ML and Space Applications

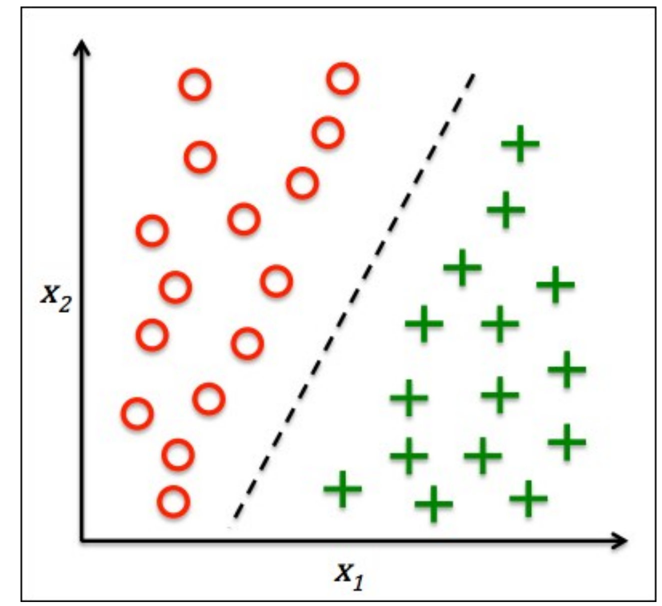
Deepak Mishra
Associate Professor
Dept. of Avionics, IIST Trivandrum
deepak.mishra@iist.ac.in

What is Machine learning ?

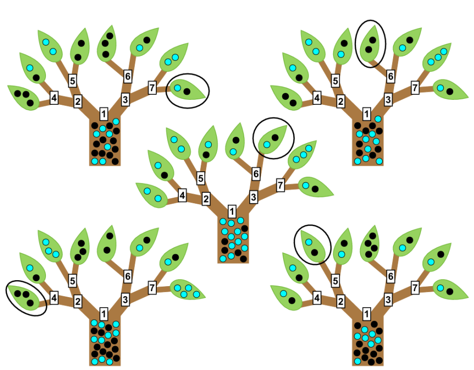
- Promise: Give me Data, I return knowledge/information/expertise
- It is the science of learning from data
- Consists of optimization + probability + many task/data specific tricks such as pre-processing/post-processing/visualization/etc.
- Often, a task specific, data dependent objective function is written and optimized
    - In many problems, the learning is nothing but optimizing an objective function to obtain the value of unknown variables
    - Objective function to optimize is data dependent
    - Values of these variables are the knowledge/information/expertise
    - Will provide an example to discuss this further

Popular ML methods

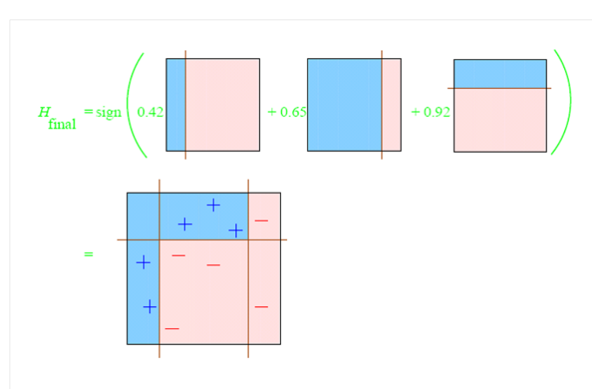
- Supervised learning
- Need a set of training cases (entities and their labels)
- Cases are usually represented by a feature vector extracted from a complex entity (e.g. graph or time series)
- A held-out set (test set) is used to measure how the learned model performs (generalization capability)



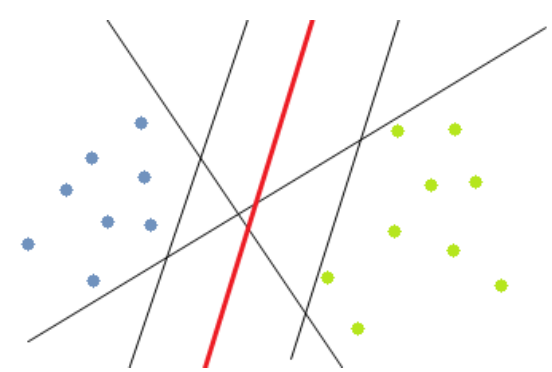
Random Forest



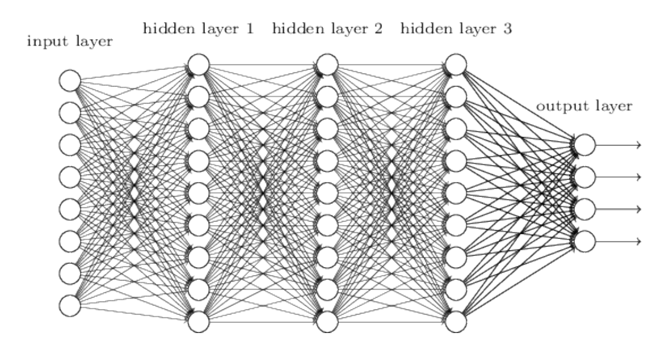
Gradient Boosting



SVM



Neural Network (Deep)



Machine Learning for Space Problems

- Engineering Problems
  - Helps to run or maintain the space instrument better (Anomaly detection, prediction of the life of any equipment etc.)
- Science Problems
  - Increases our knowledge of the space (identifying and classifying galaxies and or exoplanet etc.)

How Does NASA Use Machine Learning?

- Self-Driving Rovers on Mars – The Spirit and Opportunity Rovers

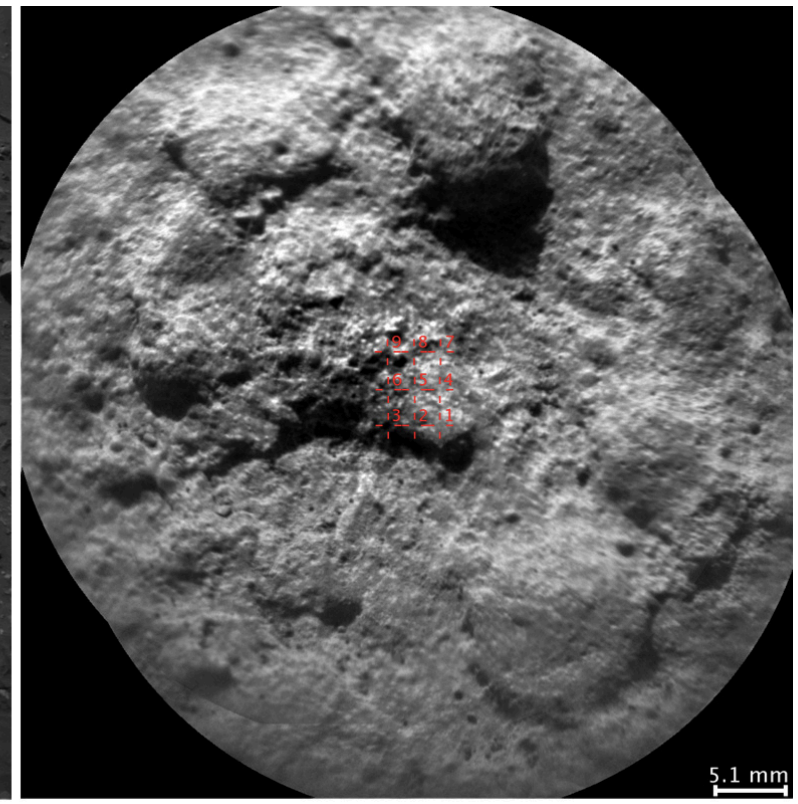
A Machine Learning based navigation and driving system for self-driving Mars rovers known as AutoNav was actually used in the Spirit and Opportunity rovers which landed on Mars as early as 2004.

