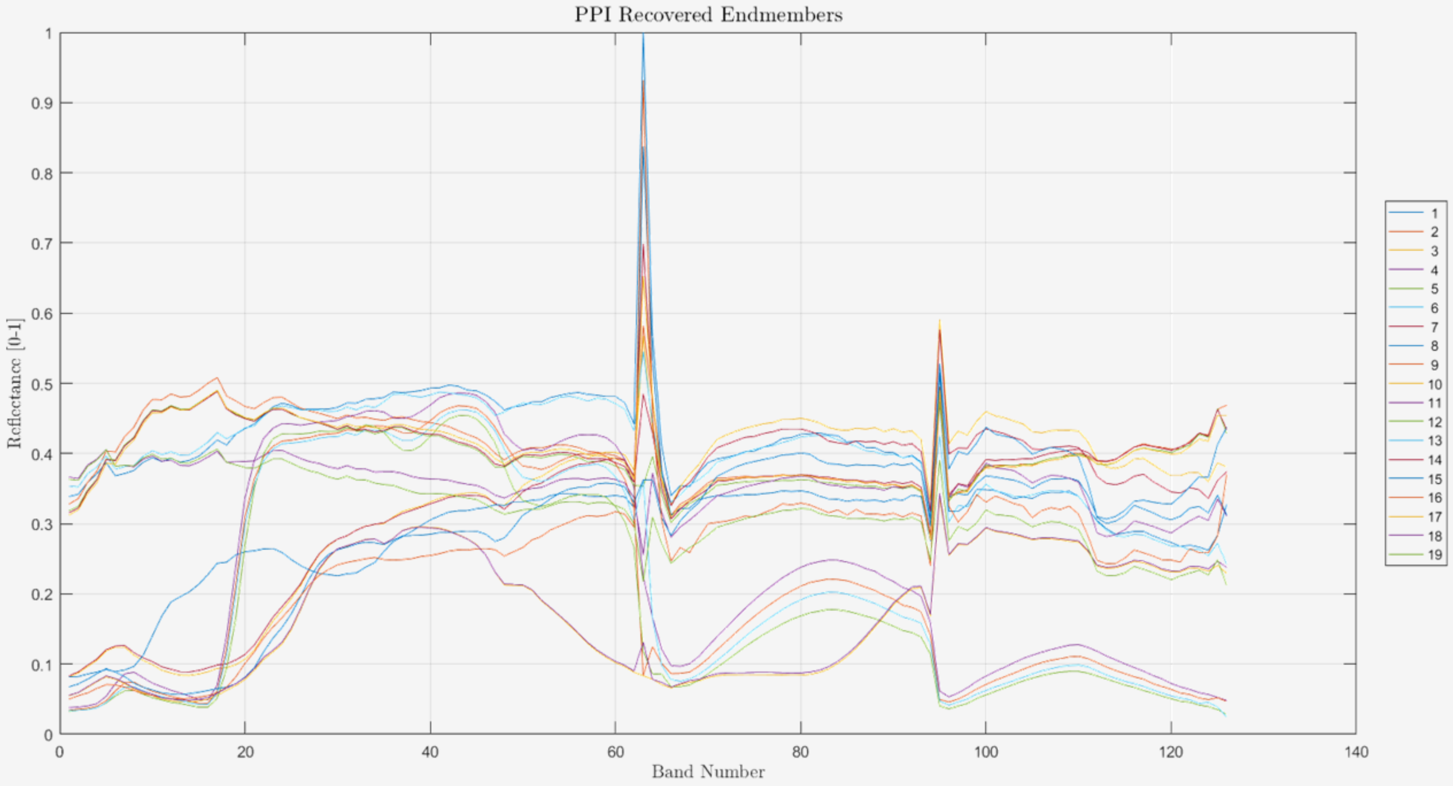
q - Number of endmembers to find **19**

numSkewers - Number of "skewer" vectors to project data onto **1000**

Outputs

U - Recovered endmembers (p x q) ...... **(126 * 19)**

Endmember extraction



Result of

N-Finder

HYPERNFINDR Performs the N-FINDR (endmember extraction) algorithm

Performs the N-FINDR algorithm to find q endmembers

and then reduce dimensionality to (q-1).

Usage

[U] = hyperNfinder(M, q)

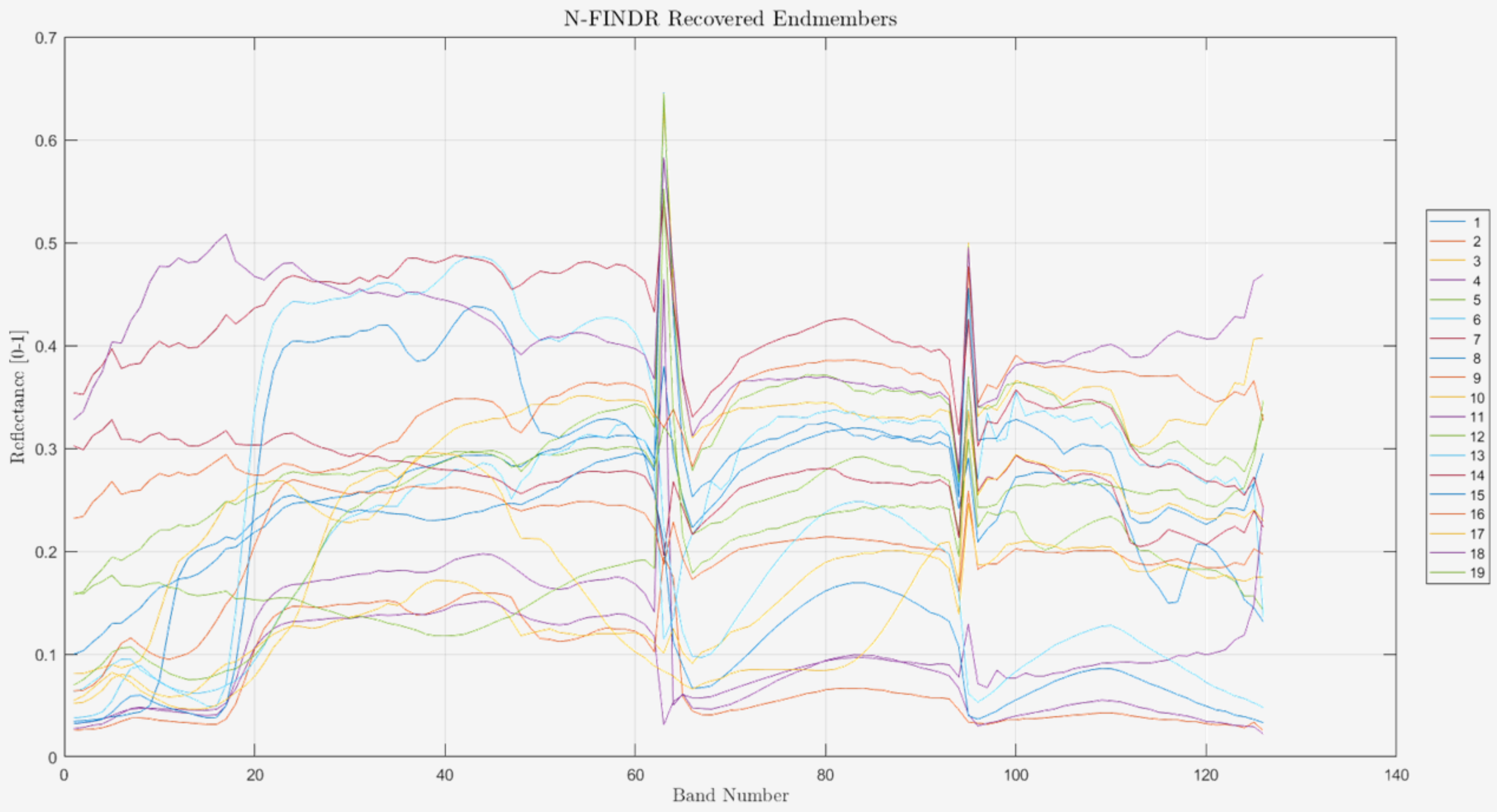
Inputs

M - 2d matrix of HSI data (p x N) ….. **Cooke City (126*224000)**

q - Number of endmembers to find ….. **19**

Outputs

U - Recovered endmembers (p x q) …… **(126*19)**



Endmember extraction

Result of

VCA

HYPERVCA Vertex Component Analysis algorithm

hyperVca performs the vertex component analysis algorithm to find pure pixels in an HSI scene

Usage

[U, indicies, snrEstimate] = hyperVca(M, q)

Inputs

M - HSI data as 2D matrix (p x N) ..... **Cooke City (126*224000)**

q - Number of endmembers to find **19**

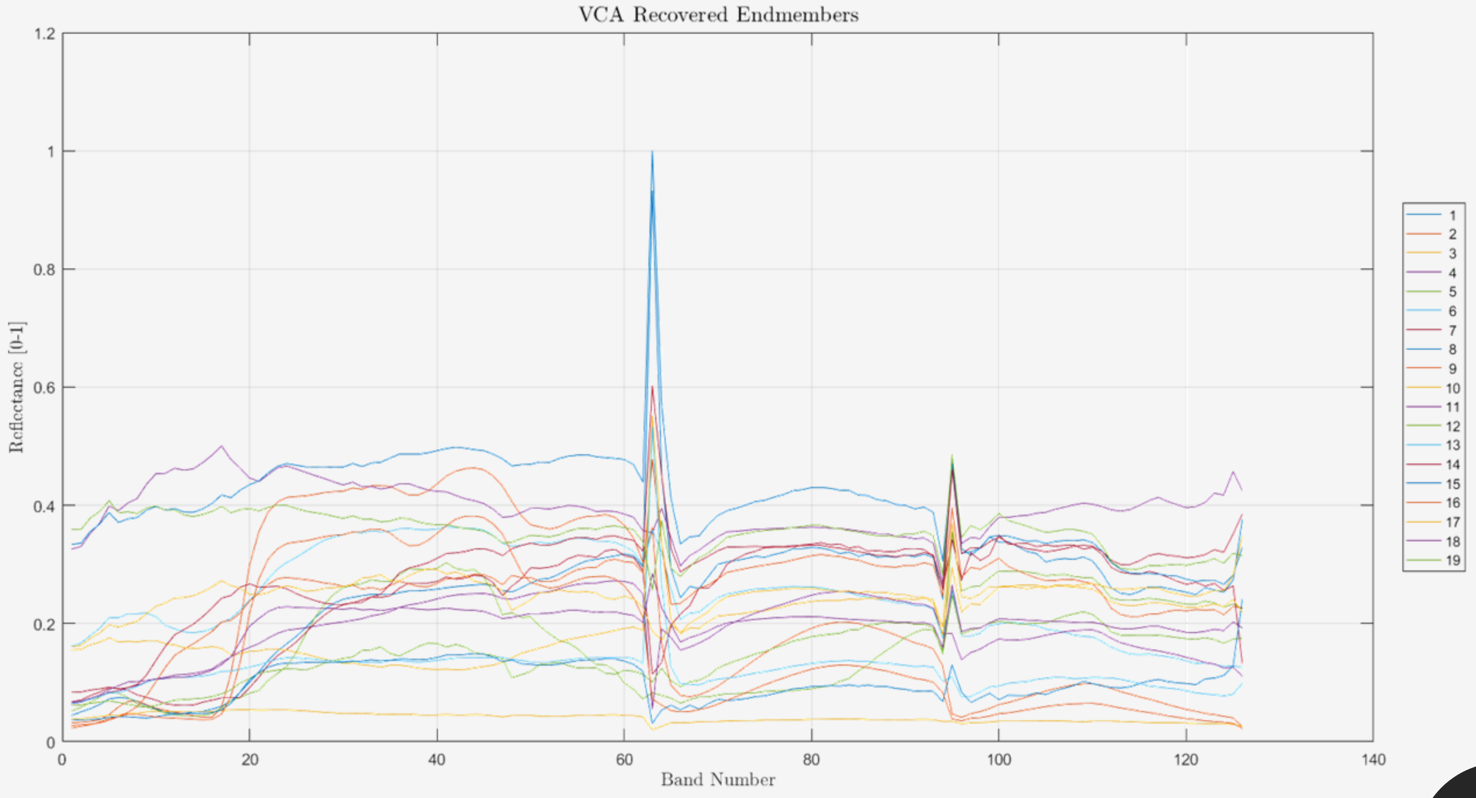
Outputs

U - Matrix of endmembers (p x q)...... **(126 * 19)**

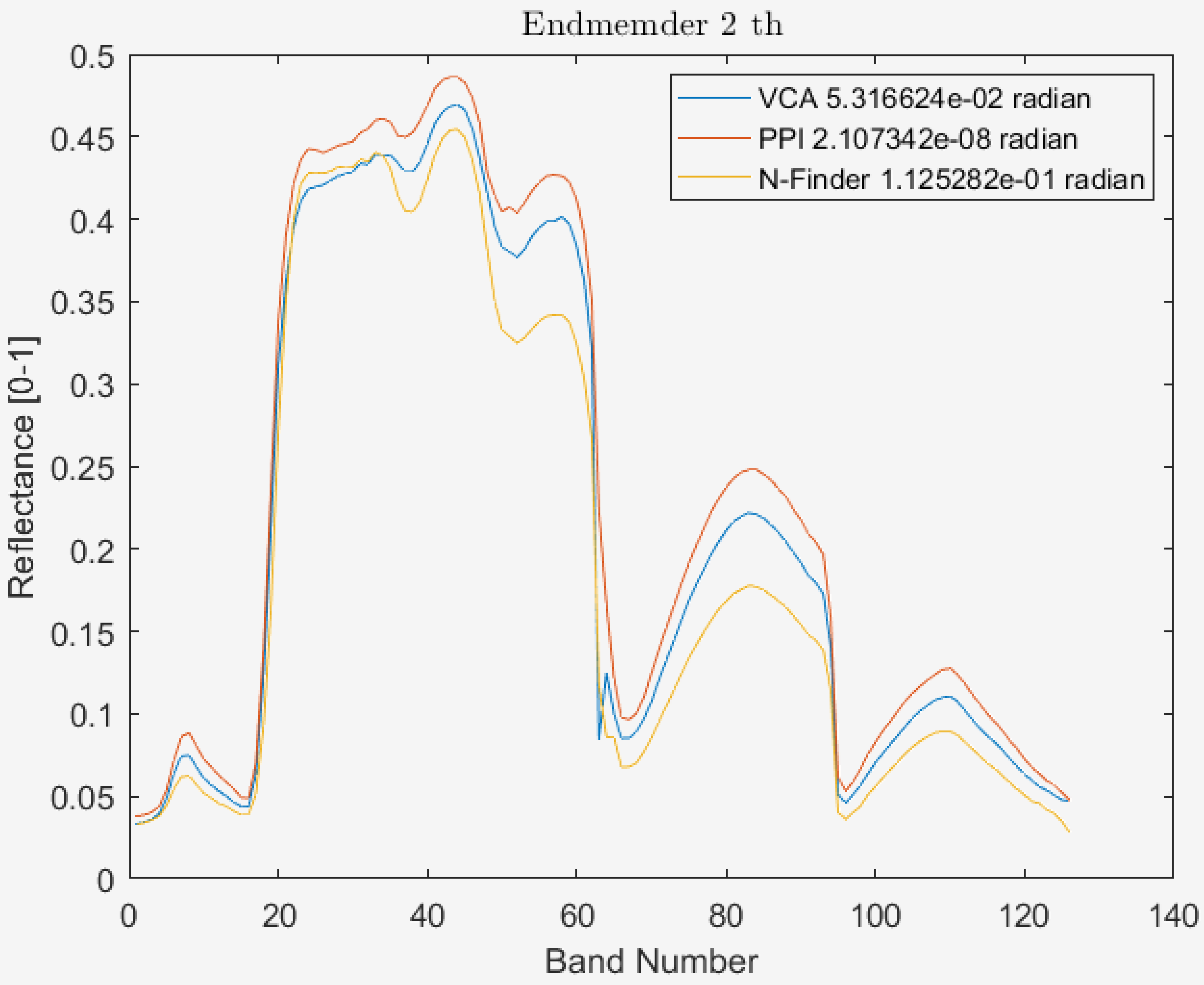
indicies - Indicies of pure pixels in U ...... **(1 * 19)**

