Aerial Vehicle Tracking using a Multi-modal Adaptive Hyperspectral Sensor

Name : Burak Uzkent

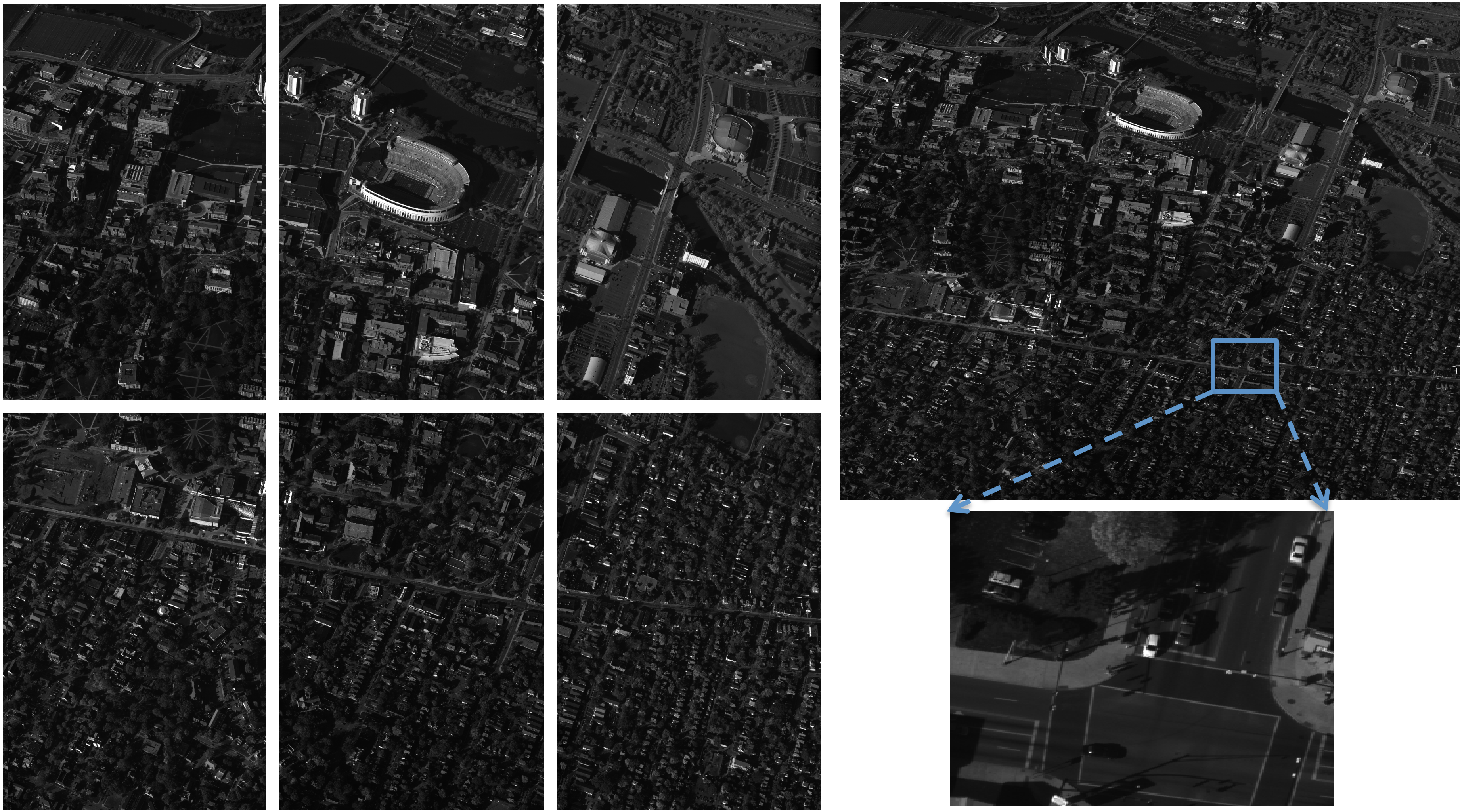
Advisor : Dr. Matthew J. Hoffman

Co-Advisor : Dr. Anthony Vodacek

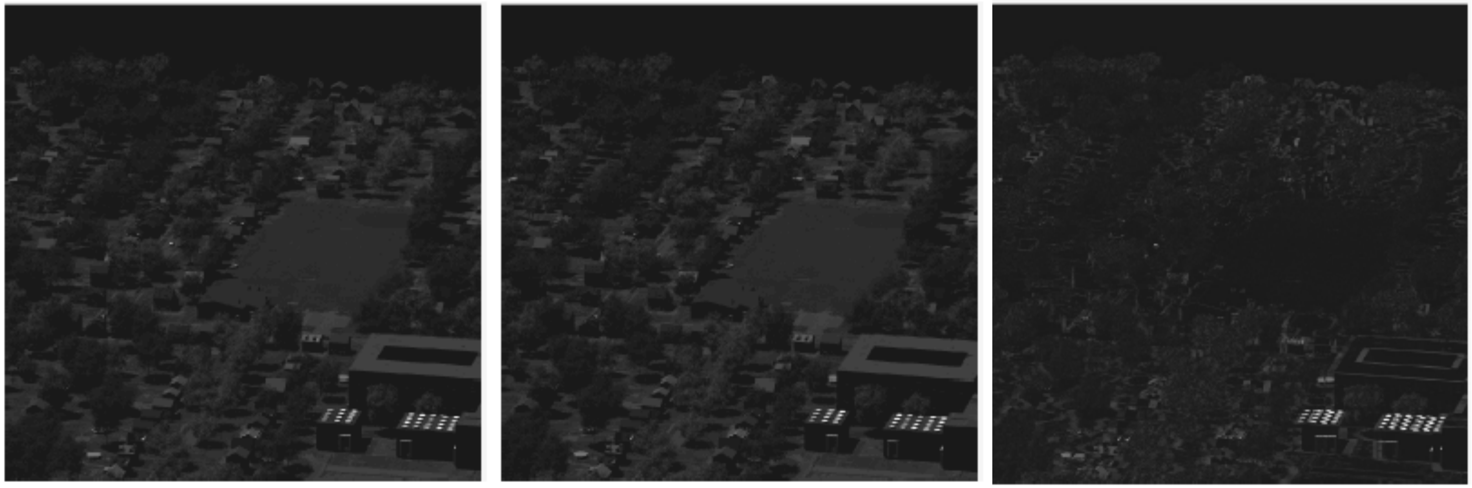
Introduction

- By using an aerial platform, we want to track all moving objects or an object of interest persistently.
- Aerial Tracking is a more challenging task than the traditional object tracking due to
  - Small number of pixels representing a vehicle
  - Large Camera Motion
  - Parallax effect due to 3-D structures in the scene.
  - Registration errors
  - Severe occlusions
- The Wide Area Motion Imagery (WAMI) Platform is the state-of-the-art sensor that is used for aerial vehicle tracking.

Low Resolution Effect - WAMI



Parallax Effect on Registration

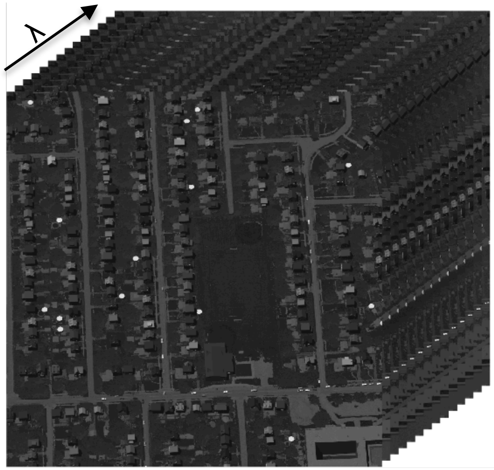


(A) (B) (E)

(C) (D) (F)

Introduction

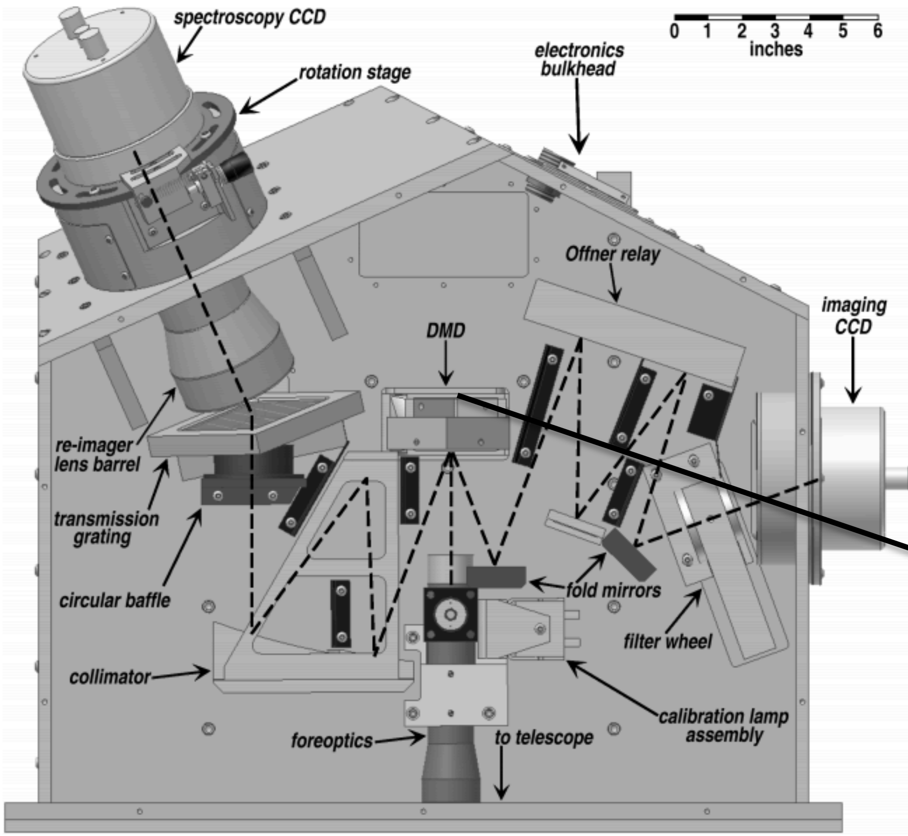
- However, still, more descriptive sensory information is required to address the challenges of aerial tracking.
- With recent advancements in the sensor technology, quick hyperspectral data acquisition is possible.