source: https://scienceandtechnology.jpl.nasa.gov/research/research-topics-list/communications-computing-software/deep-space-navigation

AEGIS (Autonomous Exploration for Gathering Increased Science)

- Another application of Machine Learning in the Mars rovers is an algorithm called **AEGIS** (Autonomous Exploration for Gathering Increased Science) which identifies martian rock formations that might be interesting on its own by using Machine Learning This is because the rover cannot send all the pictures of Mars it snaps back on Earth because there is only limited communication possible. So AEGIS decides which pictures might be interesting or important and then the rover sends them back on Earth for the NASA scientists to study.

Original Navcam



Follow-up ChemCam

NASA's Curiosity Mars rover autonomously selects some targets for the laser and telescopic camera of its ChemCam instrument. For example, on-board software analyzed the Navcam image at left, chose the target indicated with a yellow dot, and pointed ChemCam for laser shots and the image at right.

Medicine in Space – Exploration Medical Capability (ExMC)

- Now that astronauts are moving further and further into space beyond the Earth orbit, what will happen if they need medical help? They will obviously not be able to return to Earth for a check-up with a doctor! For this reason, NASA is working on **Exploration Medical Capability** that will use Machine Learning to develop healthcare options based on the anticipated future medical needs of the astronauts. These healthcare options will be created by certified doctors and surgeons and they will learn and evolve with time according to the astronaut experiences.

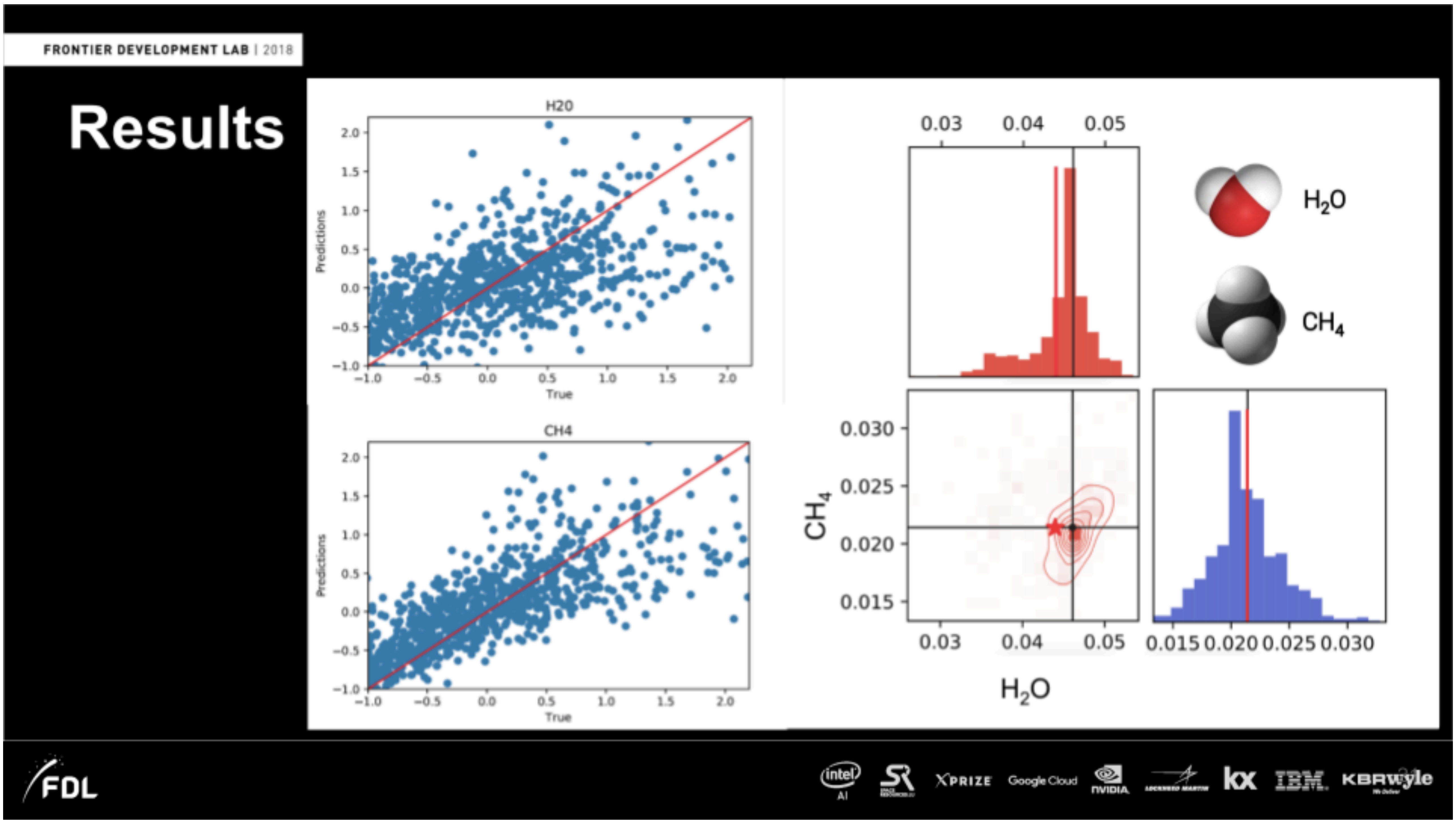
Locating Other Planets in the Universe – Planetary Spectrum Generator

- there are around **100 billion** stars in the galaxy and out of them about **40 billion** may have life. This is not science fiction, NASA actually believes we may find aliens one day! But for discovering aliens, NASA first needs to discover more and more new planets in different solar systems. Once these **exoplanets** are discovered, then NASA measures the atmospheric spectrum of these planets to find if there is any possibility of life.