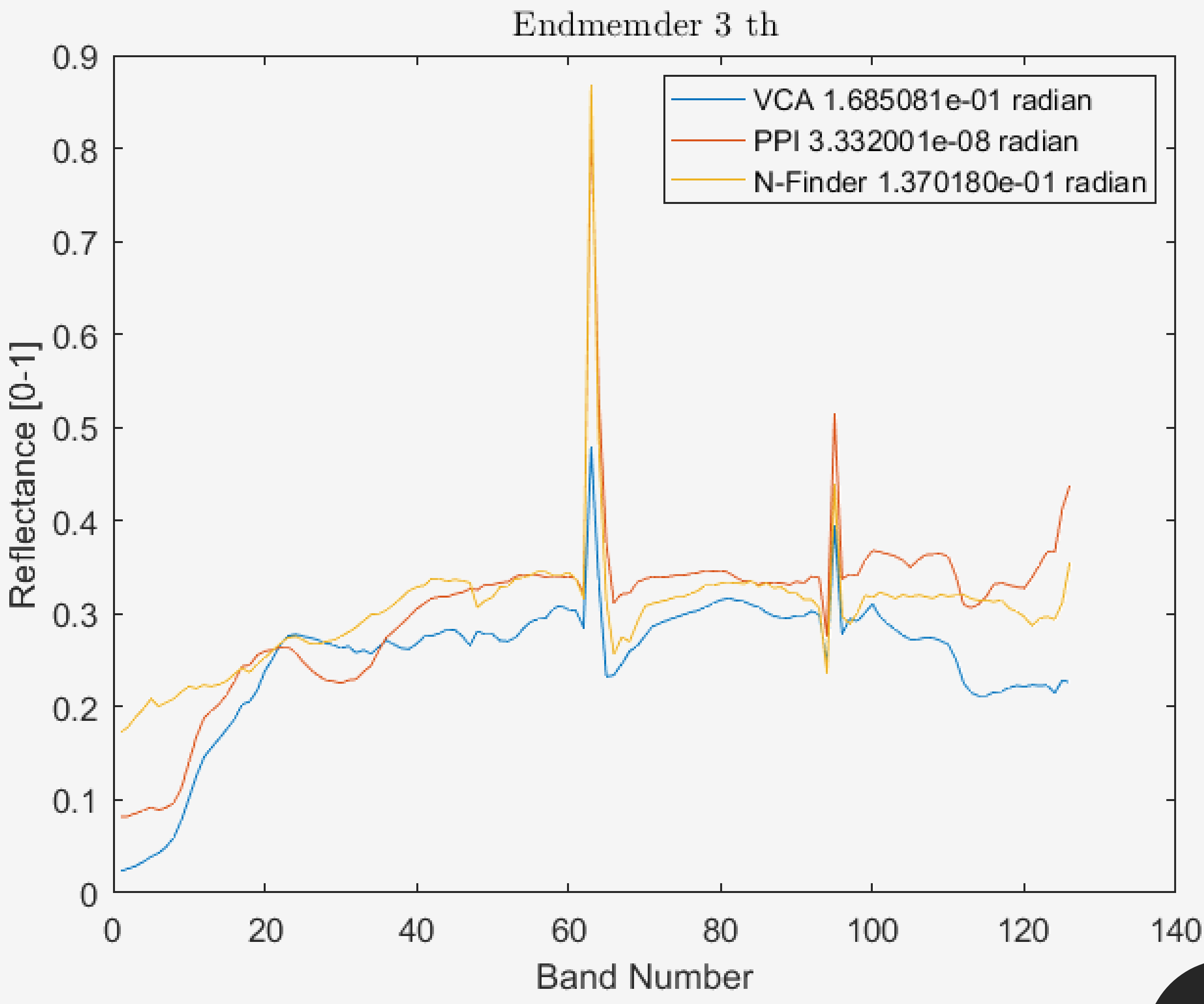
snrEstimate - SNR estimate of data [dB] **46.8222**

Endmember extraction

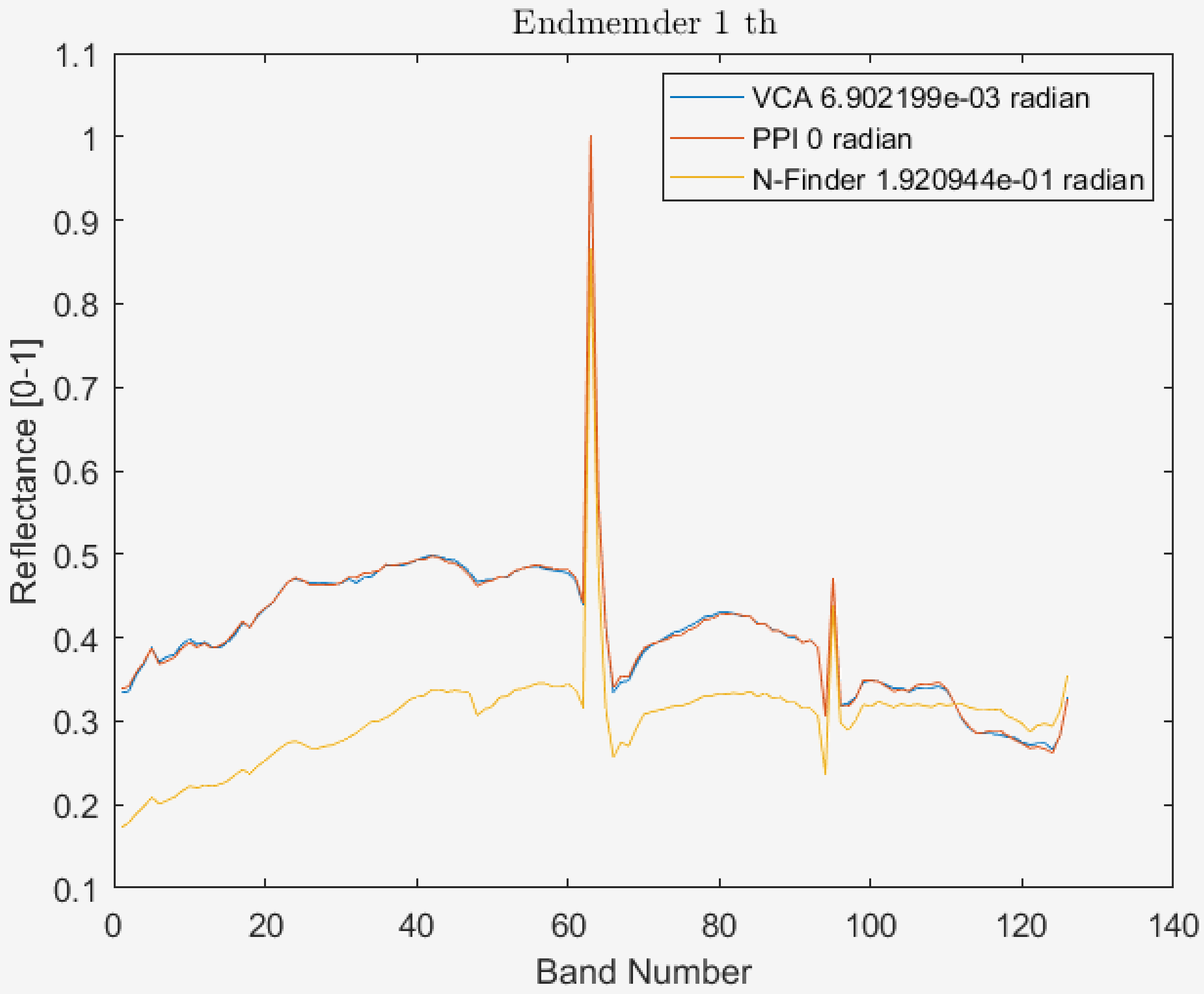


Comparison signature of endmembers using SAM for Cook City dataset



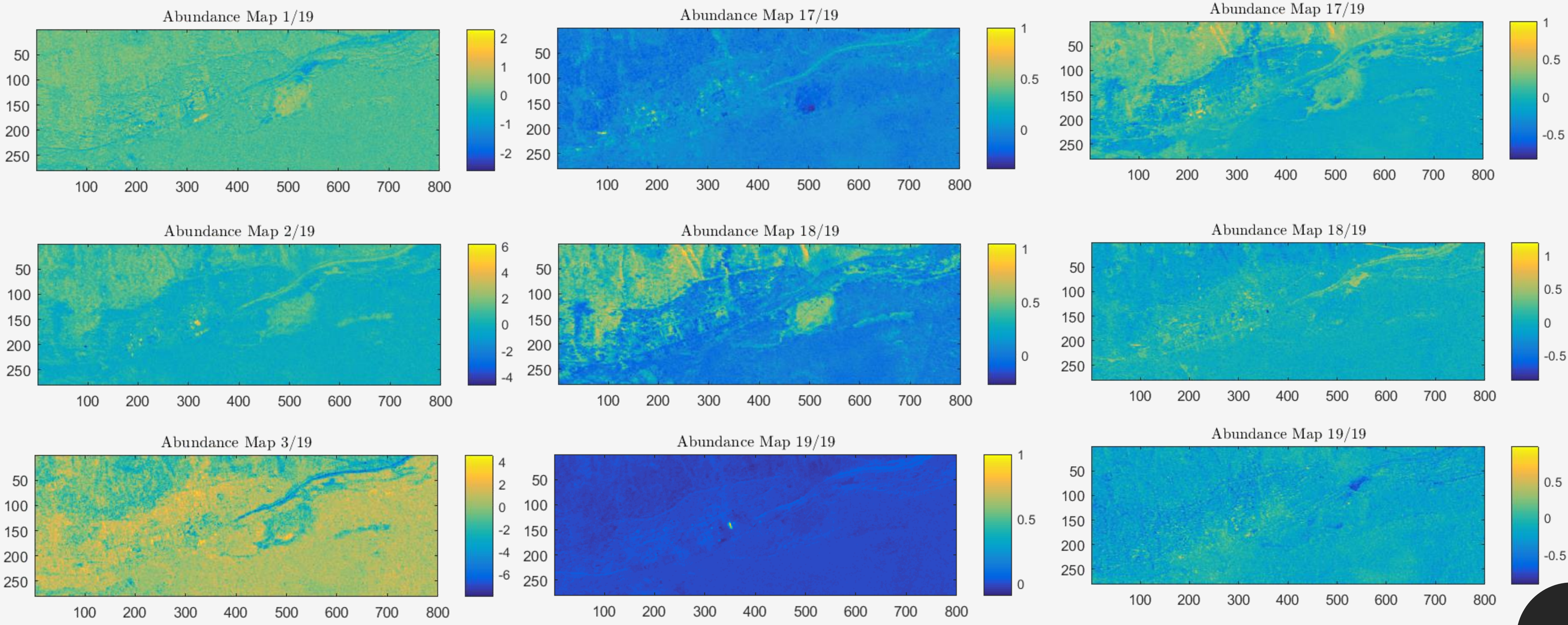


Comparison signature of endmembers using SAM for Cook City dataset



Abundance estimation

PPI NFINDER VCA



Target Detection

Structured methods for target detection

Constrained Energy Minimization

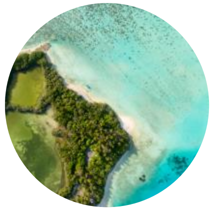
(CEM)

Adaptive Cosine Estimator

(ACE)

Signed Adaptive Cosine Estimator

(S-ACE)



Matched Filter

Generalized Likehood Ratio Test

(GLRT)

Target Detection

Unstructured methods for target detection

Adaptive Matched Subspace Detector

(AMSD)

Orthogonal Subspace Projection

(OSP)

Threshold Selection

1

Otsu

Otsu's method chooses a threshold that minimizes the intraclass variance of the thresholded black and white pixels.

CEM
Constrained Energy Minimization

1 of 5 Structured Method

- HYPERCEM Performs constrained energy minimization (CEM) algorithm
- Performs the constrained energy minimization algorithm for target detection.

Usage

[results] = hyperCem(M, target)

Inputs

M - 2d matrix of HSI data (p x N) → 126*224000

target - target of interest (p x 1) → 126*1

Outputs

results - vector of detector output (N x 1) → 224000*1

ACE

Adaptive Cosine Estimator

- HYPERACE Performs the adaptive cosin/coherent estimator algorithm
- Performs the adaptive cosin/coherent estimator algorithm for target detection.

Usage

[results] = hyperAce(M, S)

Inputs

M - 2d matrix of HSI data (p x N)) → 126*224000

S - 2d matrix of target endmembers (p x 1) → 126*1

Outputs

results - vector of detector output (N x 1) → 224000*1

GLRT

Generalized Likehood Ratio Test

3 of 5 Structured Method

- HYPERGLRT Performs the generalized liklihood test ratio algorithm
- Performs the generalized liklihood test ratio algorithm for target detection.