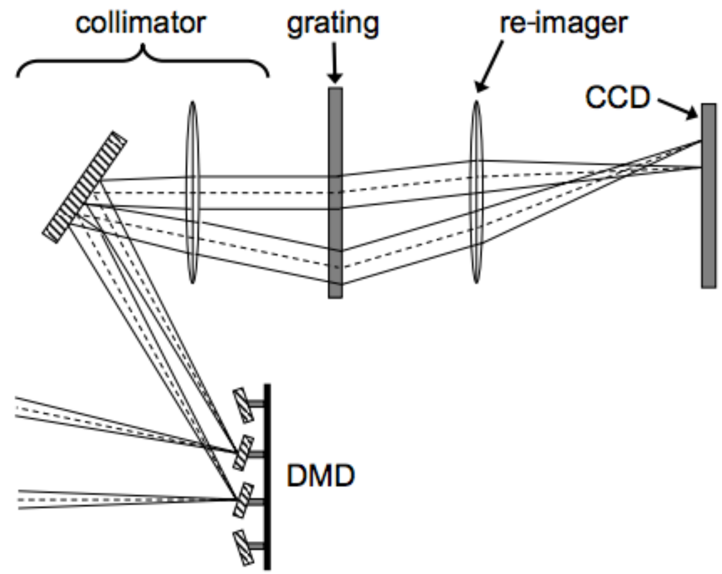
Visualization of a Hyperspectral Image Cube

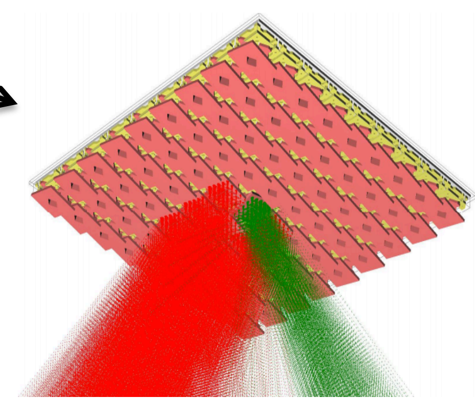
- One example of such sensor is the Rochester Institute of Technology Multi-object Spectrometer (RITMOS).

Adaptive Hyperspectral Sensor



Top View of RITMOS



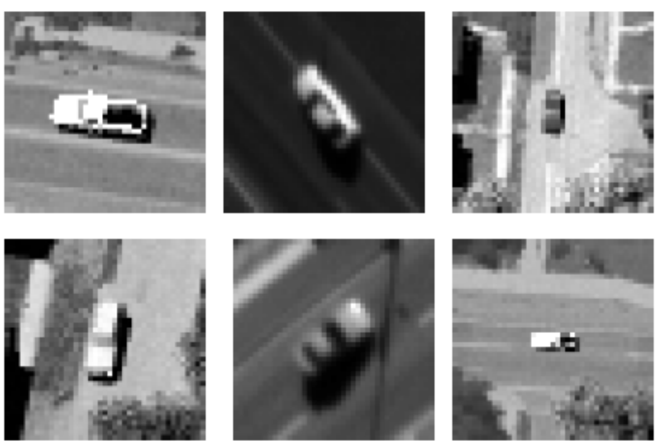


Micromirror Arrays

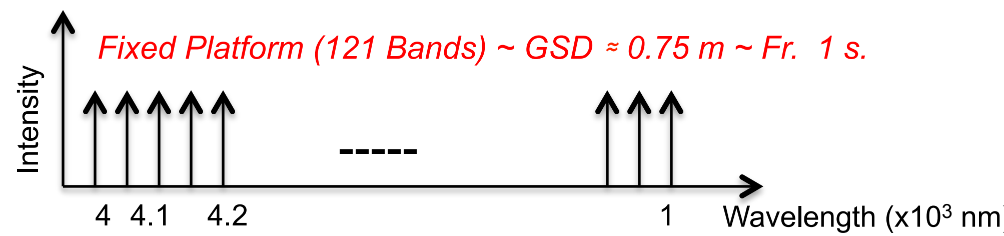
Meyer, Reed D., et al. "RITMOS: a micromirror-based multi-object spectrometer." SPIE Astronomical Telescopes+ Instrumentation. International Society for Optics and Photonics, 2004.

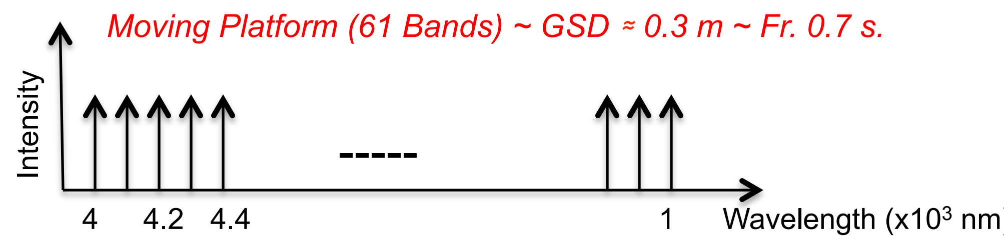
Tracking Platforms

- We simulate two different scenarios listed as
  - Fixed platform tracking.
  - Moving platform tracking from a sensor mounted onto a Drone or an Aircraft.
- Fixed platform focuses on spectral resolution whereas the moving platform focuses on higher spatial resolution.

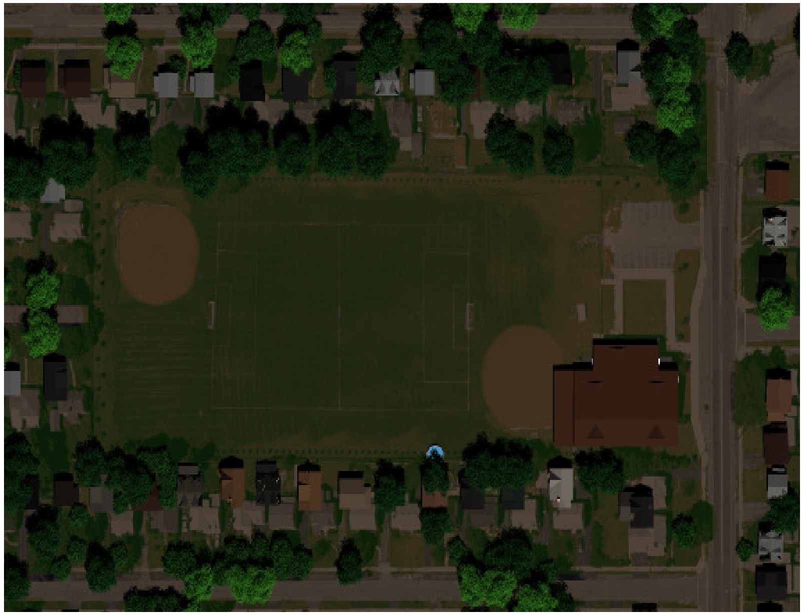


Moving Platform *WAMI* *Fixed Platform*

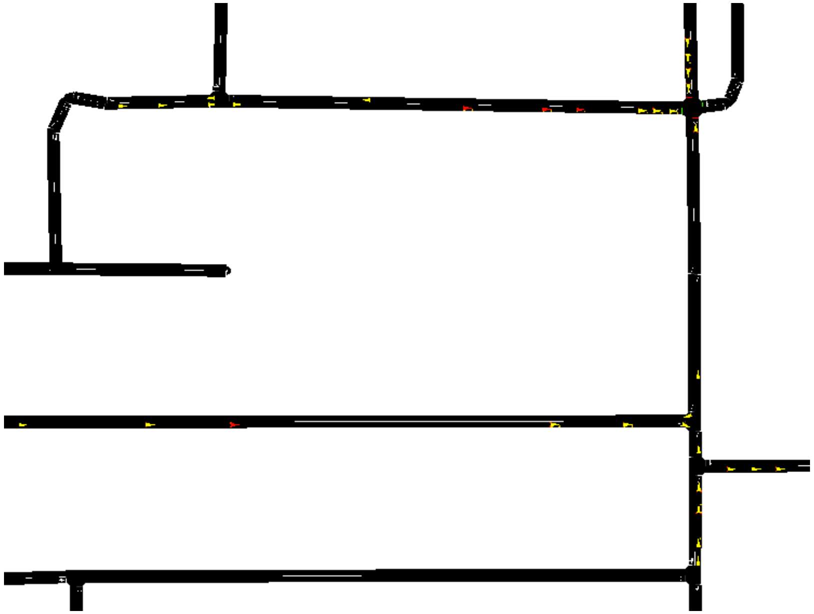




Scenario Generation from a Fixed Platform



Digital Image and Remote Sensing Platform (DIRSIG)



Simulation of Urban Mobility Platform (SUMO)

Panchromatic Image Wavelength Range	Micromirror Array Size	HSI data Wavelength Range	Focal Length	Gap between Micromirror Arrays	Platform Altitude	# Vehicles	# Paint Models	Ground Sampling Distance	# Bands	Duration
400 – 700 nm	17-17 Mm	400 - 1000 nm	225 mm	1 Mm	3000 m	89	24	0.75 m.	121	130 s.