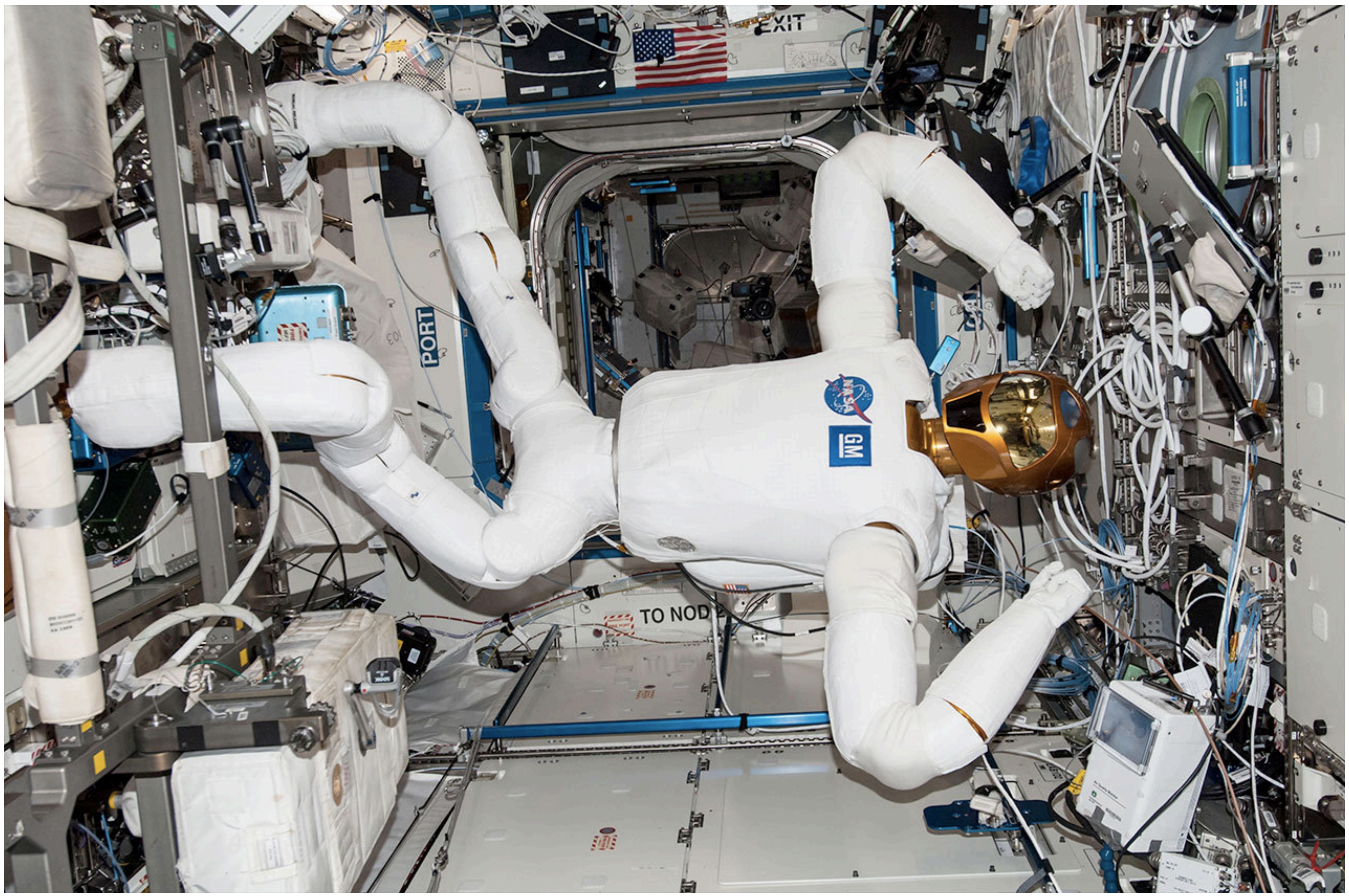
- While these steps are complicated enough, the problem is that there is no real data available for experimentation! So NASA scientists just generate the required data and that's where Machine Learning comes in.

- The **Planetary Spectrum Generator** is a tool that NASA uses to create **3-D orbits** and **atmospheric properties** of the exoplanets they find. To create a working model for the solar system, scientists use **linear regression** as well as **convolutional neural networks**. Then further fine-tuning is conducted on the model before it is ready for training.

The below image demonstrates the results generated for an exoplanet that demonstrate the amount of water and methane in the atmosphere. As you can see in the CH4 and H2O graph, the black lines denote the predictions that were made using Machine Learning and the red lines indicate the actual findings. As you can see the trained ML model is quite accurate in this situation!



A Robotic Astronaut – The Robonaut



Robonaut

- NASA has developed a **robotic astronaut**. The Robonaut was primarily developed to work alongside the astronauts in space and help them in completing tasks that were quite dangerous for humans. This was necessary as it would increase NASA's capacity for research and discovery in space which would, in turn, allow us to learn more about the solar system.

- Robonaut basically uses **Machine Learning** to “think” for itself. So the scientists or astronauts can give tasks to the Robonaut and it figures out how to perform them.

- In general, Robonaut also has many advantages over normal humans like **advanced sensors, insanely high speeds, compact design,** and much **higher flexibility**. There is a lot of advanced technology that was used to develop Robonaut which includes touch sensors at its finger-tips, full neck travel range, high-resolution camera, and Infra-Red systems, advanced finger and thumb movement, etc.

Navigation on the Moon – Deep Learning Planetary Navigation

- The **NASA Frontier Development Lab** is working on a project to provide navigation on the surface of celestial bodies including the moon! This project basically aims to provide GPS even on the lunar surface, just without using multiple very expensive satellites!

- This is done by feeding a Machine Learning System lot's of images of the moon (2.4 million in this case which luckily NASA already has!) and then creating a virtual version of the moon using neural networks.