Usage

[results] = hyperGlrt(M, U, target)

Inputs

M - 2d matrix of HSI data (p x N) → 126*224000

t - target of interest (p x 1) → 126*1

Outputs

results - vector of detector output (N x 1) → 224000*1

S-ACE

Signed Adaptive Cosine Estimator (S-ACE)

4 of 5 Structured Method

- HYPERACE Performs the adaptive cosin/coherent estimator algorithm
- Performs the adaptive cosin/coherent estimator algorithm for target detection.

Usage

[results] = hyperSignedAce(M, S)

Inputs

M - 2d matrix of HSI data (p x N) → 126*224000

S - target of interest (p x 1) → 126*1

Outputs

results - vector of detector output (N x 1) → 224000*1

MF

Matched Filter

5 of 5 Structured Method

- HYPERACE Performs Matched Filter estimator algorithm
- Performs the estimator algorithm for target detection.

Usage

[results] = hyperMatchedFilter(M, S)

Inputs

M - 2d matrix of HSI data (p x N) → 126*224000

S - target of interest (p x 1) → 126*1

Outputs

results - vector of detector output (N x 1) → 224000*1

OSP
Orthogonal Subspace Projection

1 of 2 Unstructured Method

- HYPEROSP Performs the othogonal subspace projection (OSP) algorithm
- Performs the othogonal subspace projection algorithm for target detection.

Usage

[results] = hyperOsp(M, U, target)

Inputs

M - 2d matrix of HSI data (p x N)

U - 2d matrix of background endmebers (p x q)

target - target of interest (p x 1)

Outputs

results - vector of detector output (N x 1)

AMSD

Adaptive Matched Subspace Detector

2 of 2 Unstructured Method

- HYPERAMSD Adaptive matched subspace detector (AMSD) algorithm
- Performs the adaptive matched subspace detector (AMSD) algorithm for target detection

Usage

[results] = hyperAmsd(M, U, target)

Inputs

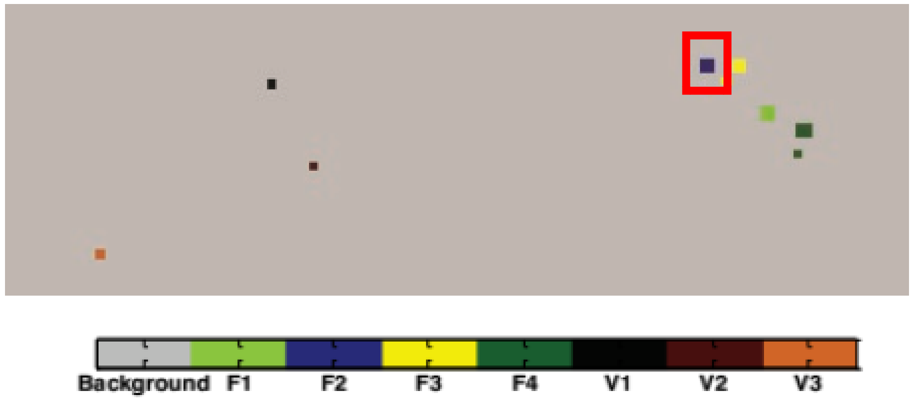
M - 2d matrix of HSI data (p x N)

B - 2d matrix of background endmebers (p x q)

target - target of interest (p x 1)

Outputs

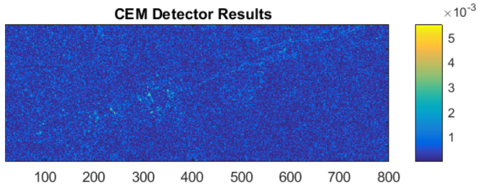
results - vector of detector output (N x 1)