RITMOS based simulation parameter settings

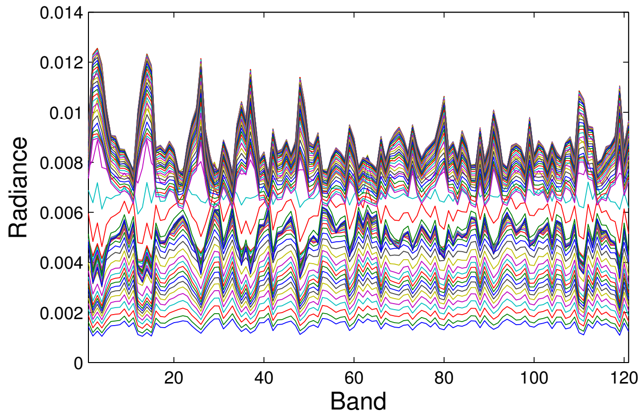
Scenario Generation

DIRSIG generated RGB image of the scenario (part of Megascene 1 area)

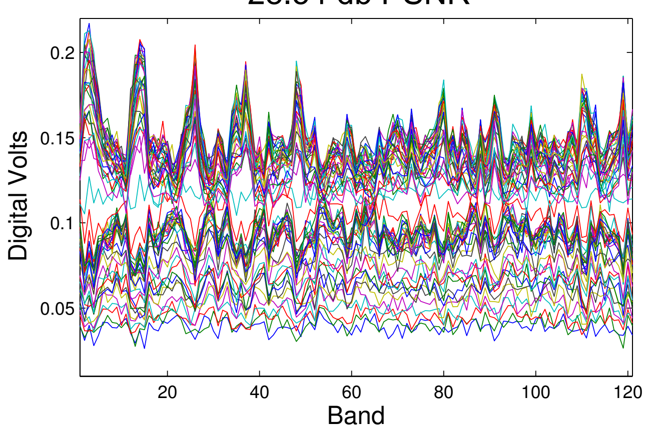
RGB image screenshot taken from Google Maps (Histogram Equalized)

Radiometric Postprocessing

- We go through a Radiometric Process to simulate a realistic data in terms of the amount noise present.
- Sensor reaching radiance values are converted to digital outputs by considering the
  - Shot Noise
  - Possion Noise
  - Saturation Noise
  - Dark Noise