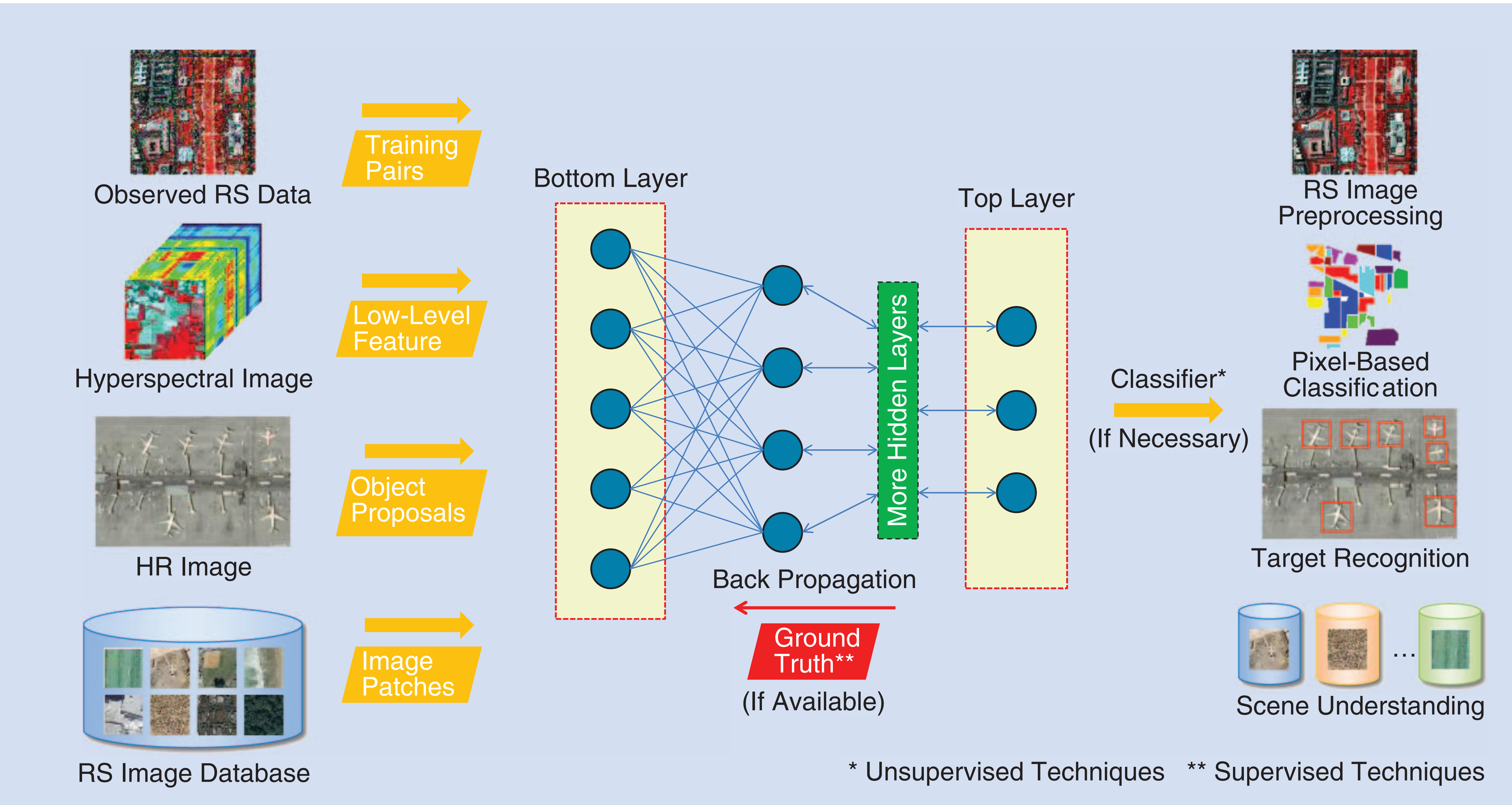
- Then if you are lost on the Moon, you can take images of your surroundings and the Machine Learning System will be able to triangulate your location on the moon by comparing your images with the already created image database of the lunar surface that constitutes the virtual moon.

Data Intensive Technology Development (DITD)

- machine learning and data mining capability for data analytics can be utilized to do following
  - Rapid Anomaly Detection for the Non-Destructive Evaluation of Composite Materials this can assist in rapidly detecting different failure modes for better design of material compositions and structures
  - Pilot Cognitive-State Assessment: The goal could be to predict aircraft pilots' cognitive state using physiological data from flight simulations while performing tasks during various alertness states to help improve pilot training and safety;
  - Enhanced Launch Vehicle Designs: The goal is to develop a framework in which to apply machine learning algorithms for improved design of space launch vehicles using data from modeling and simulation codes/programs
  - Atmospheric and Earth Science Data Analysis: The goal is to apply machine learning algorithms for rapid and enhanced data fusion and analyses, leading to better climate modeling, new insights and better science.

source; Machine Learning Technologies and Their Applications for Science and Engineering Domains Workshop NASA

Deep learning in remote sensing



AI helps in cleaning space debris

Trash truck formation can help the orbit of Earth to get clean

05 July, 2019
by Pratyaksha Dhall

In 2018, it was reported a 19 year old, amber yang, a space enthusiast helped NASA detect and protect the astronauts form the junk floating around in the space. She when she was 15 had an interest in astrophysics. She in winter break learned the coding and decoding of ins and outs astrophysics, coding, and space junk. She developed a product called Seer Tracking which used AI to track debris in the orbits.

A plan has been designed to build a robotic orbital trash truck to clean the debris form the orbits around space. This is a collaborative plan of Stanford's Space Rendezvous Lab and the European Space Agency. The plan starts with an AI navigation system to guide the the trash truck to the debris pieces but that limits only the Earth's orbit.

Computer vision and Machine Learning in Rocket Landing

SpaceX used a convex optimization algorithm to determine the best way to land the rocket, with real-time computer vision data aiding route prediction

Recent research in landing spacecraft has focused on developing algorithms that increase the level of autonomy for air and space systems. Some of the major issues for spaceship or rocket landings include vacuum stage, software errors, guidance and sensor problems etc. Machine learning and computer vision are the core optimization and evaluation techniques for successful landings.

COMPUTER VISION VIRTUAL REALITY APPLICATIONS FOR SPACE AND ENDEAVORS

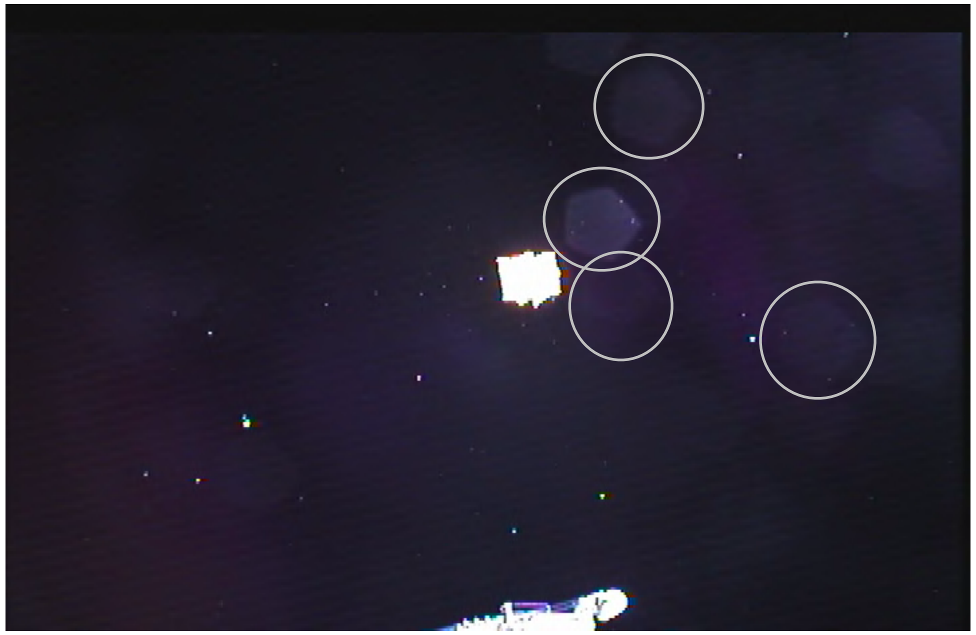
ACKNOWLEDGEMENT :
TO ALL MY BTECH, MTECH, PHD AND RESEARCH FELLOWS

INVESTIGATION OF PSLV C36 VIS SATELLITE SEPARATION IMAGES

WHAT IS THE QUESTION?

INFERENCE

- We have taken 4 most significant polygons and calculated the avg. intensity for 274 frames.