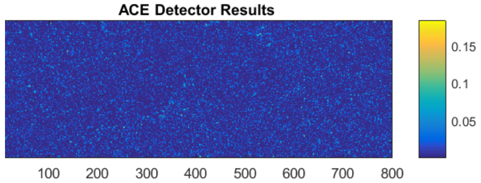
Fabric 2

Result of Structured Method

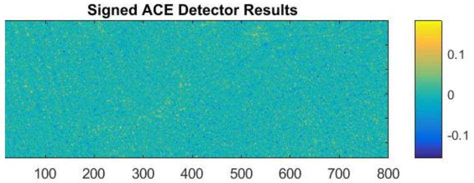
Detector Map



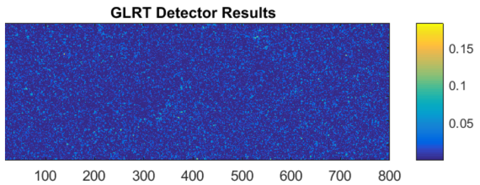
CEM



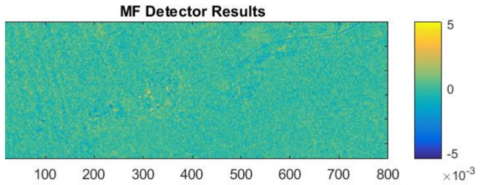
ACE



S-ACE



GLRT

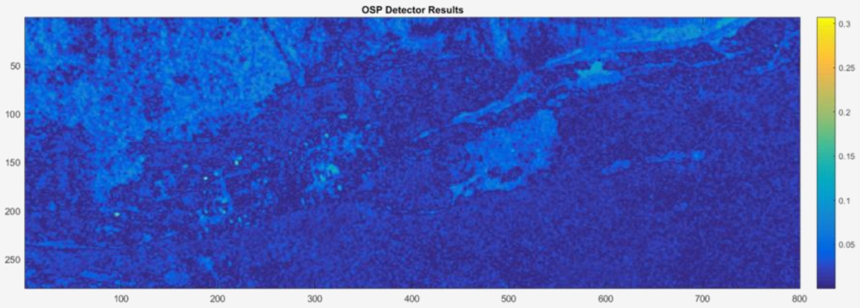


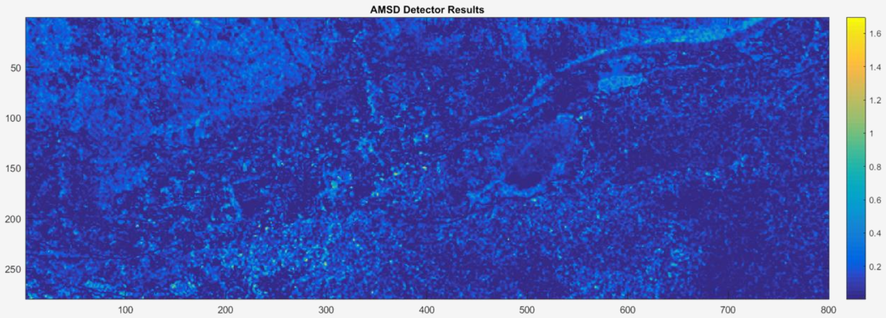
Matched Filter

OSP

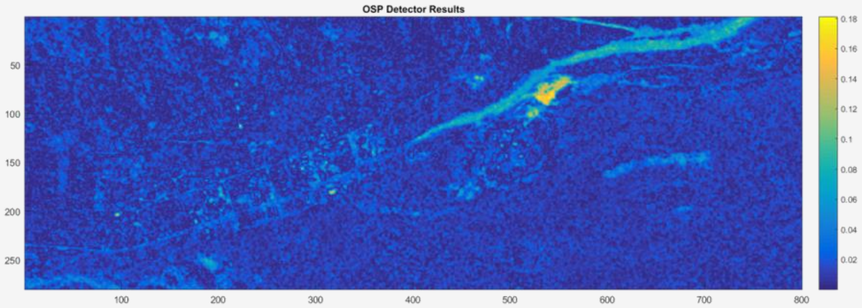
AMSD

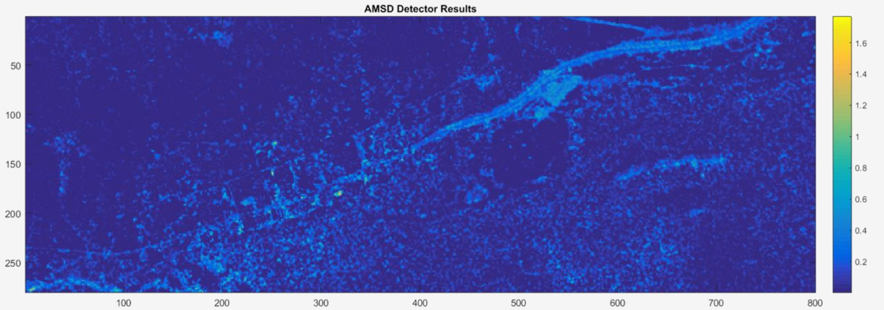
PPI



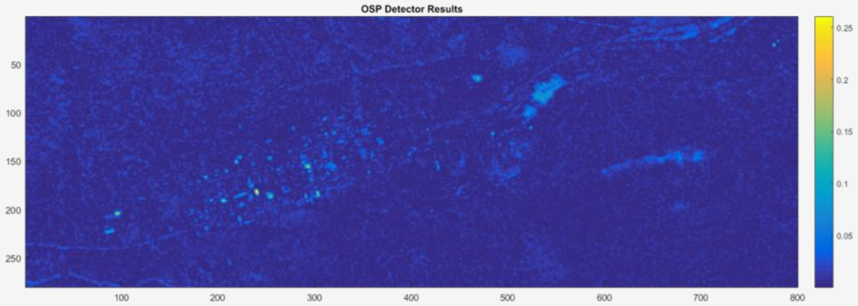


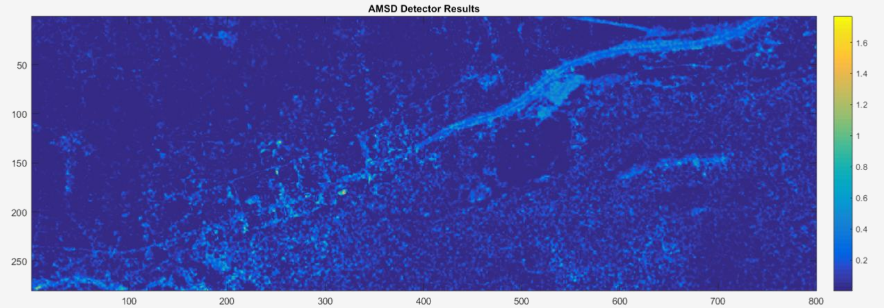
NFINDER



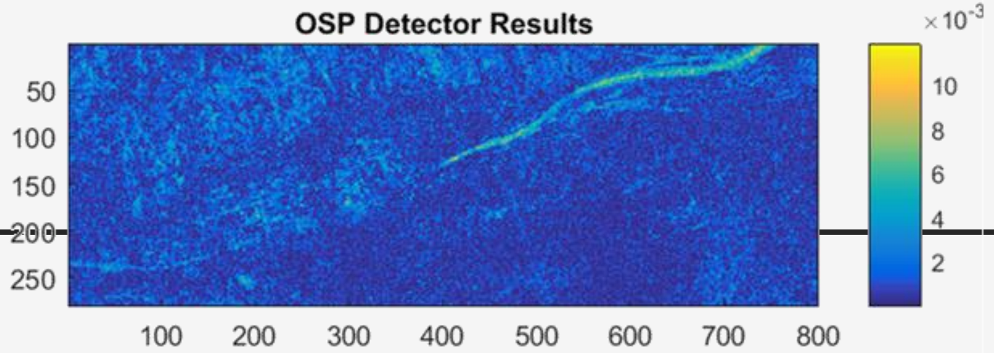


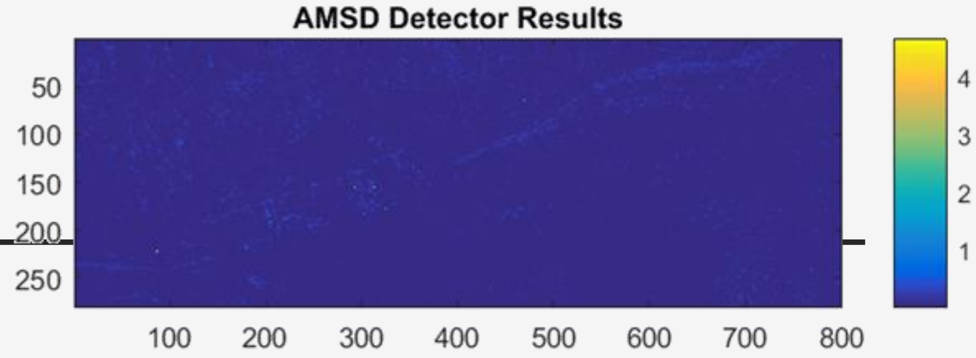
VCA



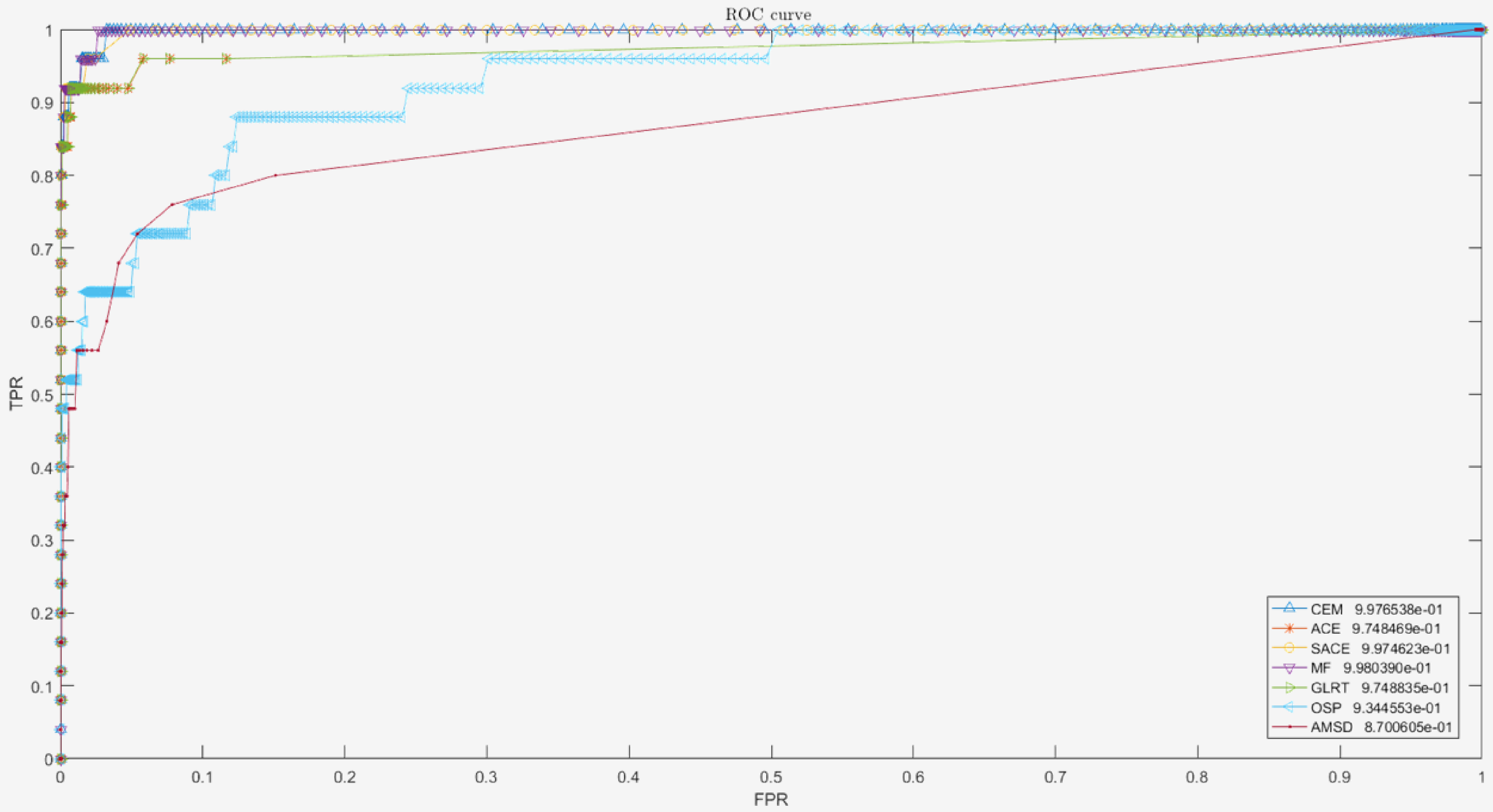


ATGP





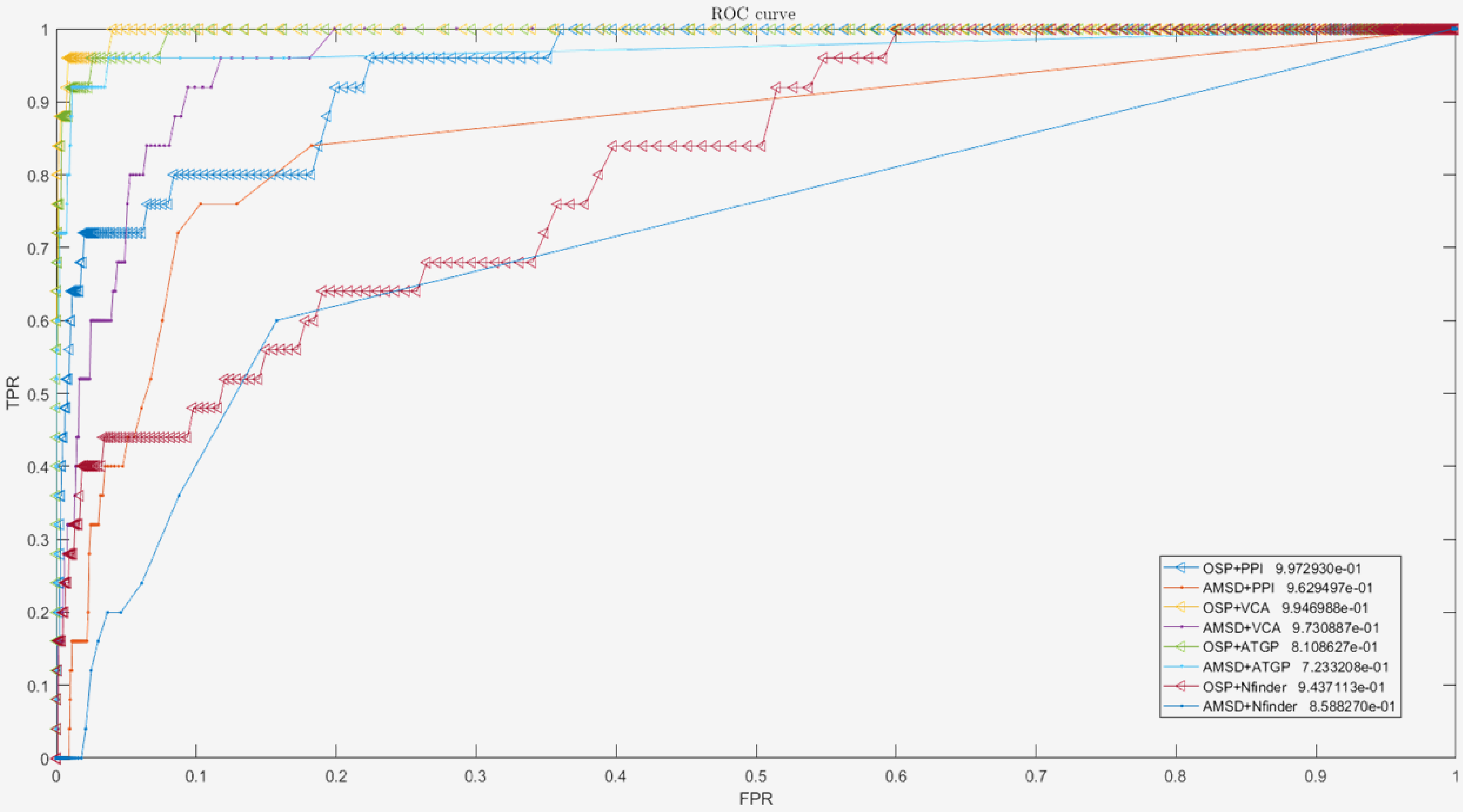
Fabric 2



Method	AUC
CEM	0.998
ACE	0.975
SACE	0.998
MF	**0.999**
GLRT	0.975
OSP	0.935
AMSD	0.871

- ROC curve of Fabric 2 data for 1000 different thresholds
- For Fabric 2 data structured methods have higher AUC than unstructured methods.

Fabric 2



Method	AUC
OSP+PPI	0.8279
AMSD+PPI	0.8271
OSP+VCA	**0.9946**
AMSD+VCA	**0.9730**
OSP+ATGP	0.8108
AMSD+ATGP	0.7233
OSP+Nfinder	0.9228
AMSD+Nfinder	0.7937

- ROC curve of Fabric 2 data for 1000 different thresholds
- The AUC of each curve calculated

Fabric 2

Method	AUC	ACC	PPV*10^-3
OSP+PPI	0.8279	0.5299	0.1163
AMSD+PPI	0.8271	0.9931	0.1116
OSP+VCA	**0.9946**	0.6638	24.09
AMSD+VCA	**0.9730**	**0.9974**	**100**
OSP+ATGP	0.8108	0.6227	0.4055
AMSD+ATGP	0.7233	0.2427	0.1116
OSP+Nfinder	0.9228	0.6292	0.4630
AMSD+Nfinder	0.7937	0.9965	0.1116

Method	AUC	ACC	PPV
CEM	0.998	0.6806	1
ACE	0.975	0.9982	1
SACE	0.998	0.8197	1
MF	0.999	0.6821	1
GLRT	0.975	0.9982	1
OSP	0.935	0.6227	0.4055
AMSD	0.871	0.2427	0.1116

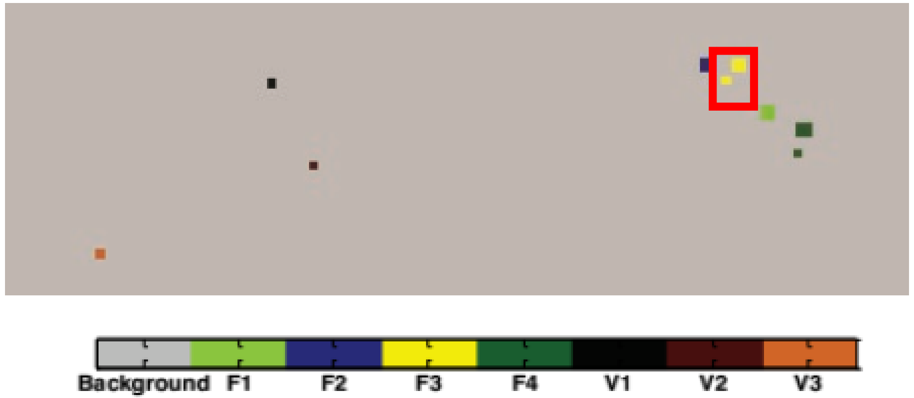
- For FABRIC3 data, it is observed that the AUC parameter for the OSP method was the highest when the VCA approach was used. In general, the area under the ROC curve for the OSP method is higher, However the overall accuracy of the AMSD method is higher, and therefore the third parameter, PPV, represents the optimal method, the OSP + VCA method.

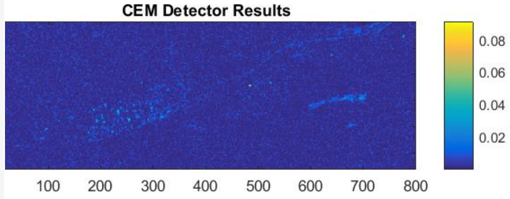
- In second table,In Unstructured method, first of all estimate number of endmembers with HFC ,that is 33 , then calculate matrix of background endmembers with ATGP and finally use AMSD and OSP to target detection.