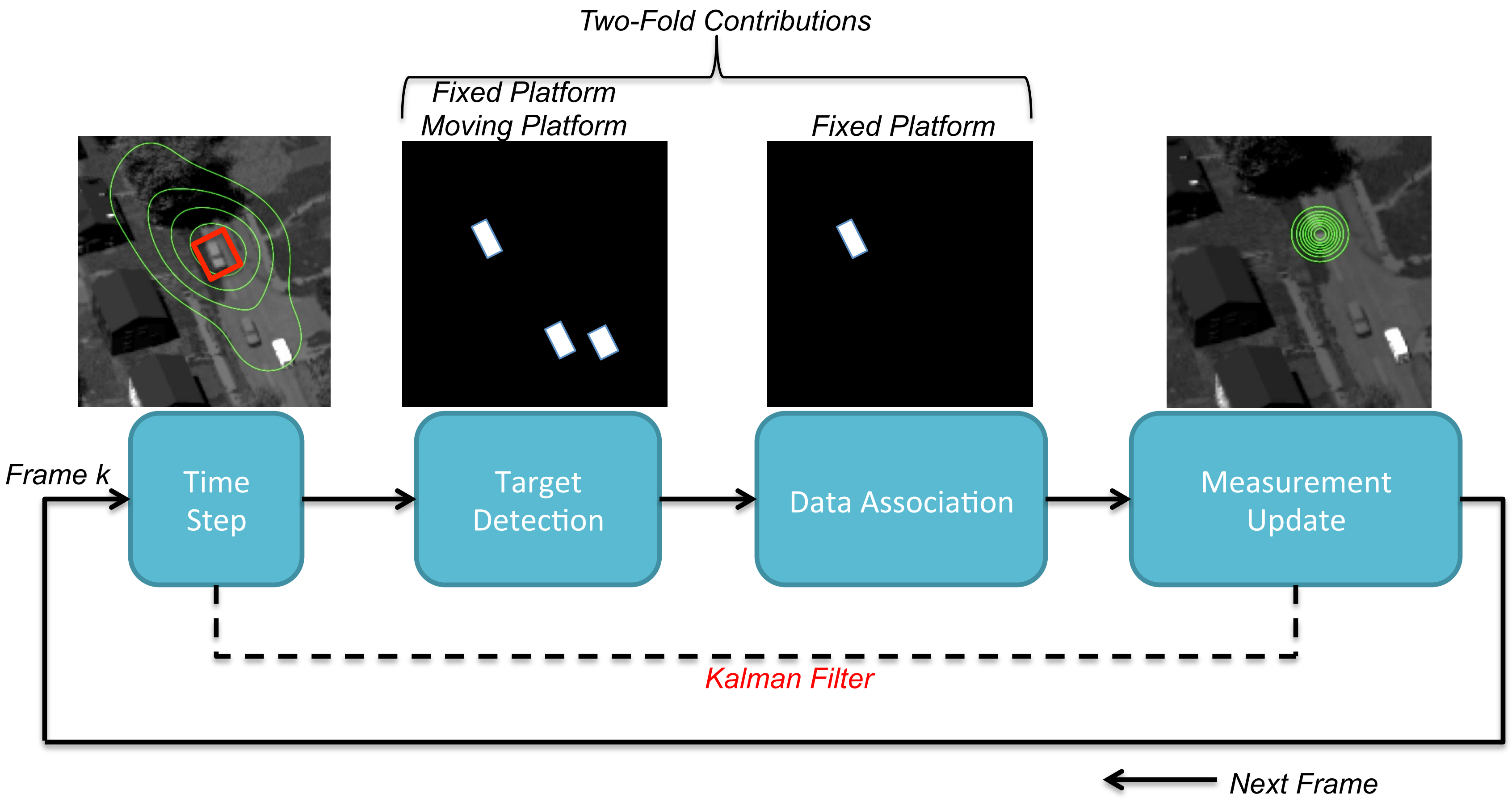


28.54 db PSNR

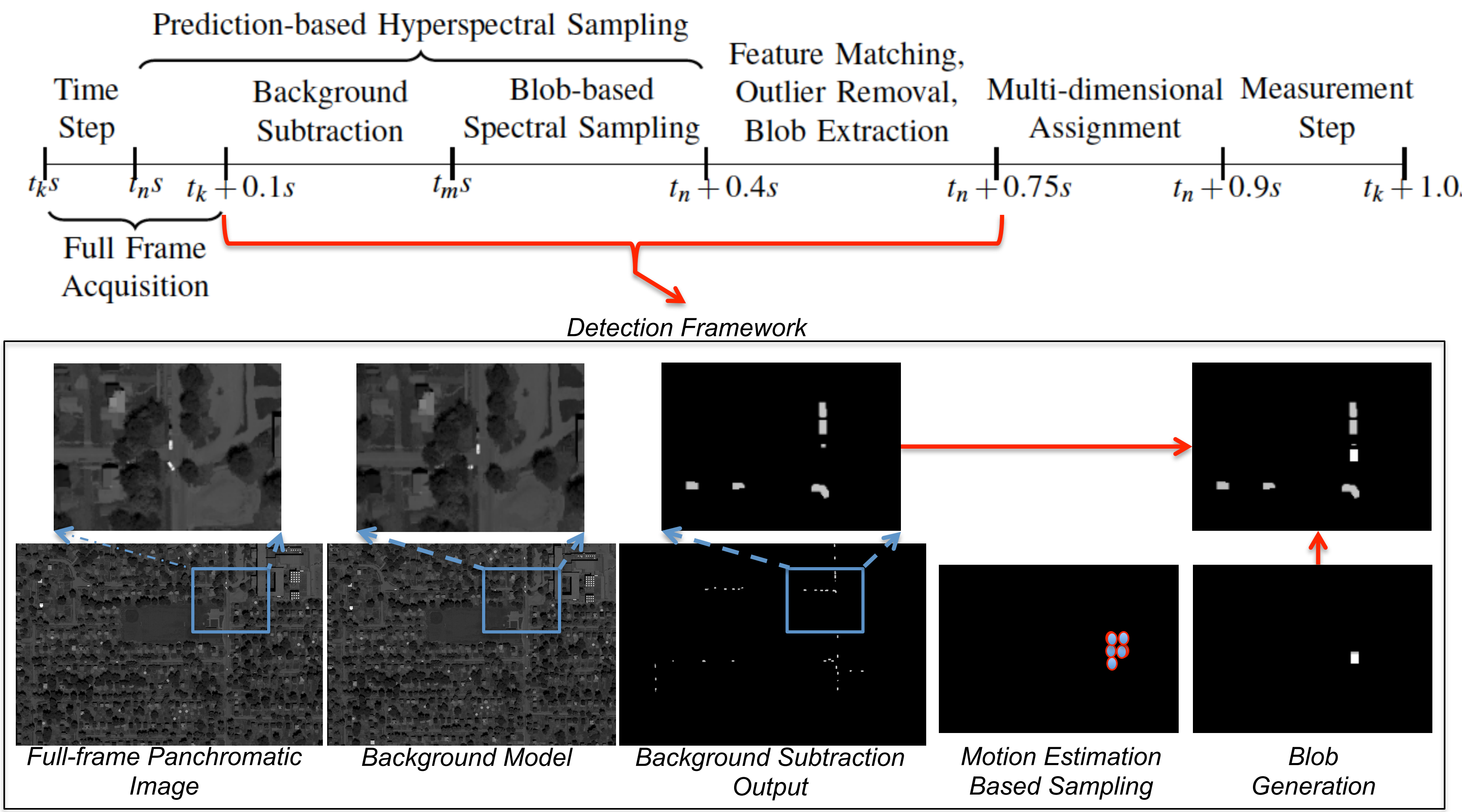


RITMOS PSNR = 30 db

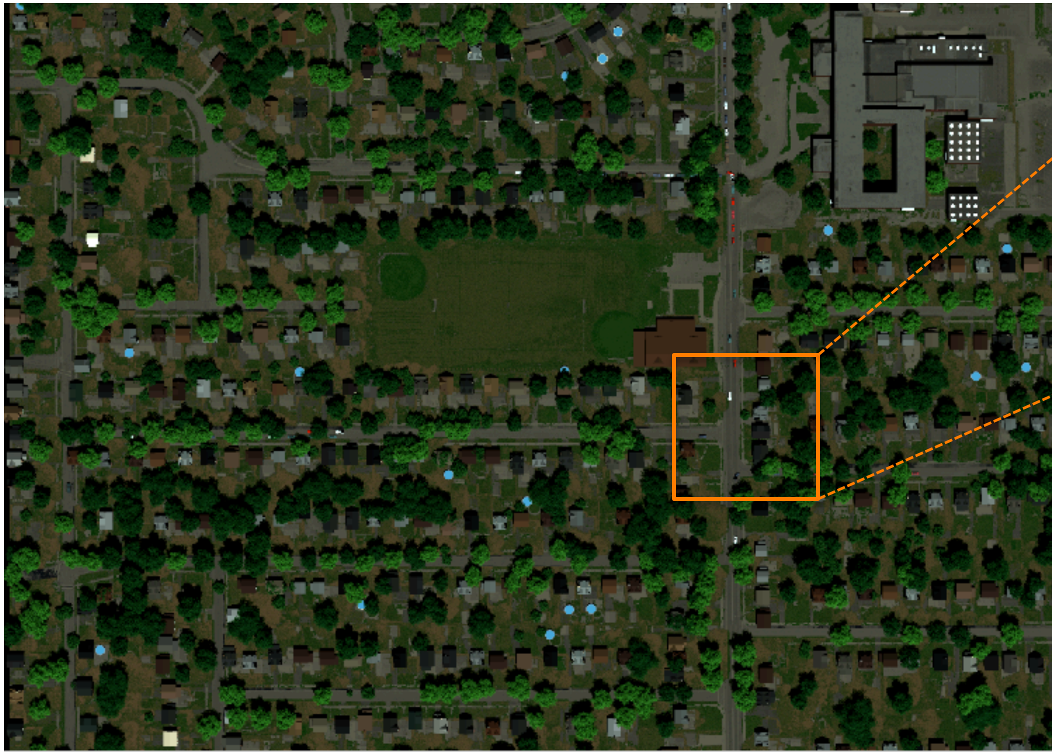
Tracking Framework



Fixed Platform Tracking (Detection)



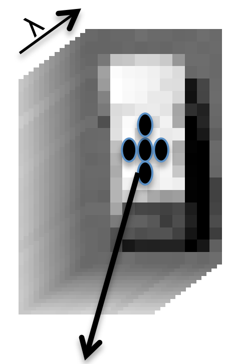
HSI Sampling for the Fixed Platform



Full Frame RGB Image of the Scene at time k (1500x1500)

Motion Detection HSI Sampling

Motion Estimation HSI Sampling



$$S_k^d = \begin{bmatrix} x_{11} & x_{21} & x_{31} & \dots & x_{m1} \\ x_{12} & x_{22} & x_{32} & \dots & x_{m2} \\ \vdots & \vdots & \vdots & & \vdots \\ x_{1b} & x_{2b} & x_{3b} & \dots & x_{mb} \end{bmatrix}$$

Feature Matrix

<u>Total HSI data acquisition time</u>

C = (D + F) * 5 ms.

D = # Blobs for Motion Detection

F = # Blobs for Motion Estimation

C < 400 milliseconds ~ (D+F) < 80

Hyperspectral Feature Matching

Blob Elimination (Motion Detection) Algorithm

Form feature matrix S_k^d for each detected blob $d = 1, ..., D$;

for $t = k - 1, ..., 2$ **do**

  if *the target is not lost at* t **then**

    $f_k^d \longleftarrow mean\{min\{SAM(S_k^d, S_t^{match})\}, min\{SAM(S_k^d, S_1^{user})\}\};$

    for $b = 1, ..., B$ **do**

      if $f_k^d > SAM\ \ Threshold$ **then**

        eliminate it;

      end

    end

    break;

  end

end

Hyperspectral Feature Matching

Virtual Blob Extraction (Motion Estimation) Algorithm

Form feature matrix G_k^d for each kernel $n = 1, ..., N$;

for $t = k-1, ..., 2$ **do**

  if *the target is not lost at* t **then**

    $f_k^n \longleftarrow mean\{min\{SAM(G_k^d, G_t^{match})\}, min\{SAM(G_k^d, G_1^{user})\}\}$;

    for $n = 1, ..., N$ **do**

      if $f_k^n < SAM\ \ Threshold$ **then**

        add n to match vector v;

      end

    end

    Cluster matched kernels, $X_{k|k-1}^v$;

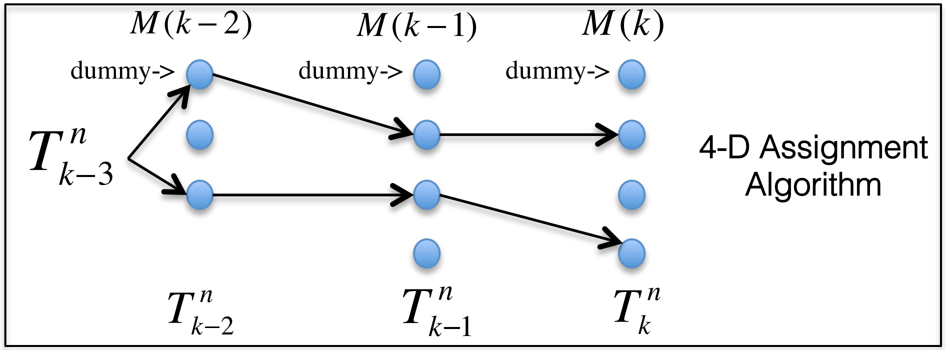
    break;

  end

end

Blob-to-Track Association

- We consider the Short-term Multiple Hyphothesis Tracker (MHT) algorithm to fuse kinematic and hyperspectral likelihoods.



1. $$C(k|A) = \sum_{y_{k-S+2}=0}^{M(k-S+2)} \sum_{y_{k-S+3}=0}^{M(k-S+3)} \sum_{y_{k-S+4}=0}^{M(k-S+4)} s(k, \{y_s\}_{s=k-S+2}^{k}, 1) c(k, \{y_s\}_{s=k-S+2}^{k})$$

2. $$c(k, \{y_s\}_{s=k-S+2}^{k}) = -ln\left(\frac{\phi(k, \{y_s\}_{s=k-S+2}^{k}, 1)}{\phi(k, \{y_s\}_{s=k-S+2}^{k}, 0)}\right)$$

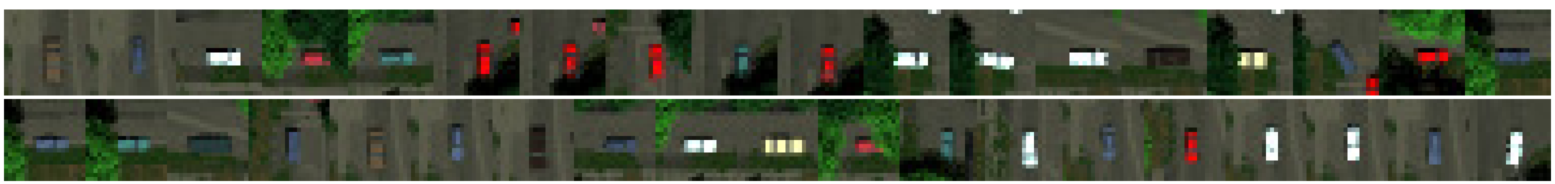
$$\tau_{kin}(s, y_s) * \tau_{feat}(s, y_s)$$

3. $$\phi(k, y_s, n) = \begin{cases} \prod_s (1 - P_D)^{1-u(y_s)} (P_D \tau(s, y_s))^{u(y_s)} & n = 1 \\ \prod_s V^{-u(y_s)} & n = 0 \end{cases}$$

Poore, Aubrey B., and Alexander J. Robertson III. "A new Lagrangian relaxation based algorithm for a class of multidimensional assignment problems." Computational Optimization and Applications 8.2 (1997): 129-150.

Results on Fixed Platform (Building)

- The baseline methods are
  - Multi-dimensional kinematic only tracker (S-D KT)
  - Hyperspectral Likelihood only tracker (HT)
  - Two-dimensional kinematic only tracker (2-D KT)
  - Multi-dimensional feature-aided tracker without HSI based Blob Extraction step (S-D FAT W/o Blob Ext.)
  - Multi-dimensional feature aided tracker without hyperspectral likelihood integration (S-D FAT W/o HSI L-hood)



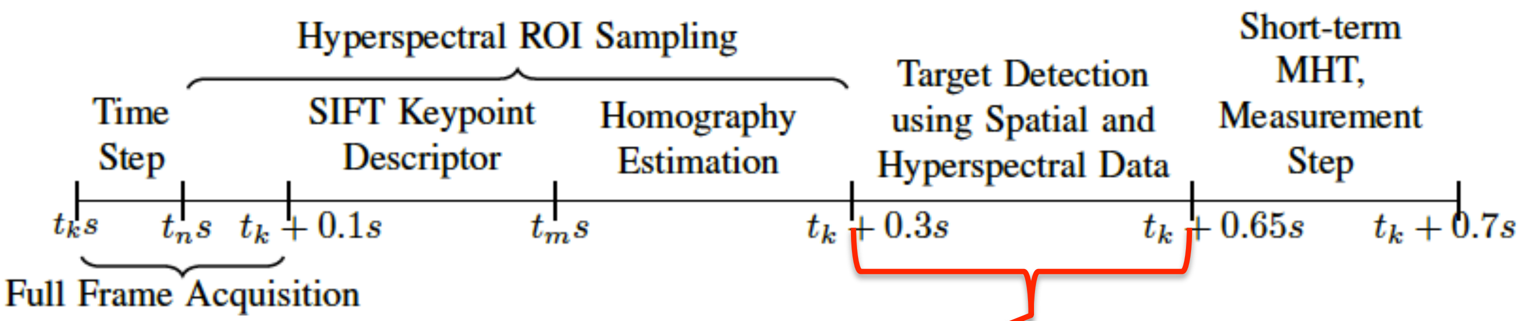
Metric / Tracker	6-D (KT)	HT	2-D (FAT)	6-D (FAT) W/o HSI L-hood	6-D (FAT) W/o Blob Ext.	6-D (FAT)
Precision	24.88	39.82	39.23	44.12	32.88	57.63
Recall	24.88	35.85	39.15	43.64	29.38	57.13

Average Precision and Recall values for the Fixed Platform Tracking

B. Uzkent, M. J. Hoffman, and A. Vodacek, "Integrating Hyperspectral Likelihoods in a Multidimensional Assignment Algorithm for Aerial Vehicle Tracking", IEEE Journal Of Selected Topics In Applied Earth Observations And Remote Sensing (Accepted).