NOTE:

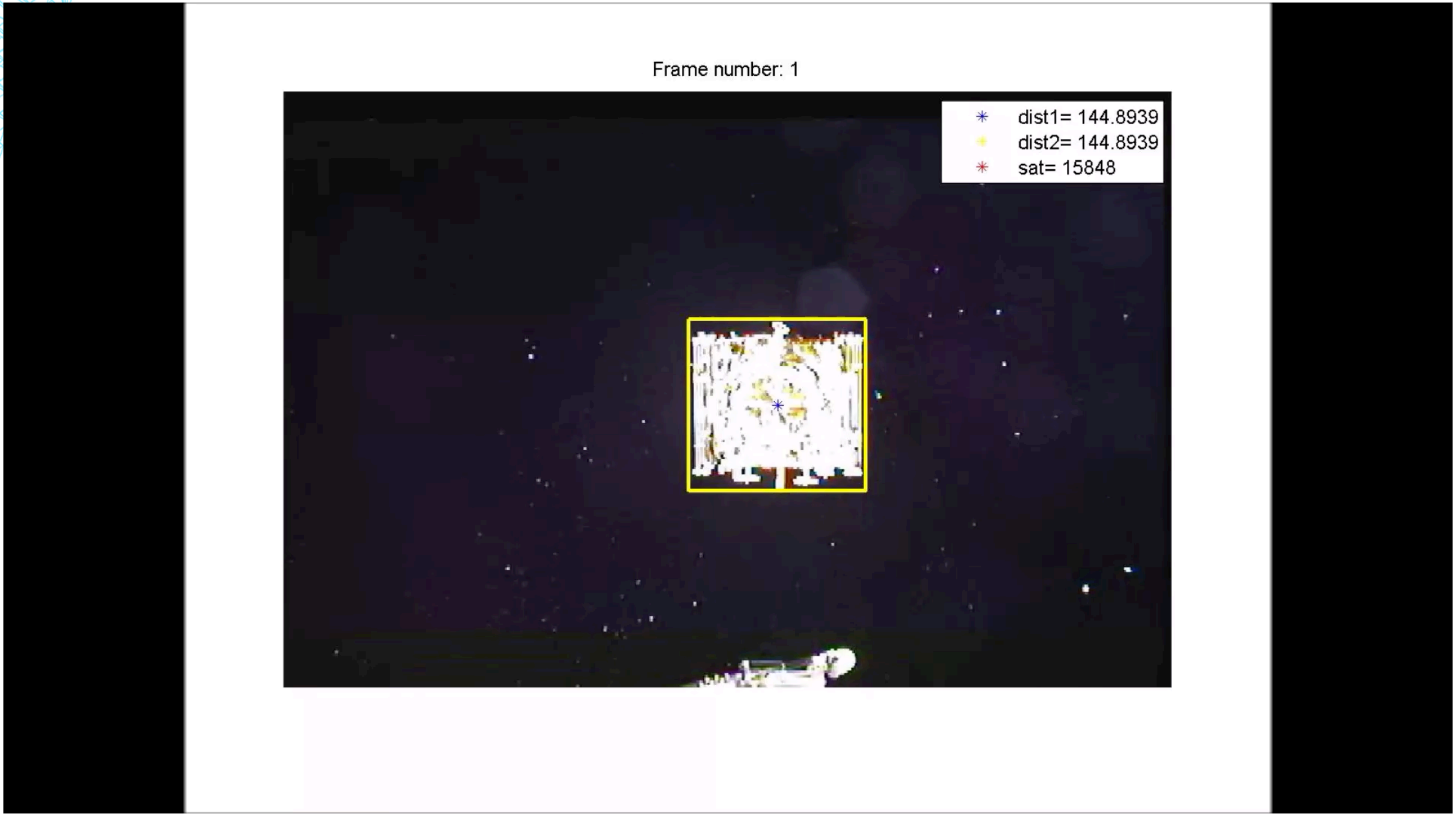
• Polygons are static across all frames with relative motion correction.

• The avg. intensity is **a function of the reflections from other objects.**

• These objects have a perfect **Geometric shape**, no real object in space has this.

- For all frames the position of the polygons does not change.