Fabric 3

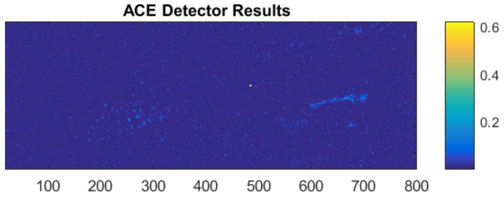
Result of Structured Method

Detector Map

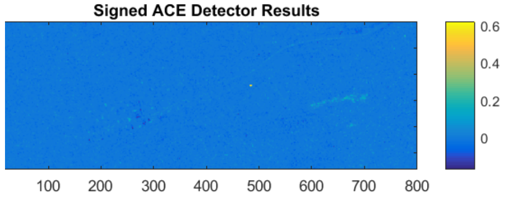




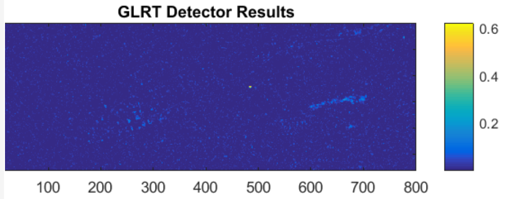
CEM



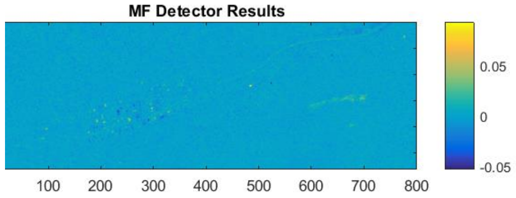
ACE



S-ACE



GLRT



Matched Filter

OSP

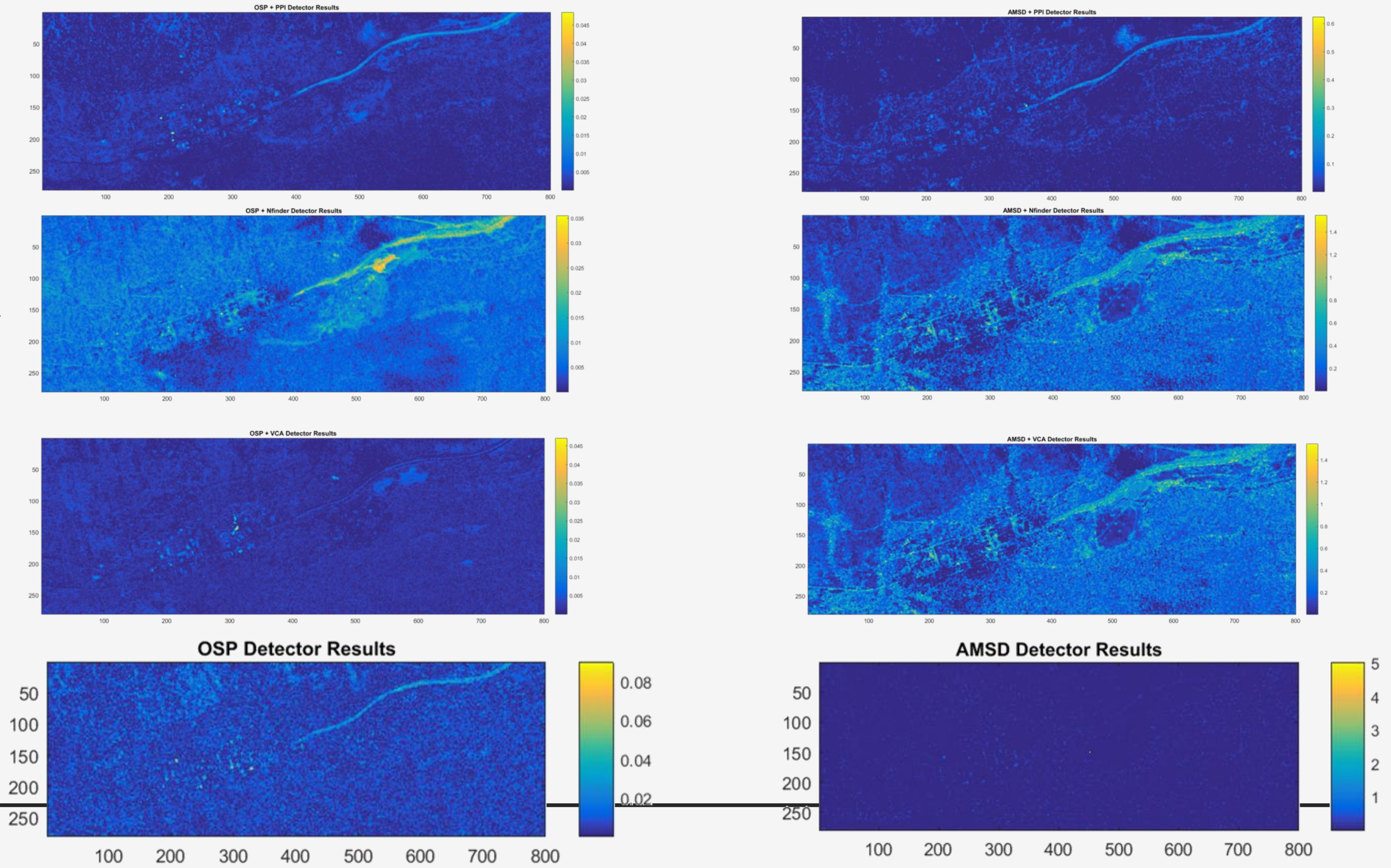
AMSD

PPI

NFINDER

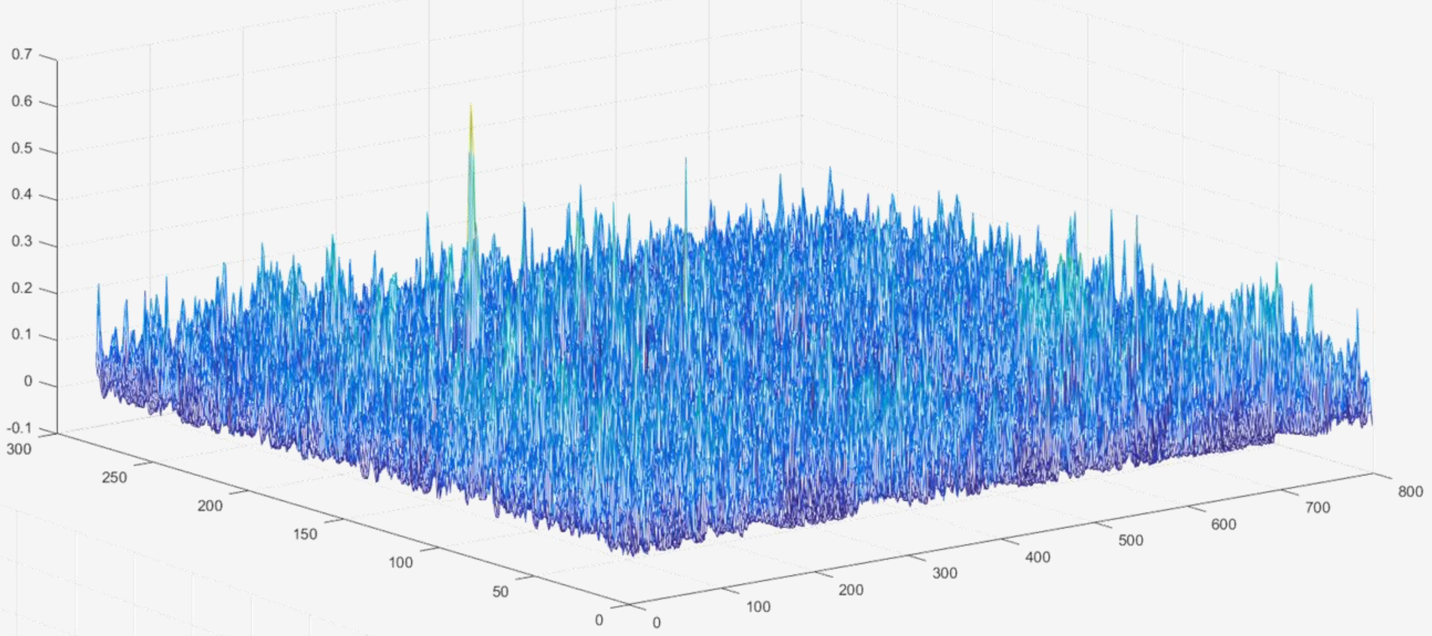
VCA

ATGP



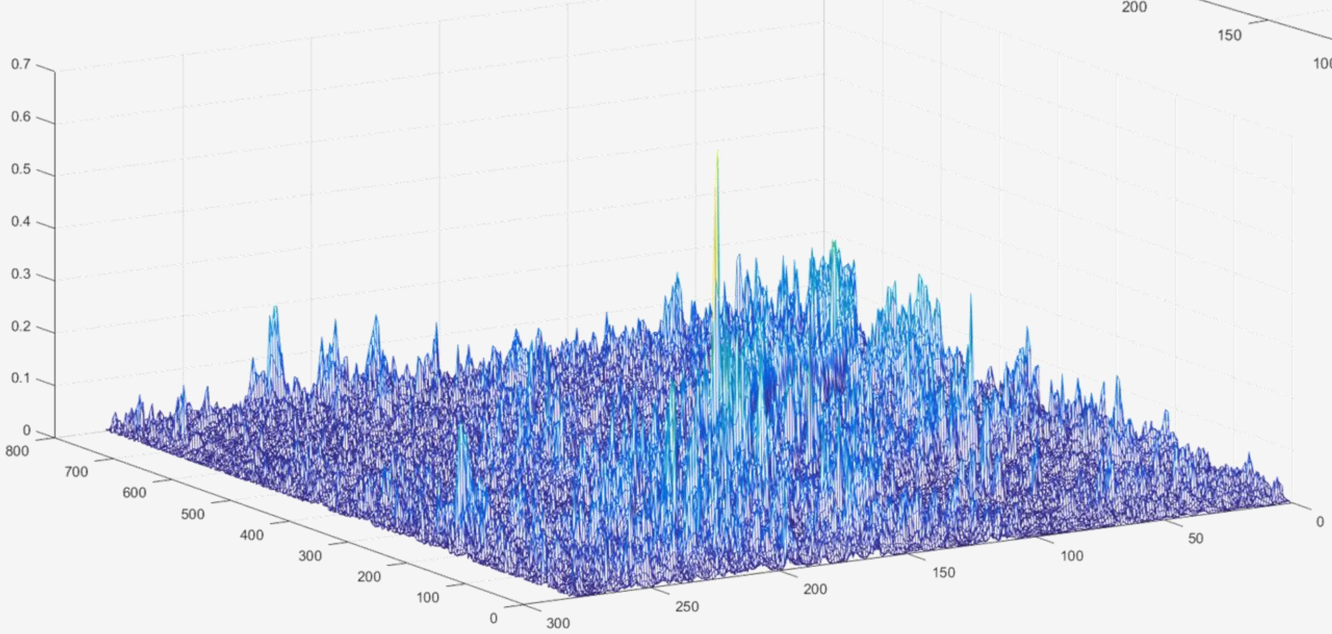
AMSD + VCA

AMSD Detector Results



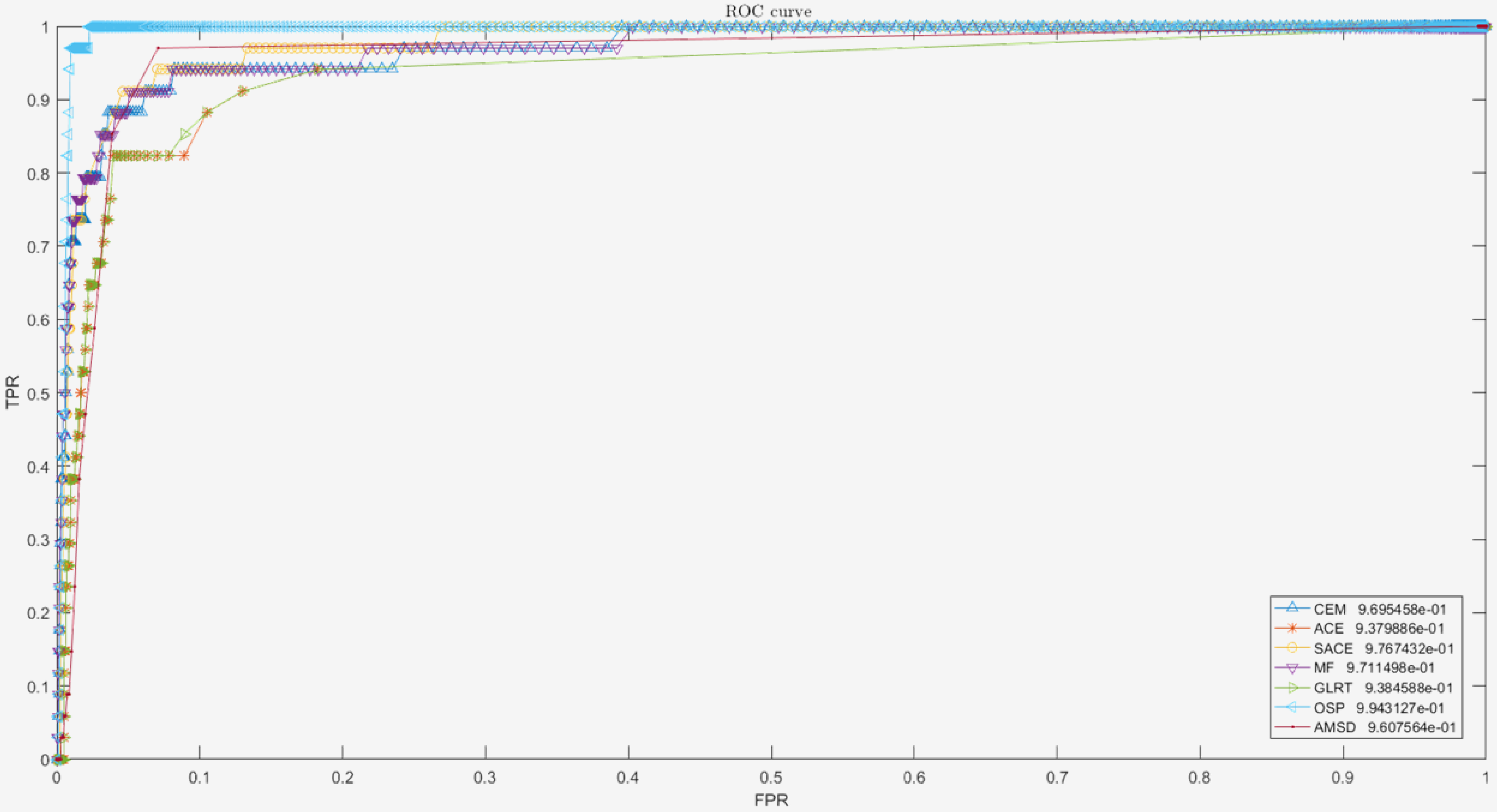
AMSD + PPI

AMSD Detector Results



- Result of Unstructured Method
- Detector Map

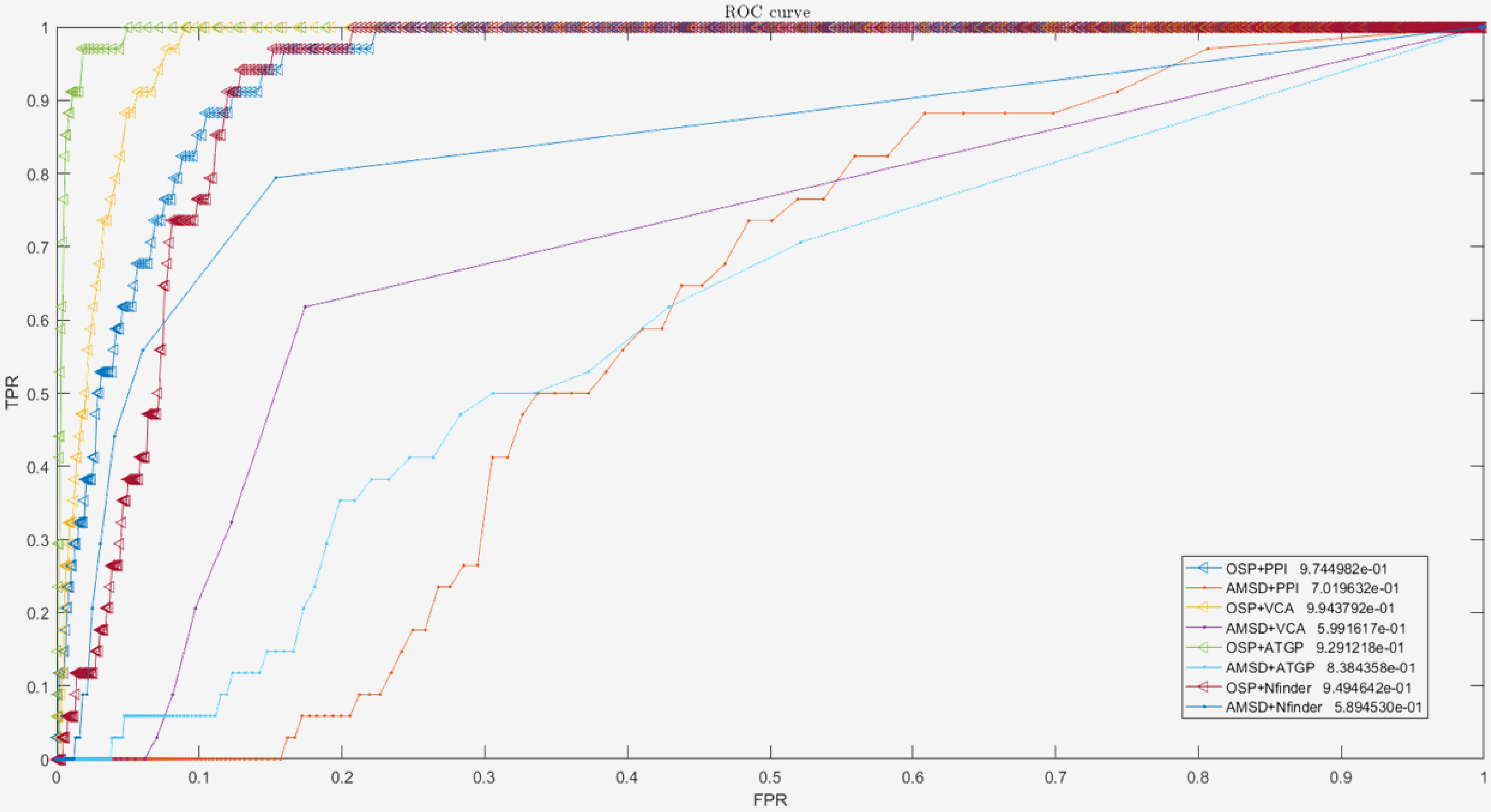
Fabric 3



Method	AUC
CEM	0.970
ACE	0.938
SACE	0.977
MF	0.972
GLRT	0.939
OSP	0.995
AMSD	0.961

- ROC curve of Fabric 3 data for 1000 different thresholds
- The AUC of each curve calculated

Fabric 3



Method	AUC
OSP+PPI	0.9744
AMSD+PPI	0.7019
OSP+VCA	**0.9943**
AMSD+VCA	0.5991
OSP+ATGP	0.9291
AMSD+ATGP	0.8384
OSP+Nfinder	0.9494
AMSD+Nfinder	0.5894

- ROC curve of Fabric 3 data for 1000 different thresholds
- AUC for OSP is higher than AMSD method

Fabric 3

Method	AUC	ACC	PPV*10^-3
OSP+PPI	<u>0.9744</u>	0.7278	0.2217
AMSD+PPI	0.7019	<u>0.9973</u>	0.1517
OSP+VCA	**0.9943**	0.7307	**0.9185**
AMSD+VCA	0.5991	<u>0.9890</u>	0.1517
OSP+ATGP	<u>0.9291</u>	0.5822	0.1716
AMSD+ATGP	0.8384	0.2211	0.1517
OSP+Nfinder	<u>0.9494</u>	0.6369	0.2783
AMSD+Nfinder	0.5894	<u>0.9738</u>	0.1517

Method	AUC	ACC	PPV*10^-3
CEM	0.970	0.5677	0. 5795
ACE	0.938	0.9964	0.2292
SACE	0.977	0.6325	0.4180
MF	0.972	0.5763	0. 6742
GLRT	0.939	0.9964	0.2298
OSP	**0.995**	0.5822	0.1716
AMSD	0.961	0.2211	0.1517

- For FABRIC3 data, it is observed that the AUC parameter for the OSP method was the highest when the VCA approach was used. In general, the area under the ROC curve for the OSP method is higher, However the overall accuracy of the AMSD method is higher, and therefore the third parameter, PPV, represents the optimal method, the OSP + VCA method.and as you on detector maps OSP+Nfinder has good results.