Tracking from a Moving Platform (Drone)

Proposed Tracking Framework for the Moving Platform



Target Detection Framework for the Moving Platform

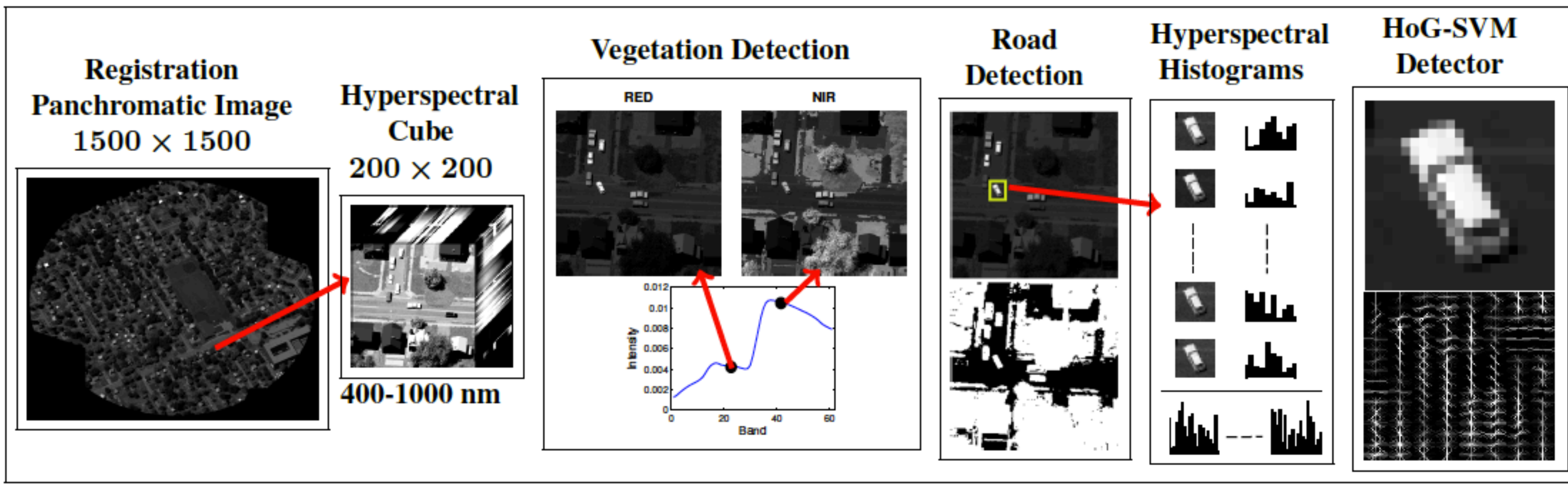
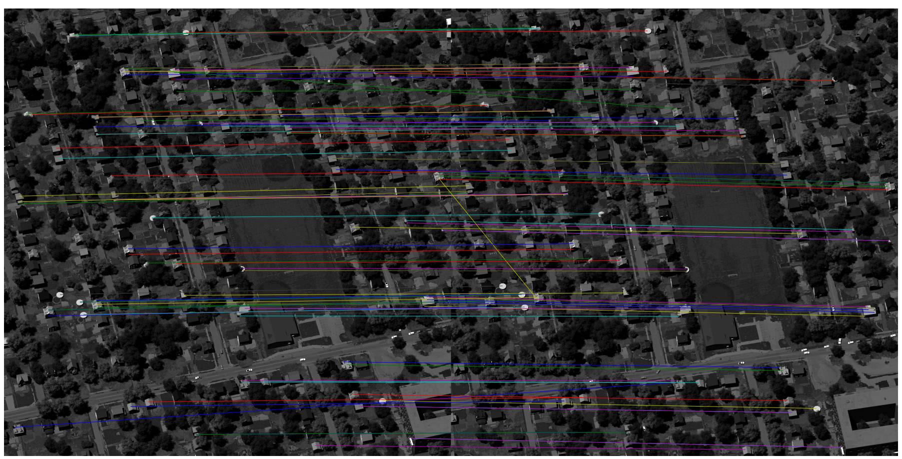
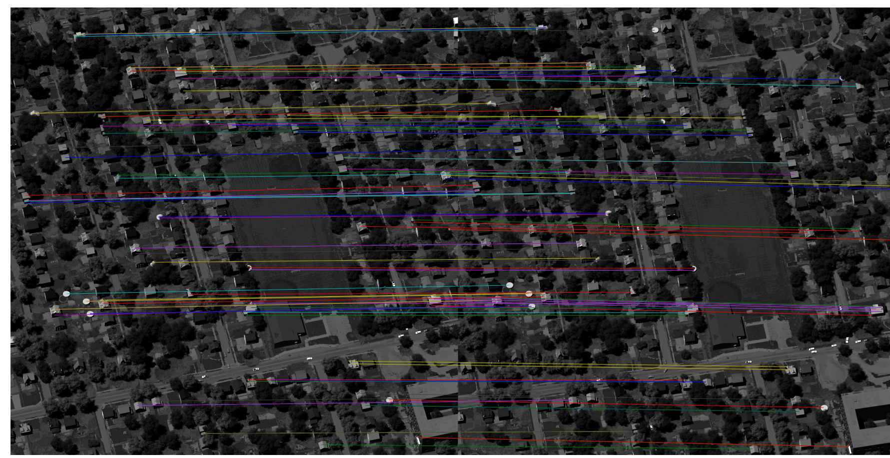


Image Alignment

- Panchromatic images are used to compute homography between the reference and input frame.
- We use the SIFT to find keypoints and describe keypoints with gradient histogram features.
- The keypoints are matched and RANSAC is used to remove outliers and compute homography.



Matched Keypoints

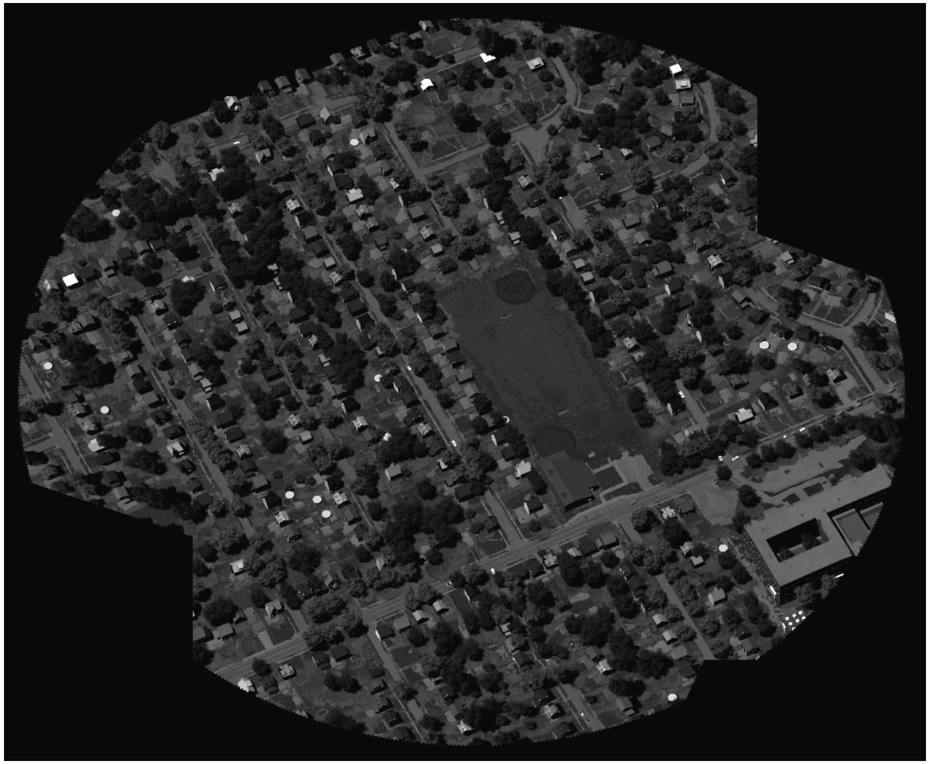


After RANSAC

Image Alignment

- The homogaphy, *H*, between frame *1* and frame *k* is computed as

$$H_{k,1} = H_{k,k-1} * H_{k-1,k-2} * H_{k-2,k-3}, \ldots, H_{2,1}$$



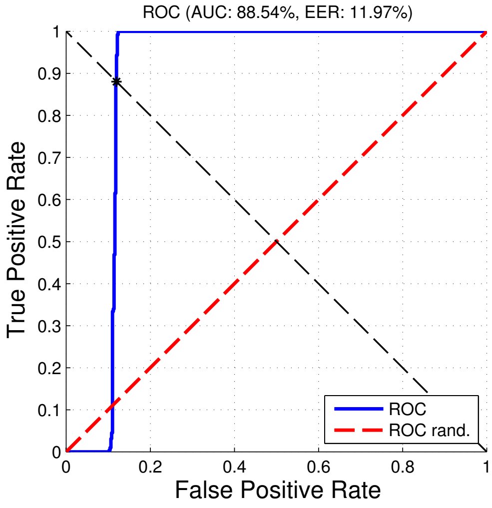
Aligned images of the entire video overlaid to a canonical frame

Background Removal

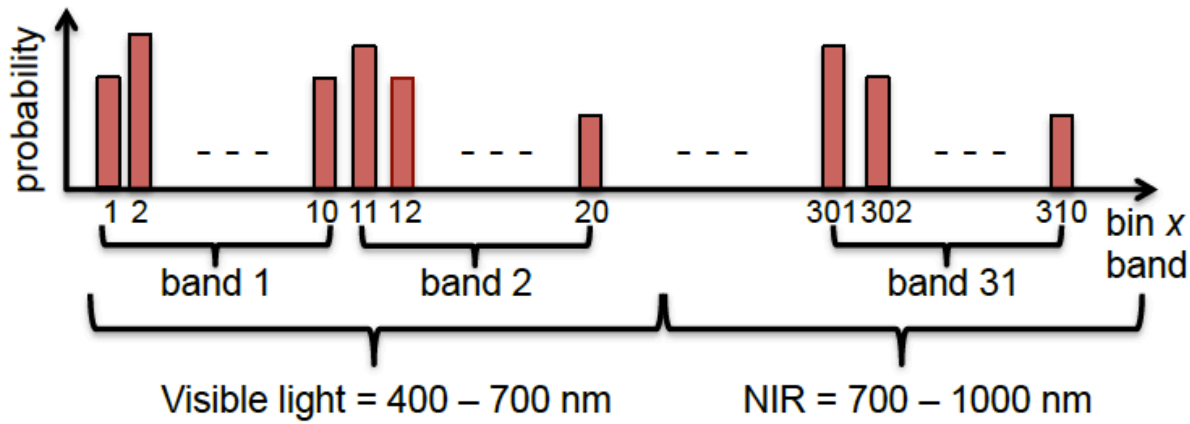
632 nm

800 nm

$$V_{MAP}(x,y) = \frac{I_{NIR}(x,y) - I_{RED}(x,y)}{I_{NIR}(x,y) + I_{RED}(x,y)}$$