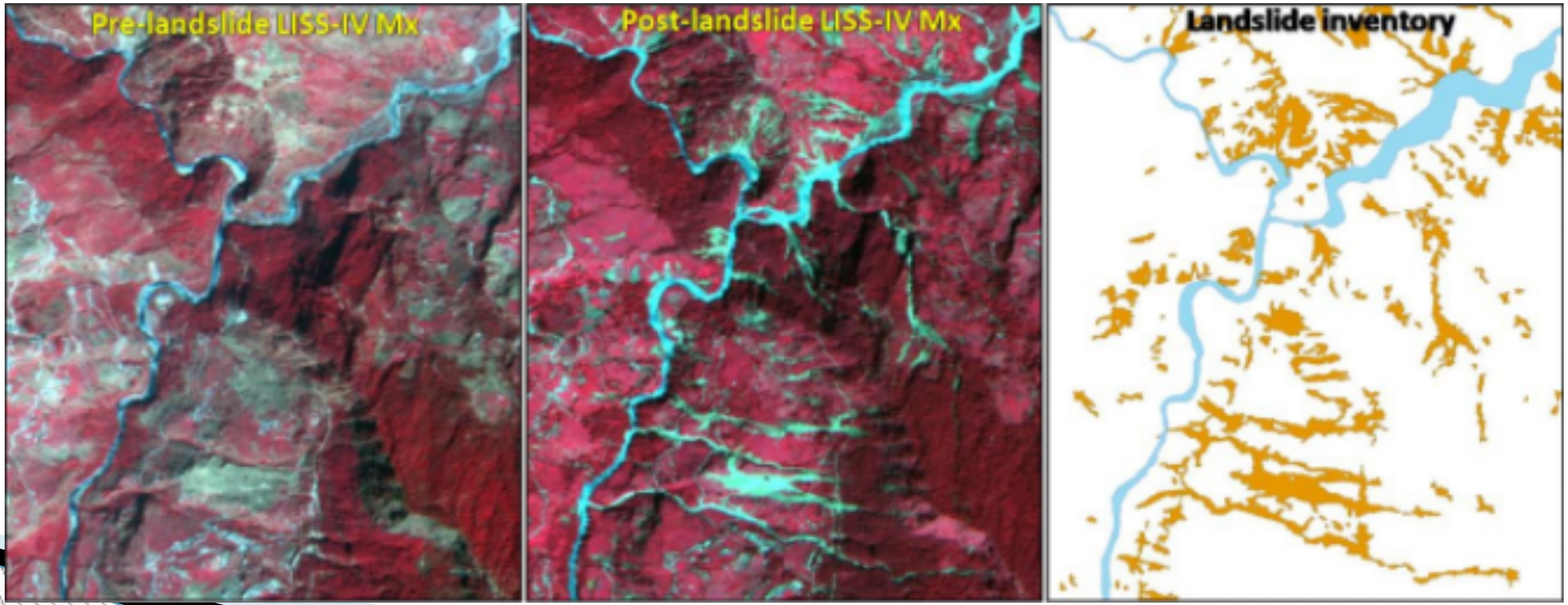
TRAJECTORY OF THE MAJOR OBJECTS



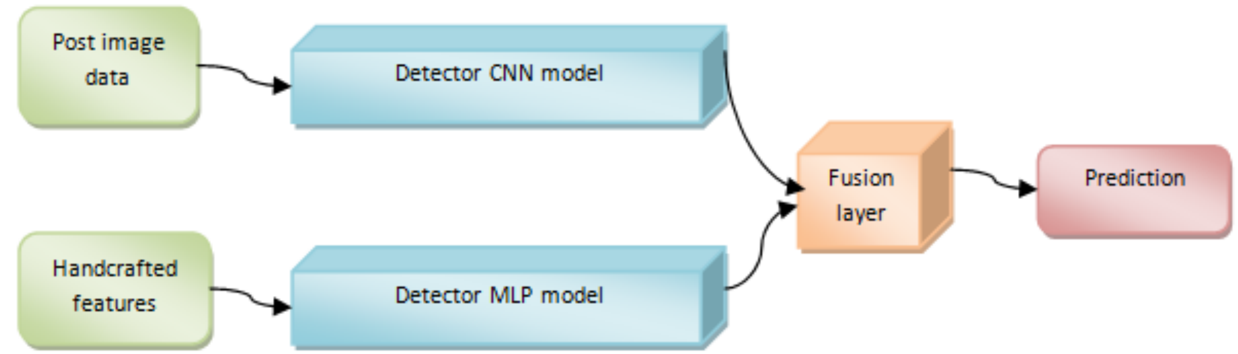
High resolution multispectral image analysis for landslide identification

IIST – NRSC collaborative project IIST/RDP/02/2015

- Objective:- Automatic detection of landslides from multispectral satellite images using python open source tools.
- Study area:- Mizoram and Assam (83D15 and 83D16)
- 3 band (green, red and NIR) multispectral images from Resourcesat 2 taken with a high resolution sensor Linear Imaging Self-scanning System IV (LISS IV).

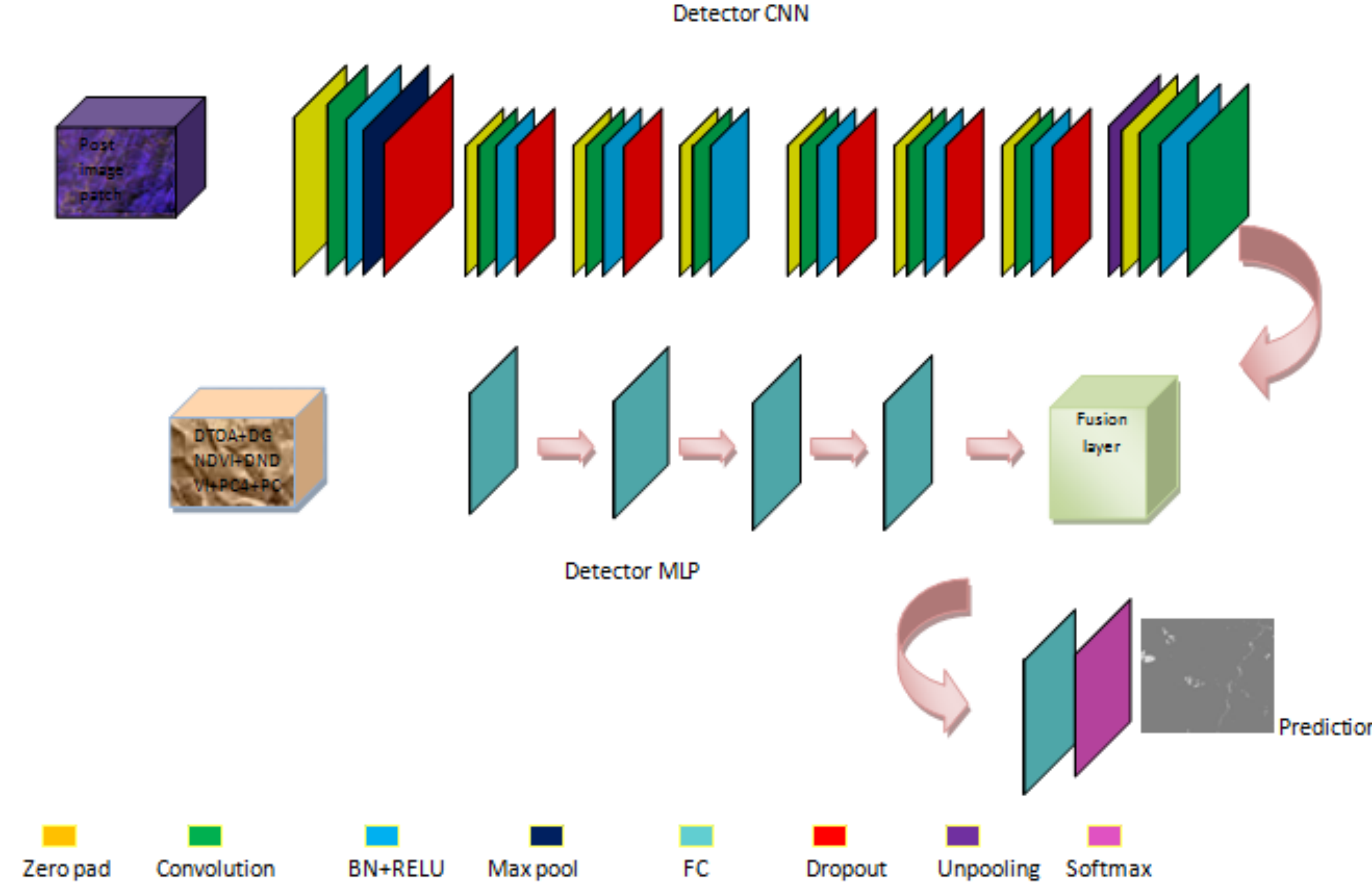


LsD-n model

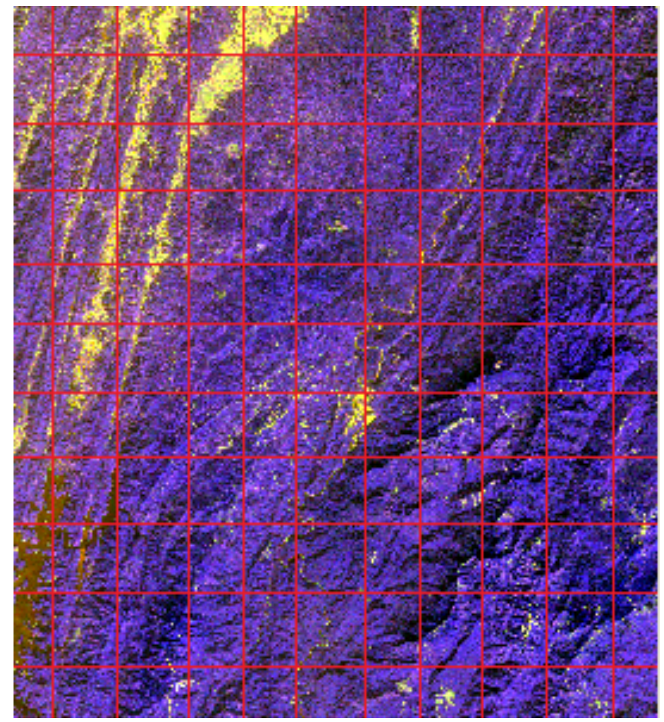


LsD-n model

- Derived from SegNet basic architectur
- Input image size 16x16, Built using Keras deep learning library
- 2 class (landslide & non- landslide) pixel-wise classification
- 483 training , 104 validating and 104 testing images
- Trained on Titan Xp 12 GB GPU with cuDNN v5.1 acceleration
- Total params: 2,510,415