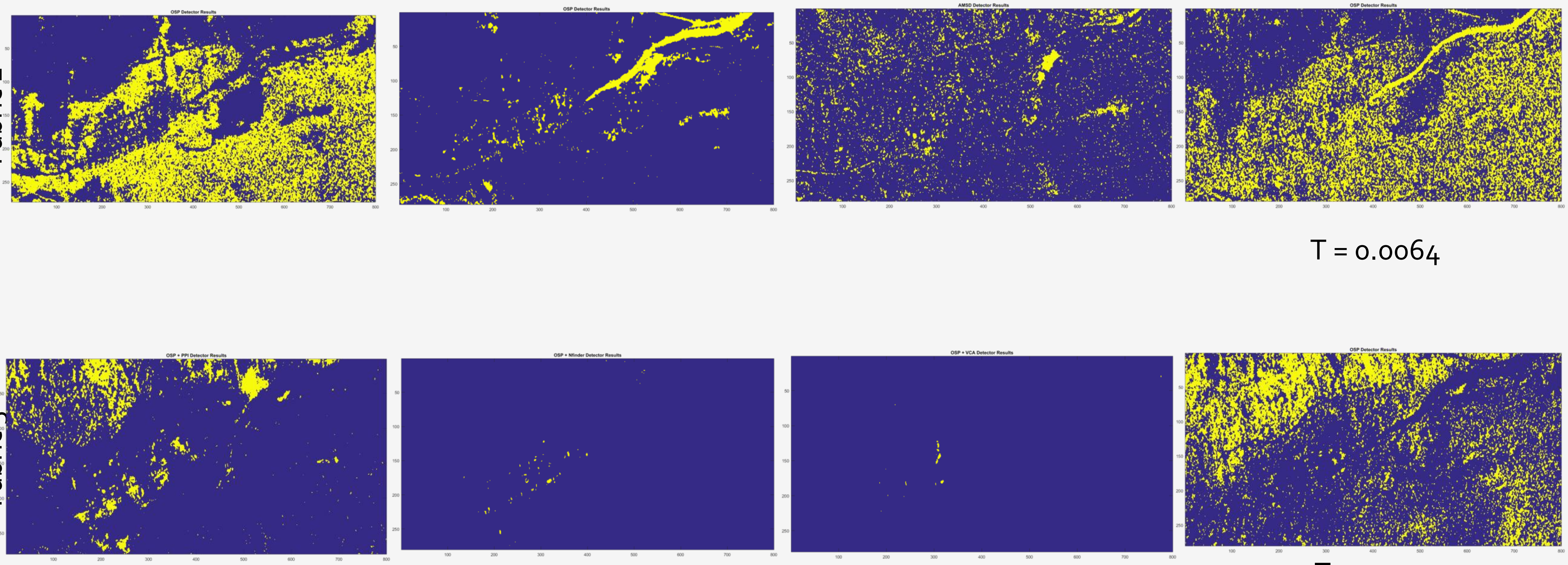
Result of OTSU Threshold Selection

Method	Fabric 2	Fabric 3	Vehicle 1
CEM	0.0028	0.0035	0.0032
ACE	0.0182	0.0183	0.0150
S-ACE	0.0166	0.0155	0.0122
MF	0.0027	0.0035	0.0032
GLRT	0.0180	0.0181	0.0149
OSP	0.0064	0.0011	0.0068
AMSD	0.0783	0.0648	0.0903
OSP+PPI	0.0103	0.0019	0.0078
AMSD+PPI	0.1611	0.0404	0.0565
OSP+Nfinder	0.0329	0.0029	0.0260
AMSD+Nfinder	0.1561	0.2872	0.1648
OSP+VCA	0.0086	0.0085	0.0312
AMSD+VCA	0.1258	0.0720	0.1628

PPI NFINDER VCA ATGP

Fabric 2

Fabric 3



T = 0.0064

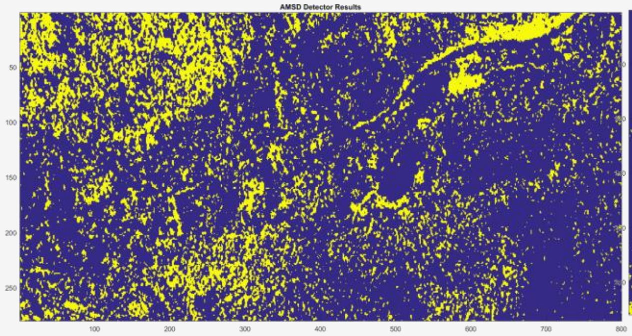
T = 0.0011

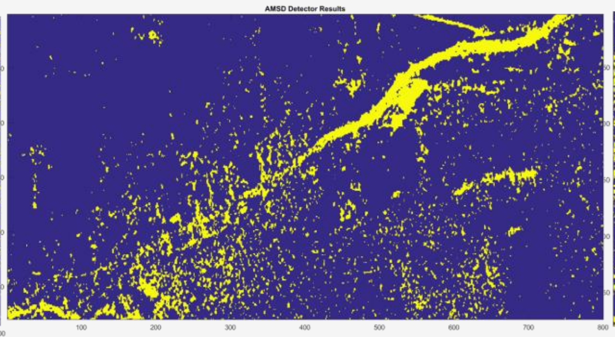
Threshold Selection Result on OSP Method

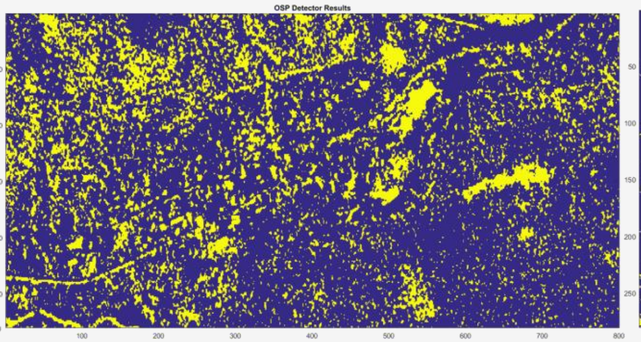
- This algorithm seems to work optimistically or because the amount of target data is small, this algorithm has low accuracy

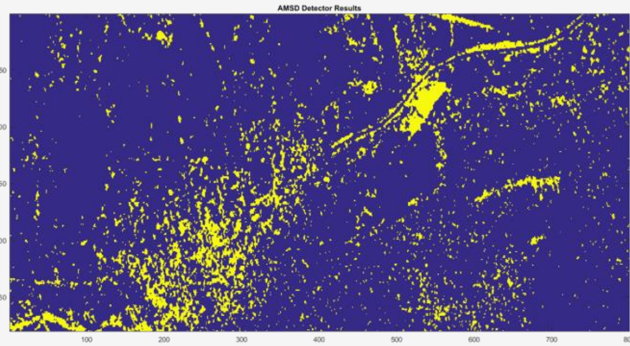
PPI NFINDER VCA ATGP

Fabric 2



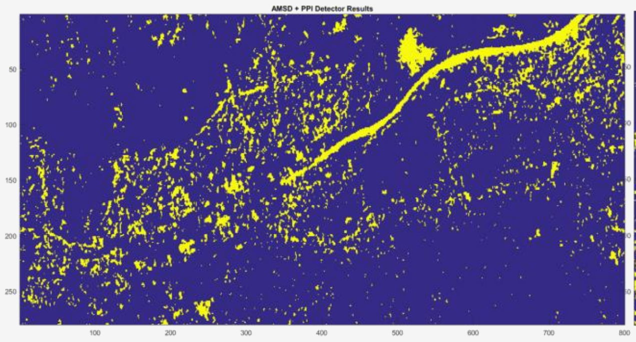


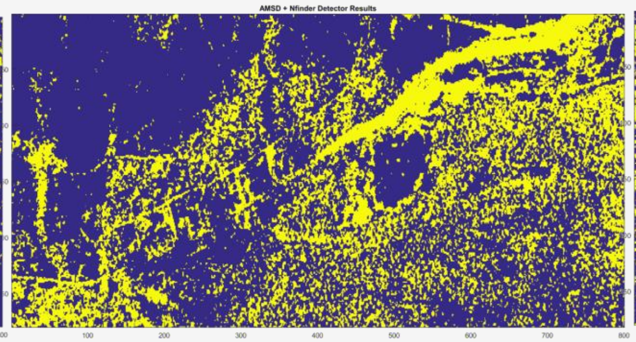


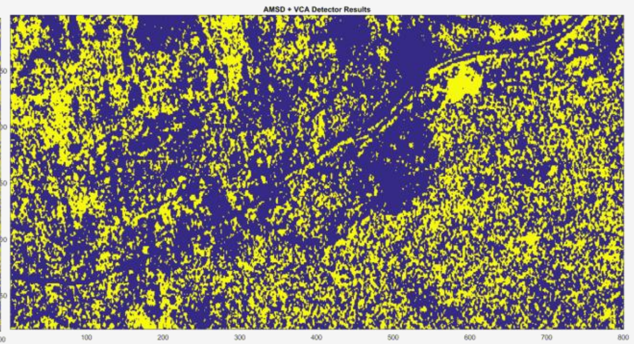


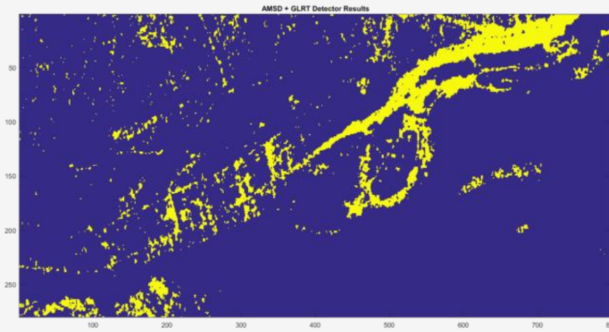
T = 0.0064

Fabric 3





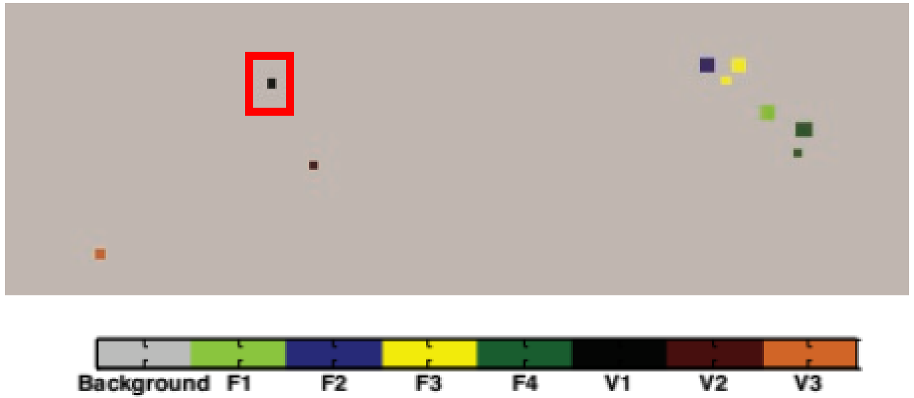




T = 0.0011

Threshold Selection Result on AMSD Method

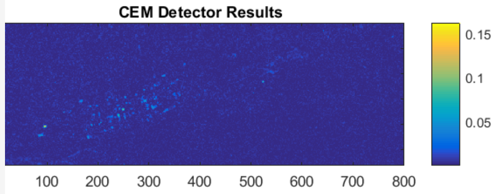
- This algorithm seems to work optimistically or because the amount of target data is small, this algorithm has low accuracy