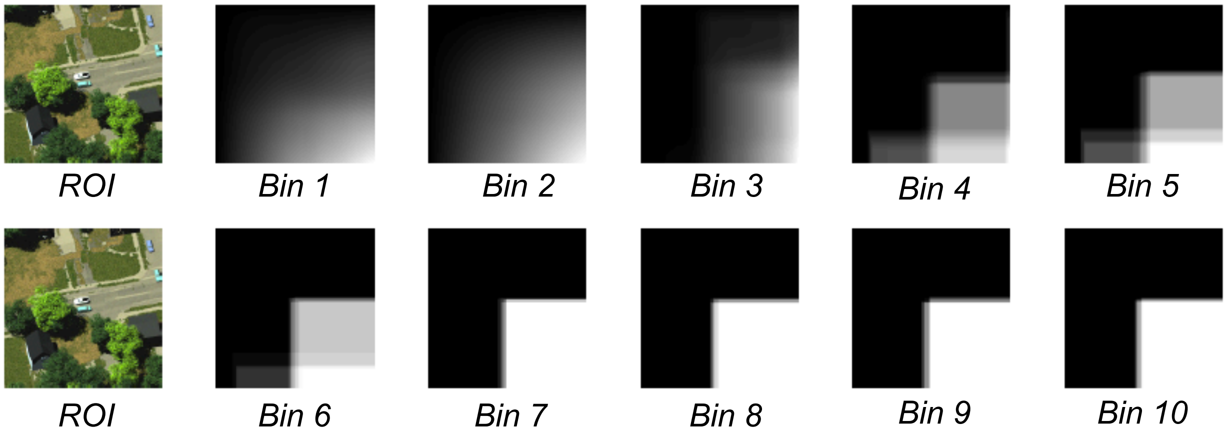


Hyperspectral Similarity Score Assignment



**Computing hyperspectral histogram for 40000 pixels is costly.*

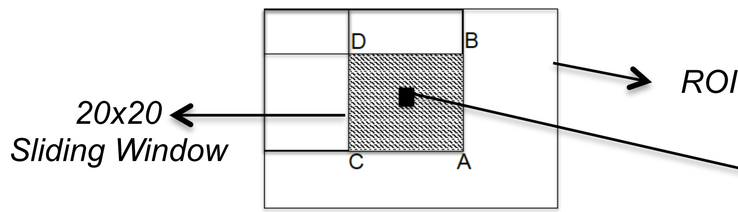
- Gradient Integral images are computed for each bin of every band.



310 integral Images are computed in total.

Hyperspectral Similarity Score Assignment

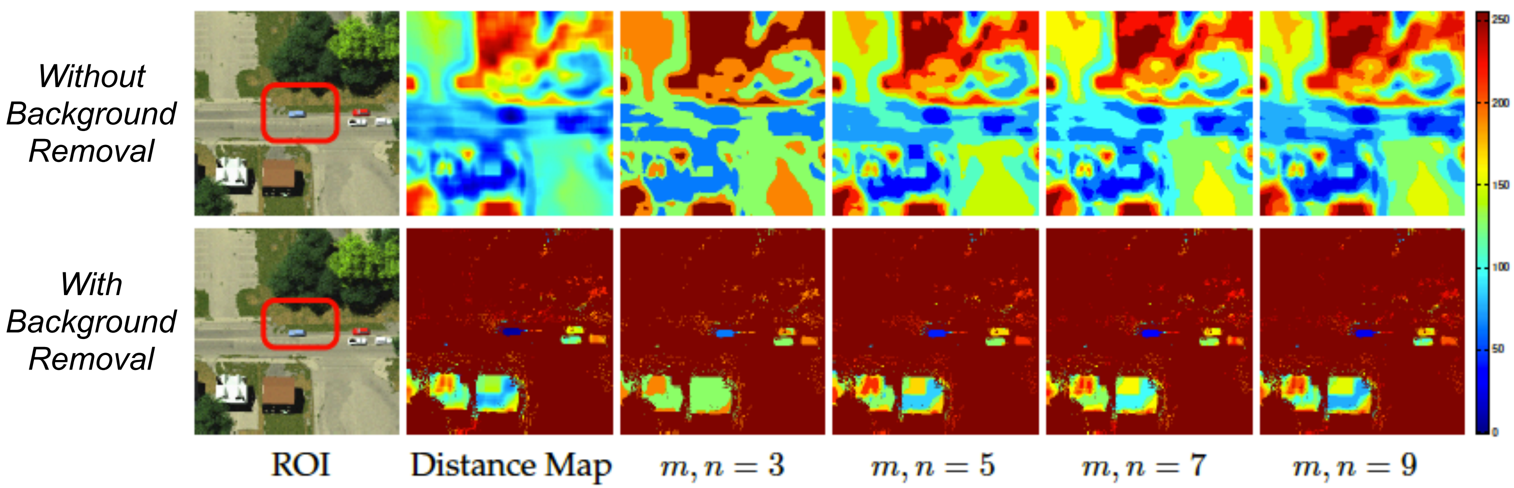
- In O(nx3), we can compute a hyperspectral histogram of a pixel.



$$ROI(x_i, x_j, n_m) = A + D - B - C$$

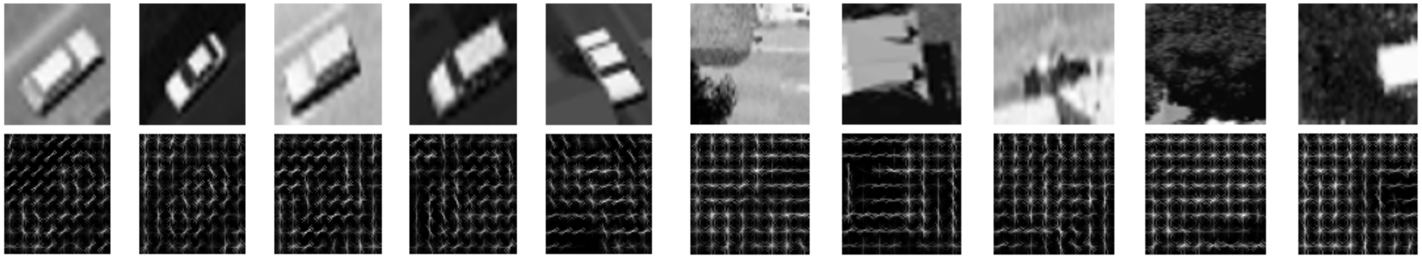
Integral Image Theorem

- Likelihood maps are applied a threshold by with the Multilevel Otsu's Threshold method.

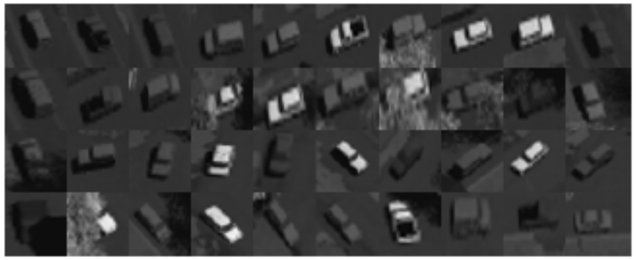


Vehicle Confirmation Module

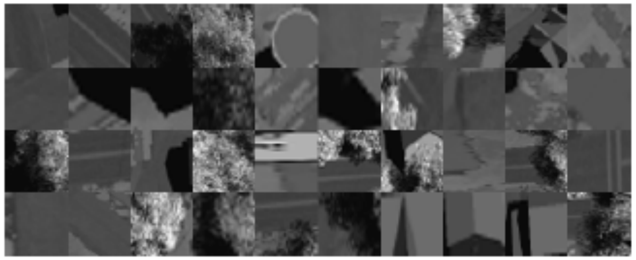
Vehicle Confirmation Workflow



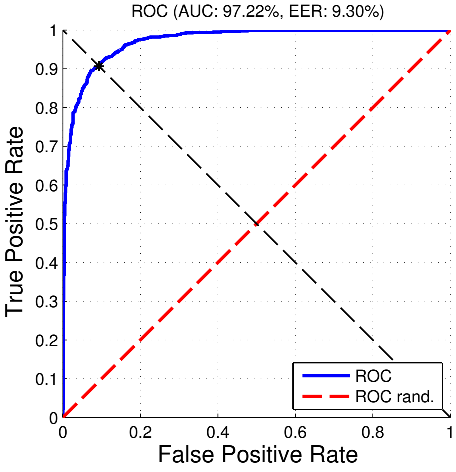
Blobs and Corresponding HoG Features



Positive Samples in Training Set (2400 Samples)



Negative Samples in Training Set (2400 Samples)



Results on Moving Platform (Drone)

- The baseline methods are
  - Nearest Neighbor Tracker (NN)
  - Probabilistic Data-Association Filter (PDAF)
  - Multiple Hypothesis Tracker Filter (MHT)
  - Mean-shift Tracker
  - A state-of-the-art traditional object tracker (OFDS)
  - Likelihood of Features Tracking tracker (LoFT)

- - - -  - - - -

Metric / Tracker	NN	PDAF	MHT	Mean-Shift	OFDS	LOFT	Ours
Precision	39.25	26.17	39.20	8.88	12.66	60.30	69.78
Recall	34.65	14.19	35.07	8.88	12.66	40.50	60.35

Average of 43 Tracks

Module	NDVI	Road Classifier	Score Assignment	HoG-SVM
Run Time	0.002 s.	0.08 s.	0.2 s.	0.05 s.