Dataset Preparation



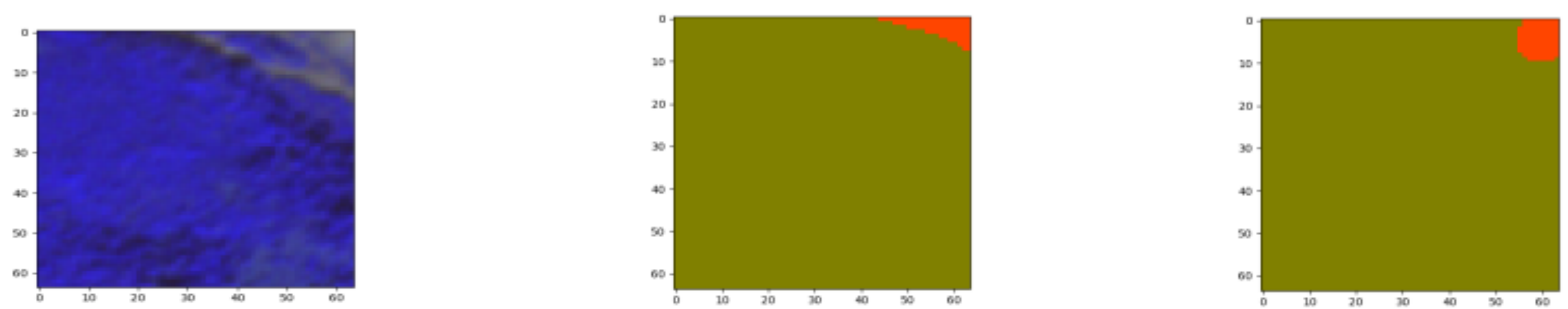
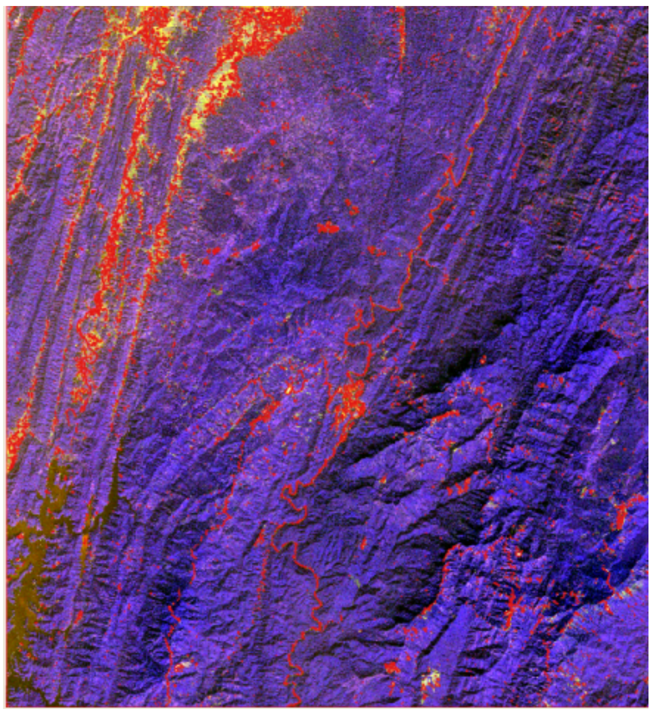


Fig: a)Input image patch b)Ground truth c) Prediction on landslide and non- landslide class