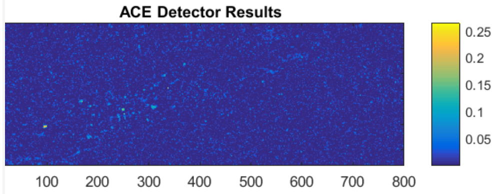
Vehicle 1

Result of Structured Method

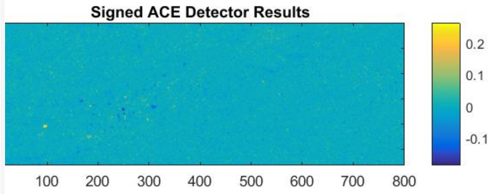
Detector Map



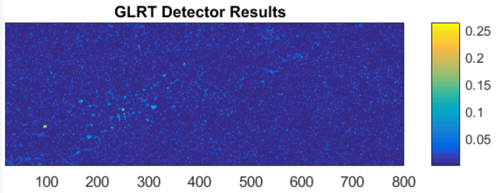
CEM



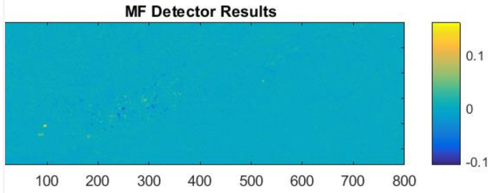
ACE



S-ACE



GLRT

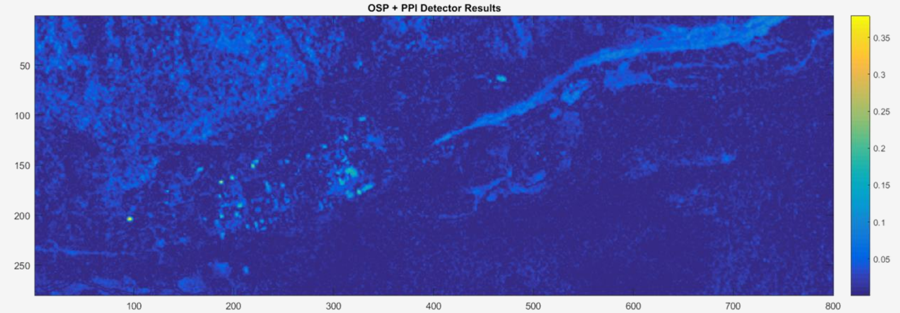


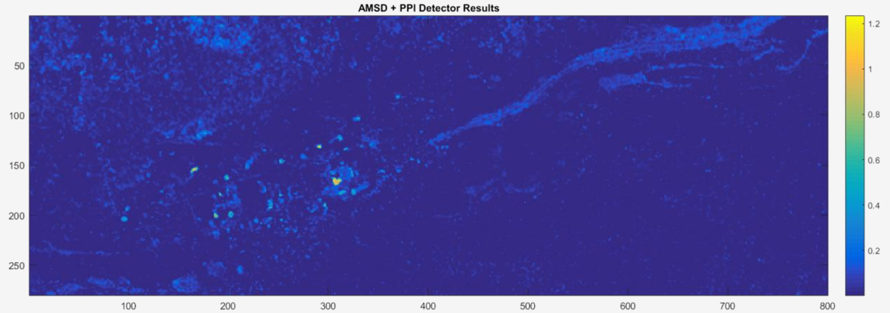
Matched Filter

OSP

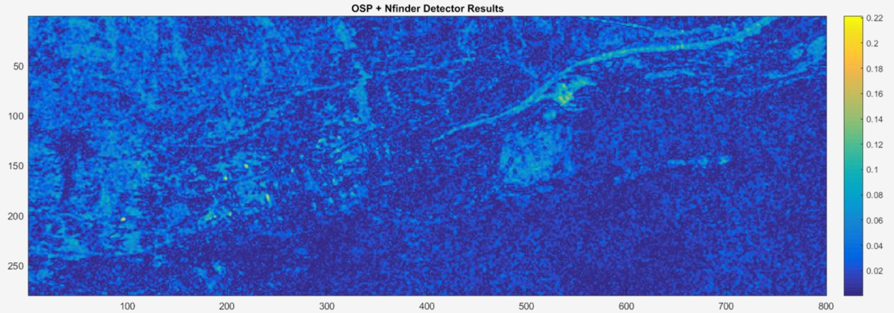
AMSD

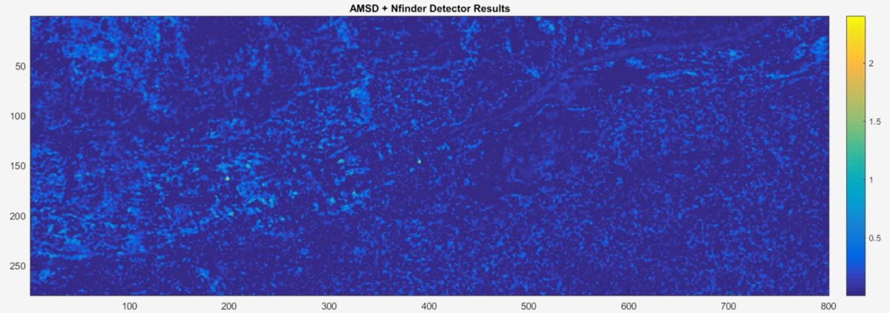
PPI



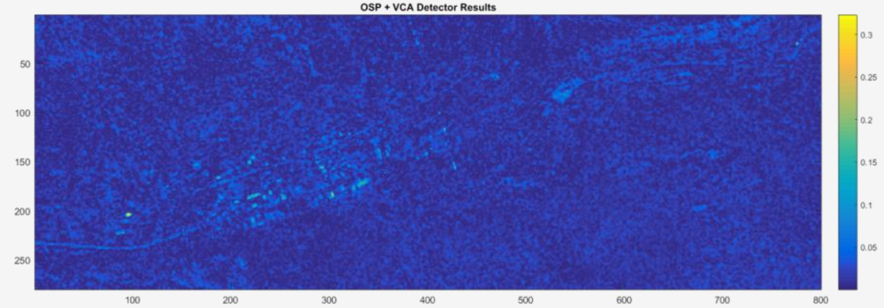


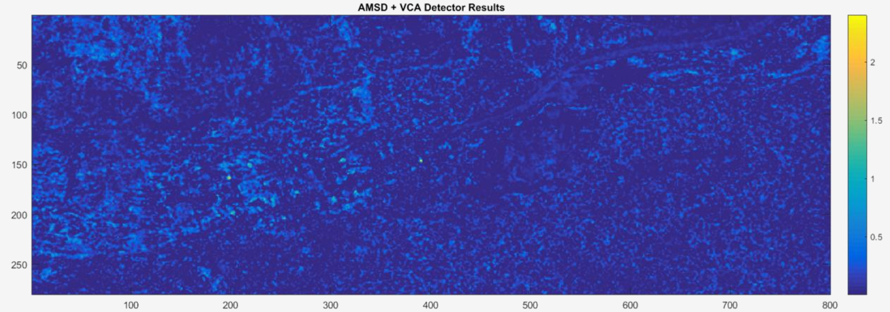
NFINDER



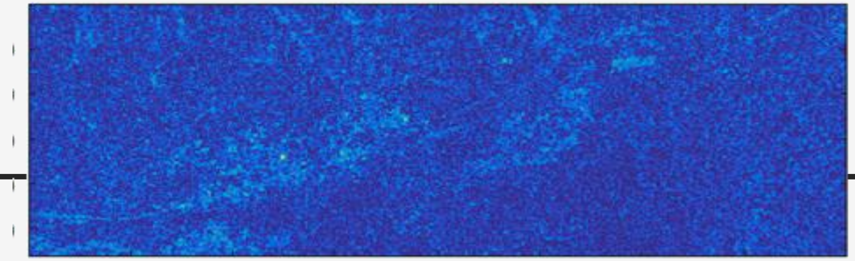


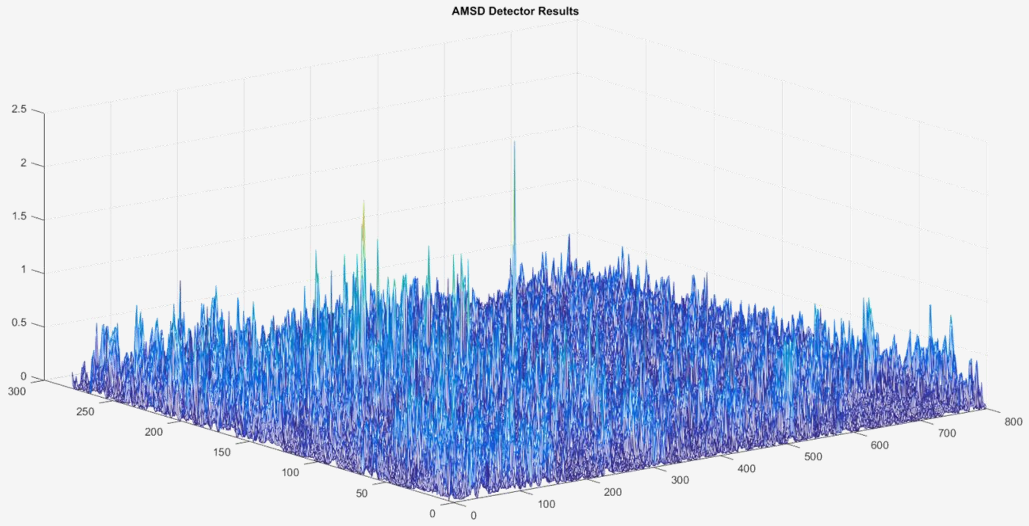
VCA



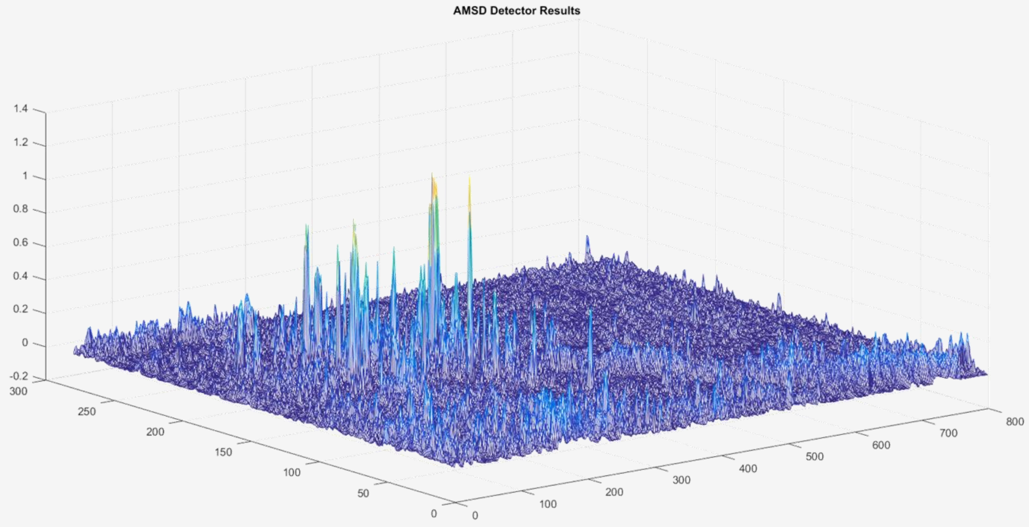


ATGP



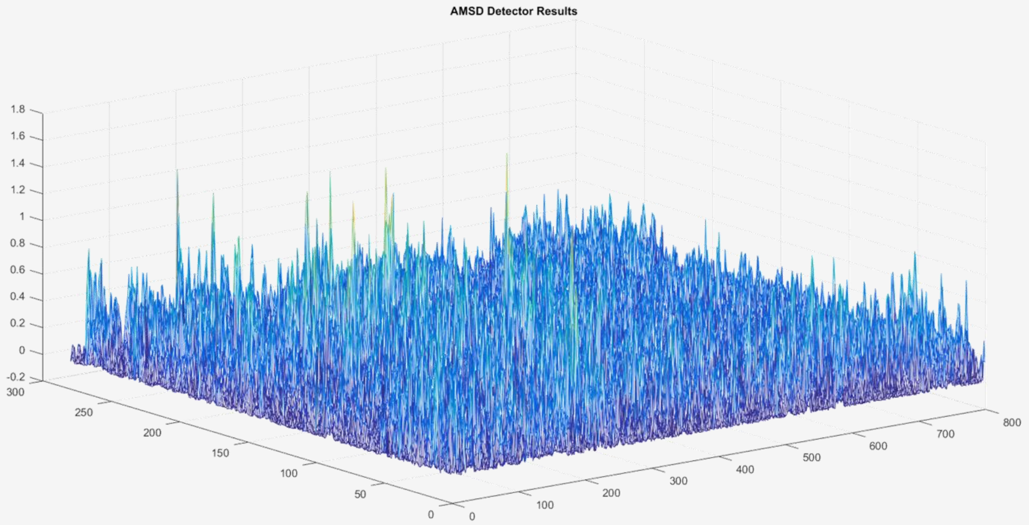


AMSD + Nfinder



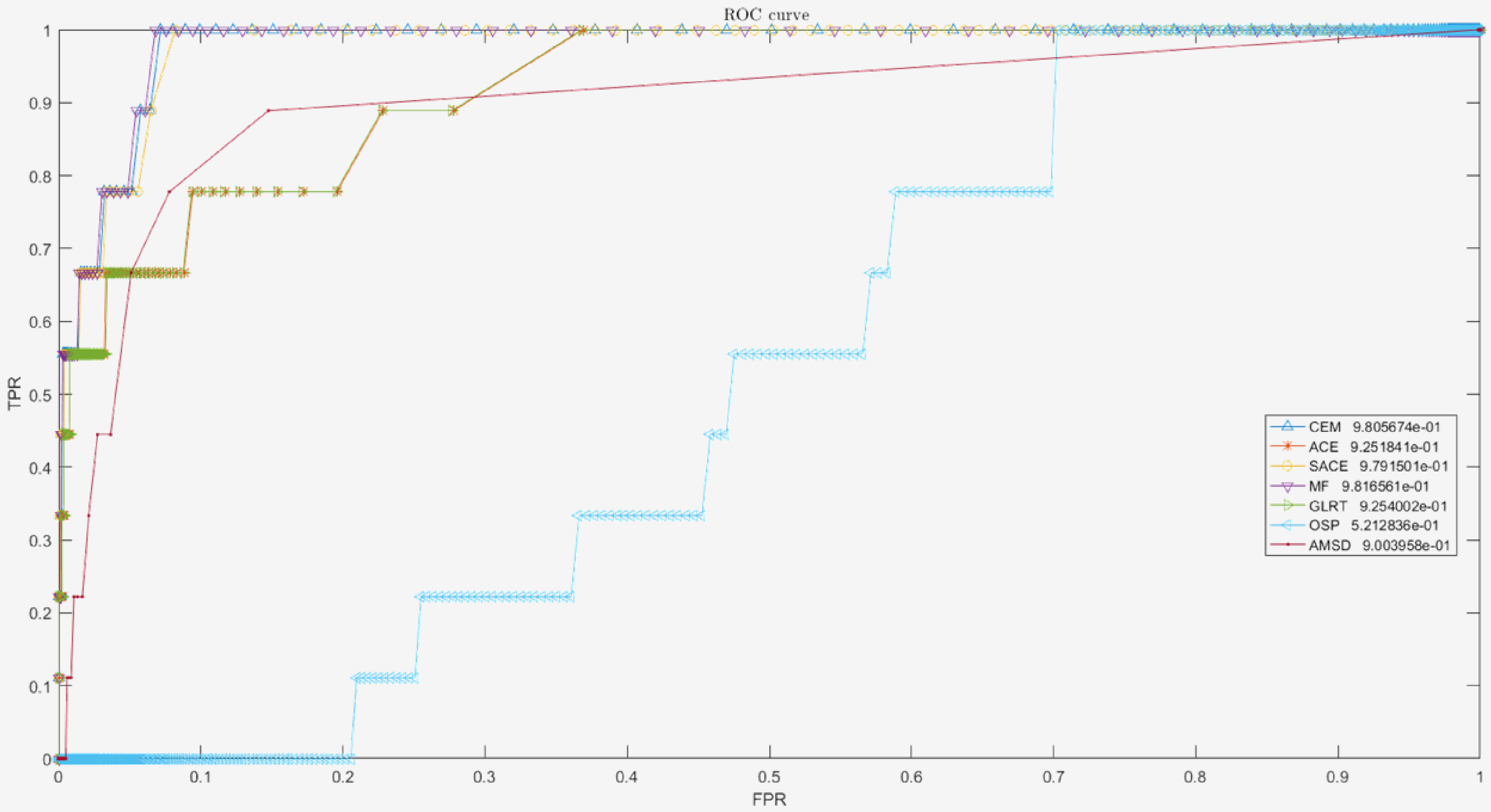
AMSD + PPI

- Result of Unstructured Method
- Detector Map



AMSD + VCA

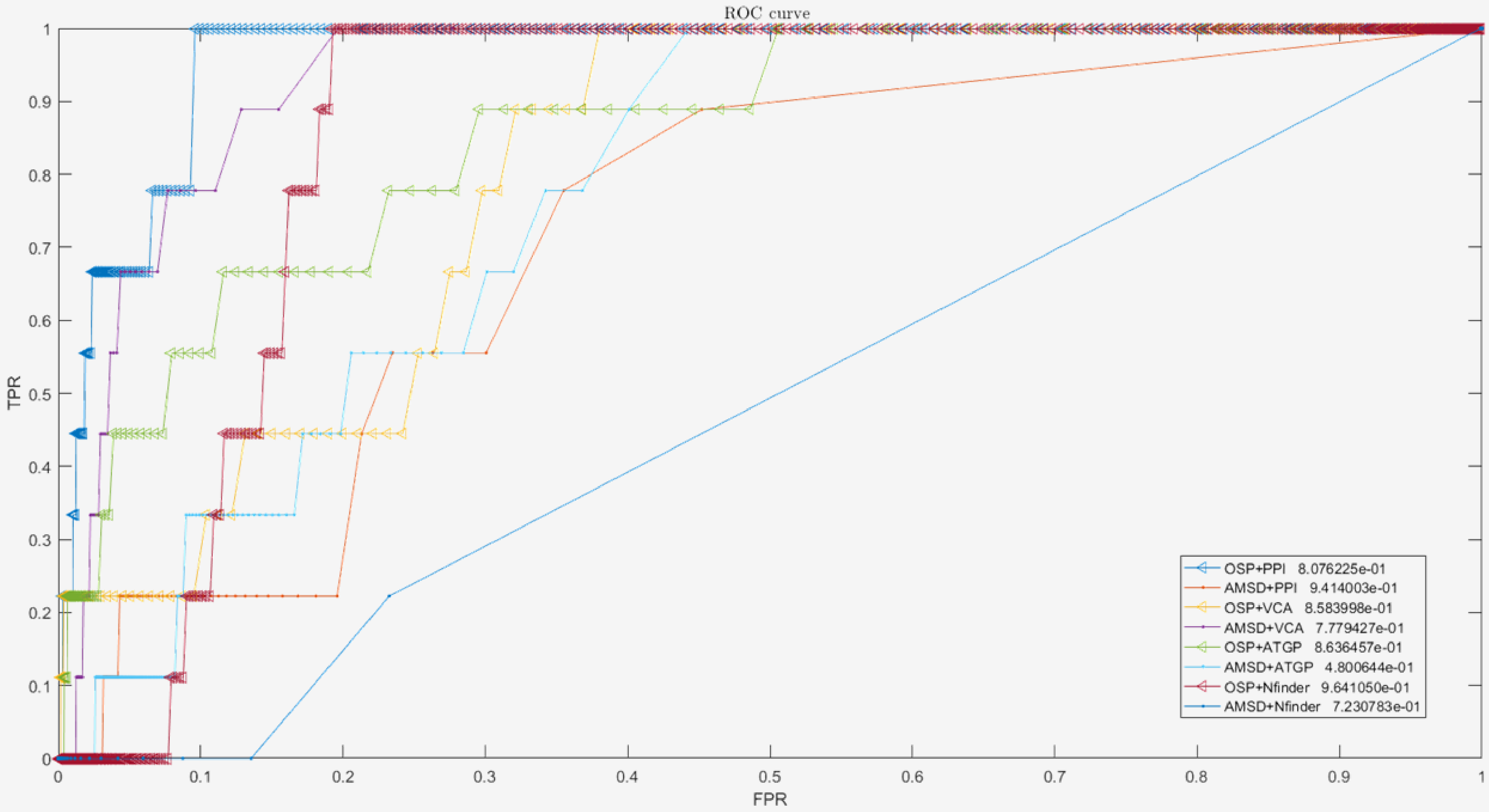
Vehicle 1



Method	AUC
CEM	0.981
ACE	0.926
SACE	0.980
MF	0.982
GLRT	0.926
OSP	0.522
AMSD	0.901

- ROC curve of Vehicle 1 data for 1000 different thresholds
- The AUC of each curve calculated

Vehicle 1



- ROC curve of Vehicle 1 data for 1000 different thresholds
- The AUC of each curve calculated

Vehicle 1

Method	AUC	ACC	PPV*10^-3
OSP+PPI	0.8076	0.5635	0. 3261
AMSD+PPI	0.9414	**0.9964**	0.0919
OSP+VCA	0.8583	0.6557	0. 160
AMSD+VCA	0.7779	0.9856	0.0458
OSP+ATGP	0.8636	0.6558	0.0589
AMSD+ATGP	0.4800	0.8323	0.0401
OSP+Nfinder	**0.9641**	0.5750	**0.3695**
AMSD+Nfinder	0.7230	0.9921	0.0405

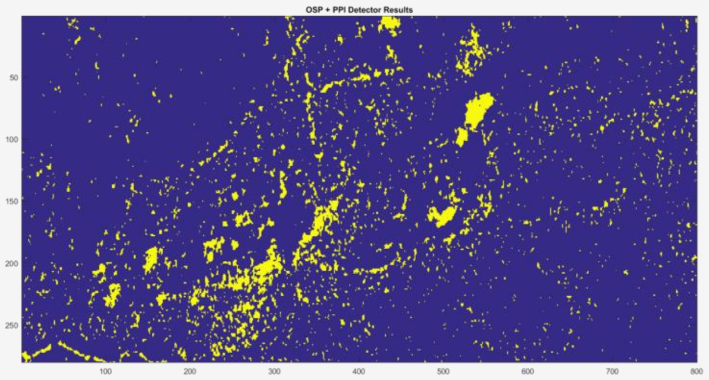