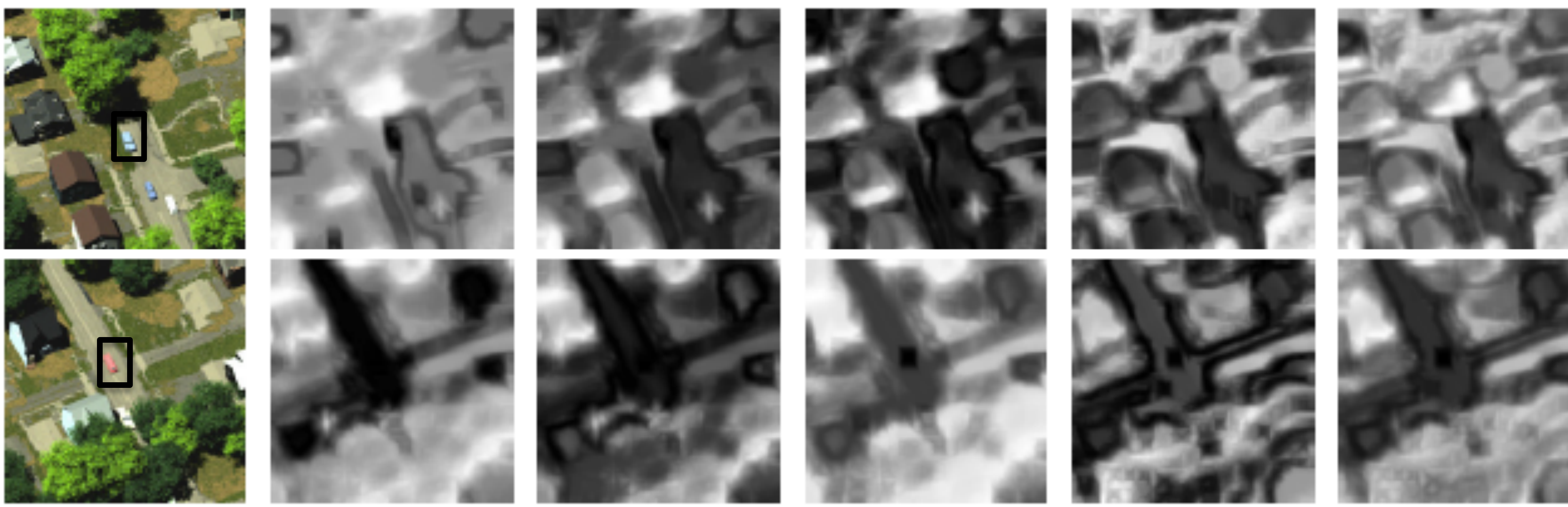
Run times for Detection Modules (Tested on a personal computer with 2.8 GHz I7 processor and 8GB RAM)

B. Uzkent, M. J. Hoffman, and A. Vodacek, " Real-time Tracking in Aerial Video using Hyperspectral Features", Computer Vision and Pattern Recognition Workshop (CVPRW), 2016. (Accepted)

The 2nd Approach for the Moving Platform

- Some vehicles are better separated from the background pixels in a certain wavelength range.

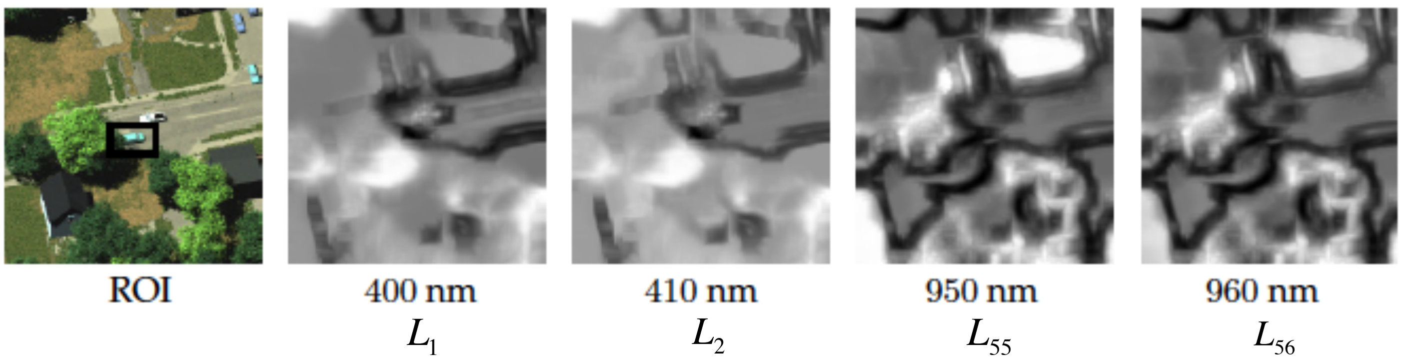
$$L_{Final} = \sum_{i}^{N} w_i L_i$$



ROI | 430 nm L_4 | 530 nm L_{14} | 630 nm L_{24} | 930 nm L_{54} | Classic Fusion L_{Final}

Adaptive Likelihood Map Fusion

- Computing a likelihood map for each band is costly.
- The likelihood maps of the adjacent bands are correlated.



- Variance Ratio method computes coefficients on the likelihood maps.

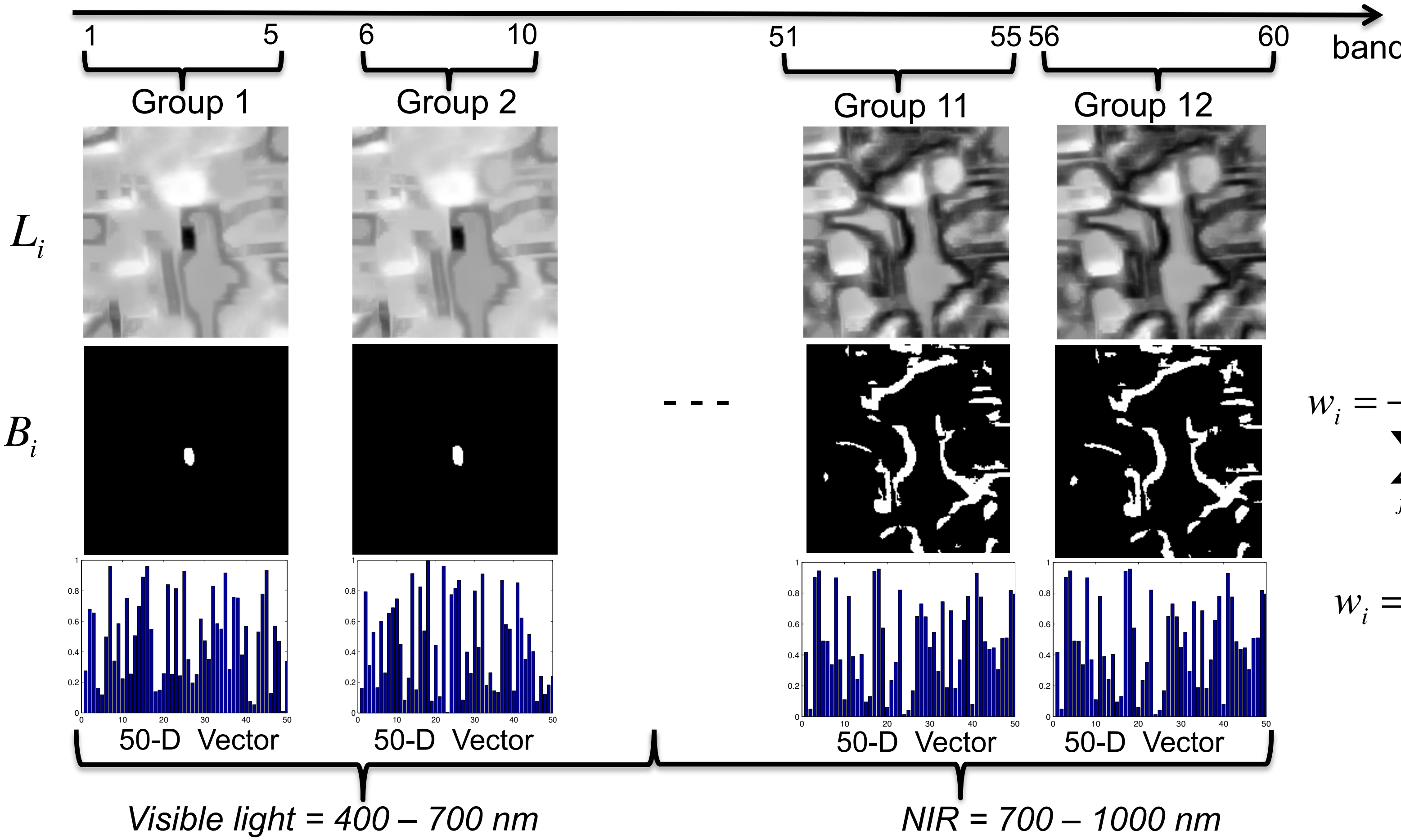
$$VR(L_i, t, b) = \frac{\mathrm{var}(L_i; (t+b)/2)}{\mathrm{var}(L_i; t) + \mathrm{var}(L_i; b)}$$

$$w_i = \frac{VR(L_i; t, b)}{\sum_j^N VR(L_j; t, b)}$$

Yin, Zhaozheng, Fatih Porikli, and Robert T. Collins. "Likelihood map fusion for visual object tracking." Applications of Computer Vision, 2008. WACV 2008. IEEE Workshop on. IEEE, 2008.

Proposed Grouping Method

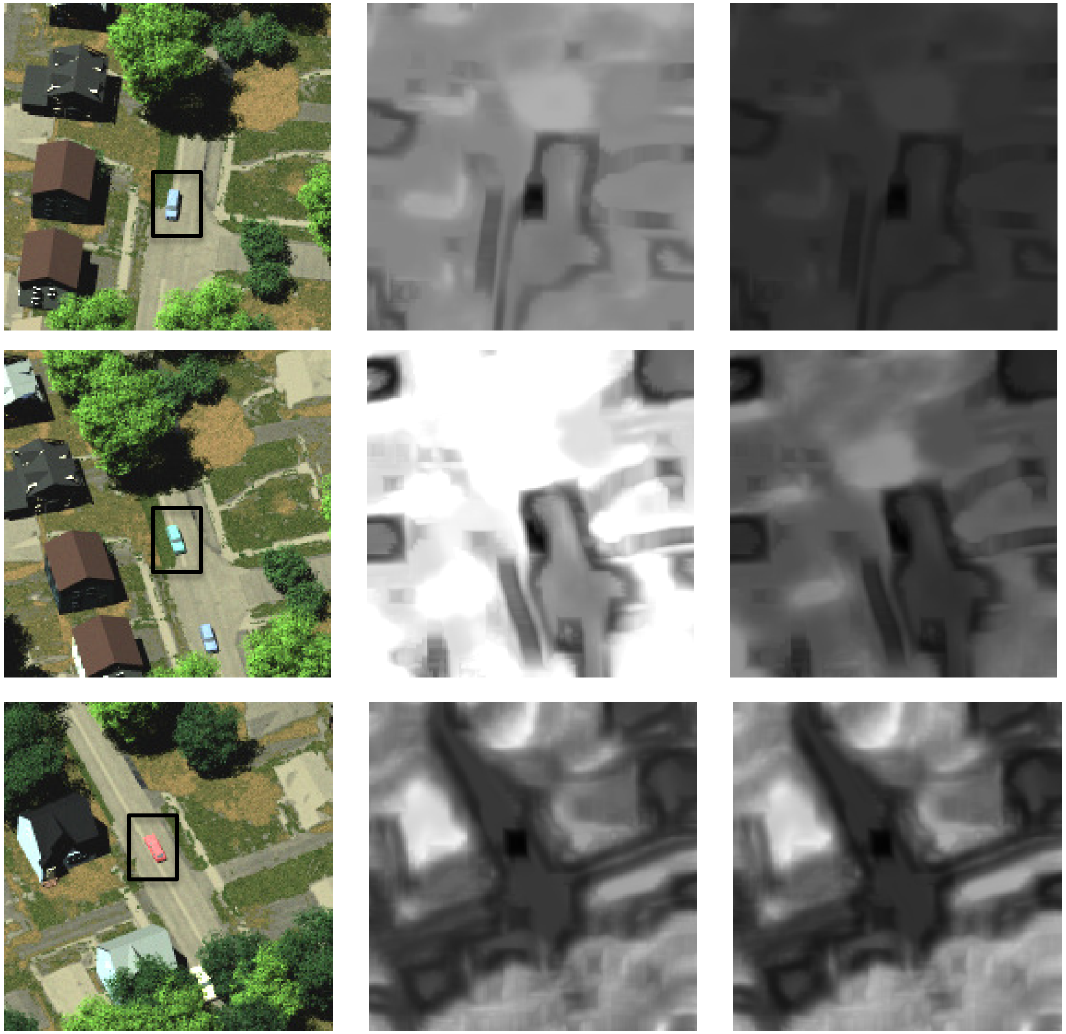
12 Groups and 60 Bands



$$w_i = \frac{\sum_{r=1}^{200}\sum_{c=1}^{200} B_i(r,c)}{\sum_{j=1}^{N}\sum_{r=1}^{200}\sum_{c=1}^{200} B_j(r,c)}$$

$$w_i = \frac{1}{1+e^{-k(w_i-x_0)}}$$

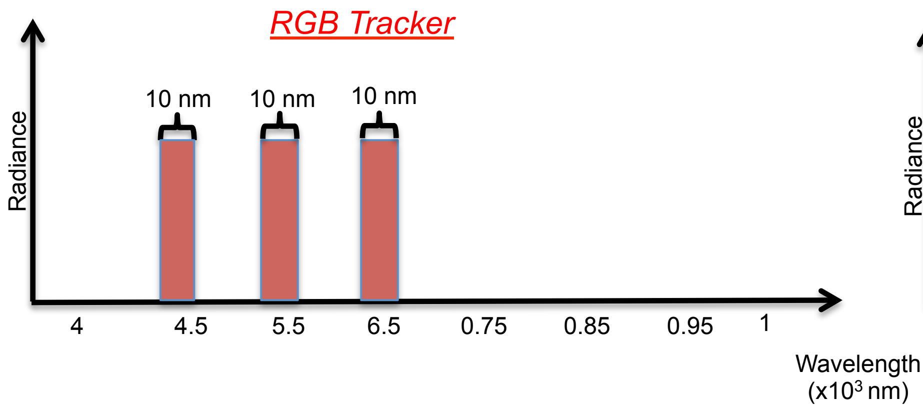
Adaptive Map Fusion Results

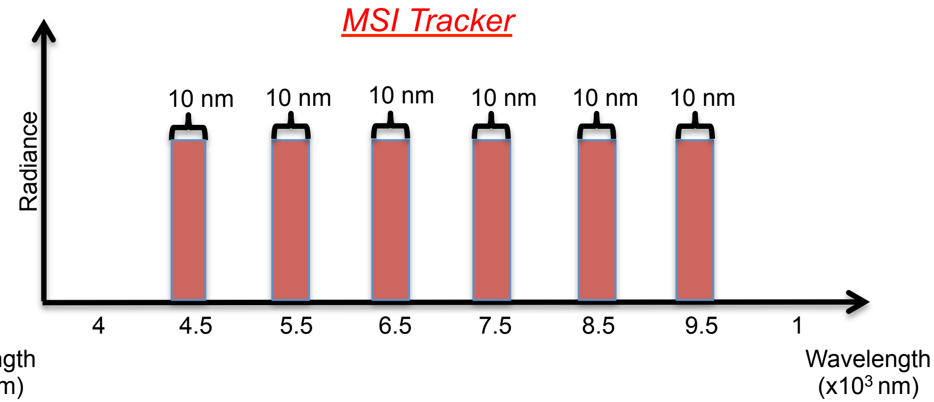


ROI *Proposed Fusion* *Classic Fusion*

Adaptive Map Fusion Results

- The baseline methods are
  - Gray-scale Image Tracker with Same Method (Gray-scale)
  - RGB Image Tracker with Adaptive Map Fusion (RGB)
  - Multispectral Image Tracker with Adaptive Map Fusion (MSI)
  - Hyperspectral Image Tracker with Classic Map Fusion
  - Hyperspectral Image Tracker with Variance Ratio Method based Map Fusion
  - 6-D Hyperspectral Likelihood-aided Tracker (6-D FAT)





Results for the 2^{nd} Approach

Metric / Tracker	Gray-Scale	RGB	MSI	HSI (Classic)	HSI (VR)	6-D FAT	Ours
Precision	28.43	39.20	55.12	50.17	48.26	69.78	64.37
Recall	04.28	35.07	50.91	45.25	44.56	60.35	57.49

Experiments on Different Challenge Levels

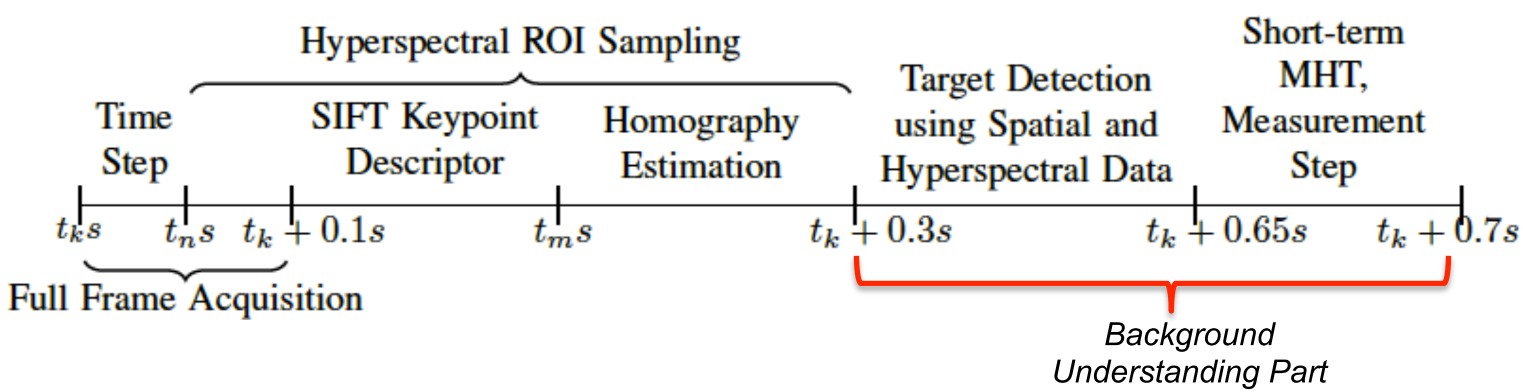
PSNR-SAM	23.49 db - 8.3 SAM	24.85 db - 7.7 SAM	26.10 db - 7.0 SAM	27.22 db - 6.6 SAM	28.11 db - 6.3 SAM	28.93 db - 6.3 SAM	30.30 db - 5.8 SAM
Precision	43.74	47.60	55.81	62.28	61.27	66.12	64.37
Recall	38.31	42.42	48.95	54.89	56.71	58.01	57.49

Metric	With Trees	Without Trees
Precision	64.37	82.54
Recall	57.39	80.54

Metric	12:00 PM	15:00 PM
Precision	64.37	82.54
Recall	57.39	80.54

Conclusions

- We considered an adaptive hyperspectral sensor to improve tracking in adverse scenes.
- We proposed tracking algorithms that tackle:
  - Frequent stop-then-go motions in fixed platforms case
  - Large uncertainty in blob-to-track association by integrating hyperspectral likelihoods
  - Background subtraction-free target detection task in moving platform case.



Future Work

- More research is required to automatically find the best number of hyperspectral bands and likelihood maps in the adaptive likelihood maps fusion method.
- The proposed approaches need to be tested with real aerial hyperspectral data.
- Can we train a deep CNN model to detect vehicles in a ROI using hyperspectral data?
- Can we train it to classify vehicles into a number of groups based on their paint models?
- Can we train a deep CNN with synthetic data and fine-tune it with real hyperspectral data?