Method	AUC	ACC	PPV*10^-3
CEM	**0.981**	0.6205	5.63
ACE	0.926	0.9945	31.77
SACE	0.980	0.6210	55.55
MF	0.982	0.6211	5.84
GLRT	0.926	0.9946	31.74
OSP	0.522	0.6558	0.0589
AMSD	0.901	0.8323	0.0401

Result of OTSU Threshold Selection

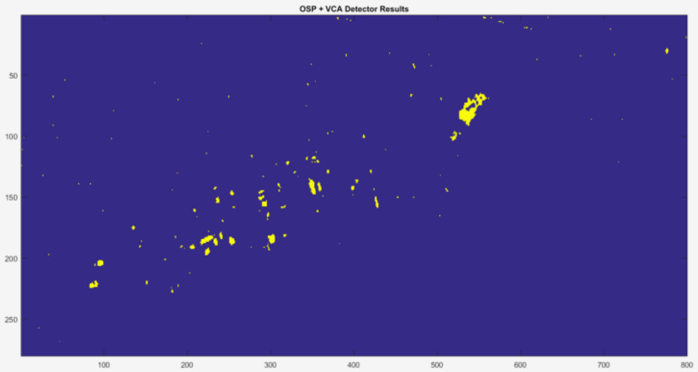
Method	Fabric 2	Fabric 3	Vehicle 1
CEM	0.0028	0.0035	0.0032
ACE	0.0182	0.0183	0.0150
S-ACE	0.0166	0.0155	0.0122
MF	0.0027	0.0035	0.0032
GLRT	0.0180	0.0181	0.0149
OSP	0.0064	0.0011	0.0068
AMSD	0.0783	0.0648	0.0903
OSP+PPI	0.0103	0.0019	0.0078
AMSD+PPI	0.1611	0.0404	0.0565
OSP+Nfinder	0.0329	0.0029	0.0260
AMSD+Nfinder	0.1561	0.2872	0.1648
OSP+VCA	0.0086	0.0085	0.0312
AMSD+VCA	0.1258	0.0720	0.1628

Vehicle 1

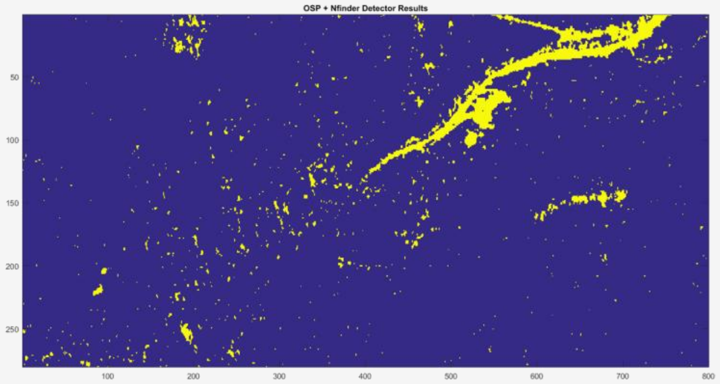
PPI



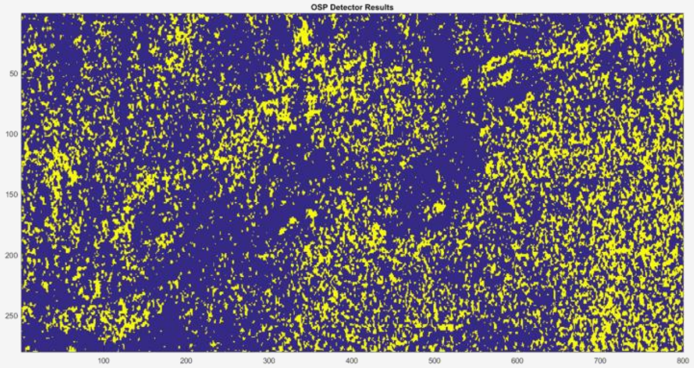
VCA



NFINDER



ATGP

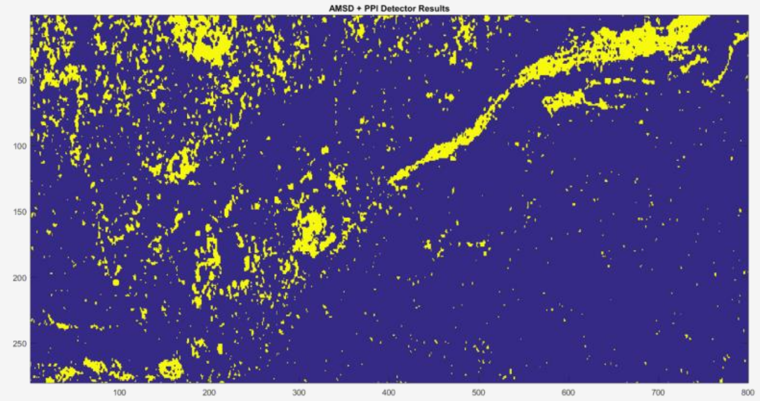


Threshold Selection Result on OSP Method

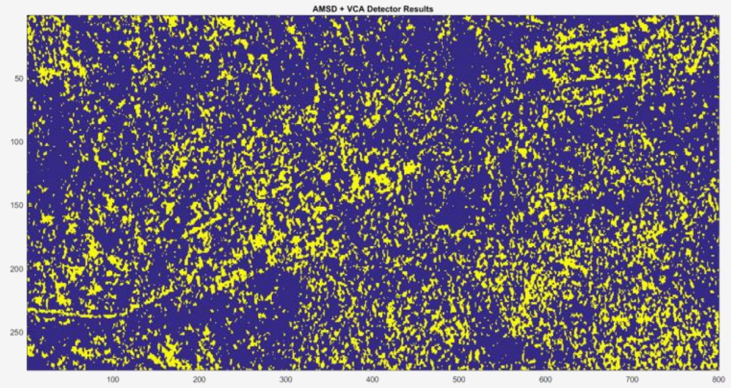
- This algorithm seems to work optimistically or because the amount of target data is small, this algorithm has low accuracy

Vehicle 1

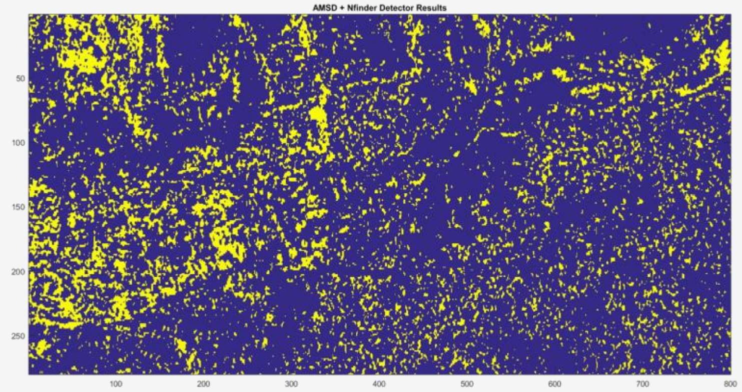
PPI



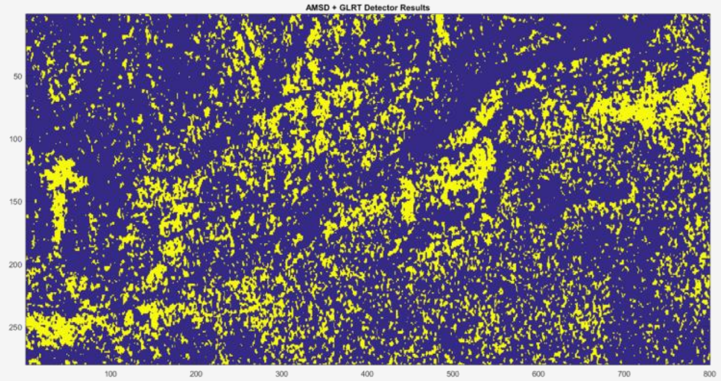
VCA



NFINDER



ATGP



Threshold Selection Result on AMSD Method

- This algorithm seems to work optimistically or because the amount of target data is small, this algorithm has low accuracy

Thank You

The End

Faezeh Zamiri

aghdam.zamiri.fa@ut.ac.ir

Linkedin.com/in/faezeh-Zamiri-1b51081bb

09224015584

References for code ,data and results:

https://github.com/FaezehZamiri/Hyperspaectralremotesensing/