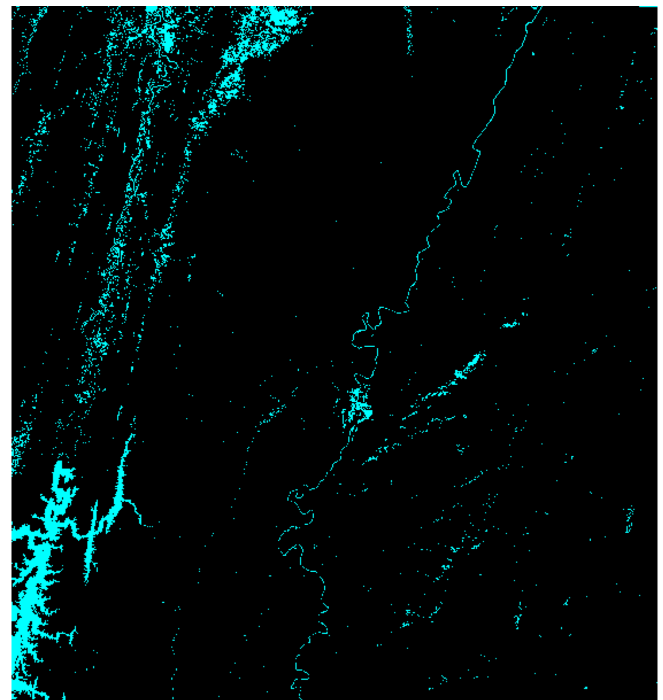
Results



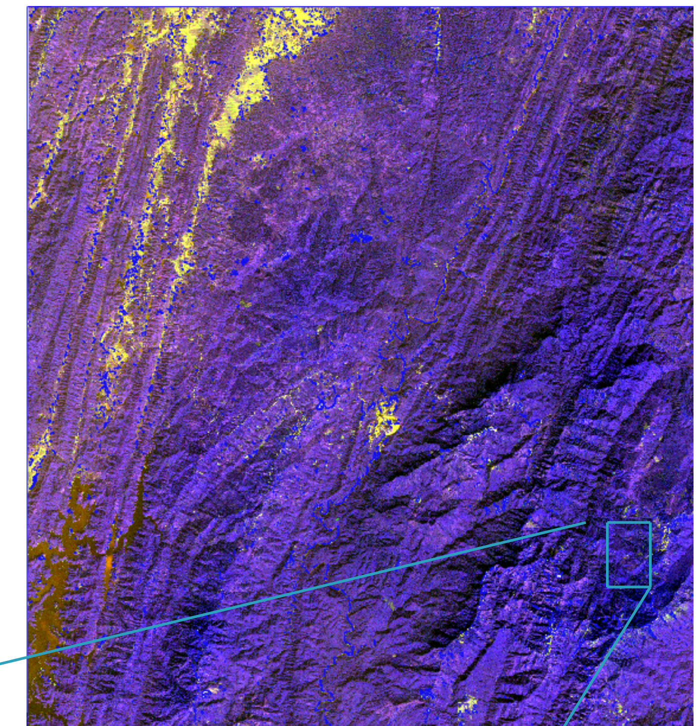
LsD-n prediction

-

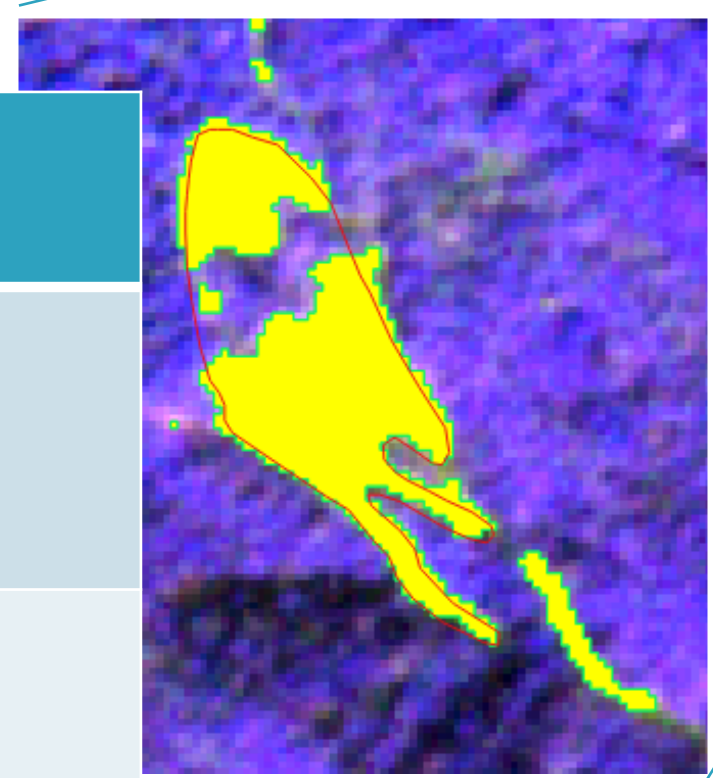
NDVI feature mask



=



After post processing



Enlarged view

Dataset	83D15	83D16
Total no. of landslides	429	134
Detected landslides	404	117

Star Cluster Identification

Using Classical and Deep Unsupervised Learning

Arnab Karmakar[1] D. Mishra[1] A. Tej[2]

[1]Department of Avionics
Indian Institute of Space Science and Technology

[2]Department of Earth and Space Science
Indian Institute of Space Science and Technology

What is a Star Cluster?

- By definition, star clusters are a group of stars that share a common origin and are gravitationally bound for some length of time.
- There are two basic categories of star clusters: Globular and Open (aka. Galactic) star clusters.

- Detecting star clusters is difficult due to the presence of dense molecular clouds, unrelated background objects and instrument noise.
- Existing methods use primitive data processing methods like Source Counts, k-Nearest Neighbour, Minimum Spanning Trees etc.
- Experiments with advanced machine learning techniques is required to discover higher level features and has a high scope of improving accuracy.
- Advanced data processing and machine learning techniques are still not very well explored by the astronomy community due to lack of structured dataset.

Problem Statement

Given a patch of the sky, we would like to detect whether there is a star cluster present in that patch of the stellar image (and data), and if so, we would like to detect it with precision and predict some of the properties of that star cluster.

Therefore Objectives of our work are,

- detection and localisation of star cluster in a patch of sky
- predict specific astronomical properties of that cluster
- exploring the possibility of advanced data processing algorithms to extract high level abstraction of features in star clusters

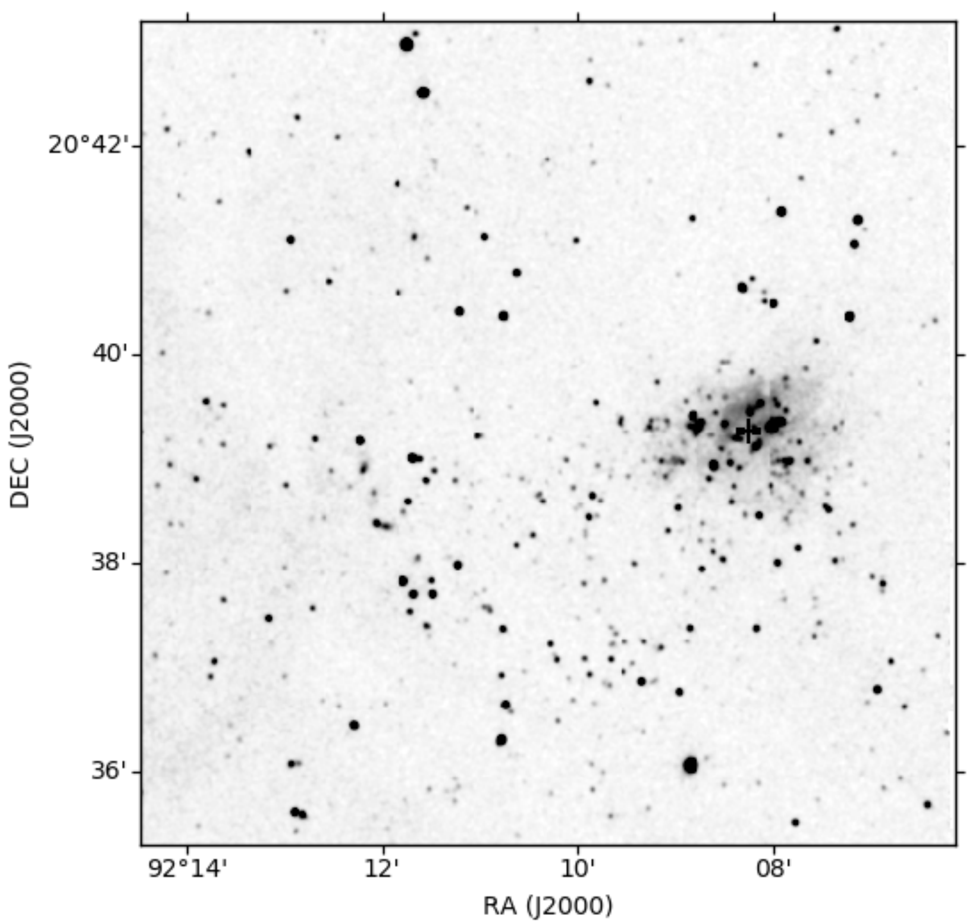
Data Collection

- **Near Infrared Data from 2MASS**
  NIR (K_s band, $2.17\mu m$) data for point sources as well as images around given cluster position is obtained from the Two Micron All Sky Survey (2MASS) Point Source Catalog (PSC) and 2MASS image service archive respectively.

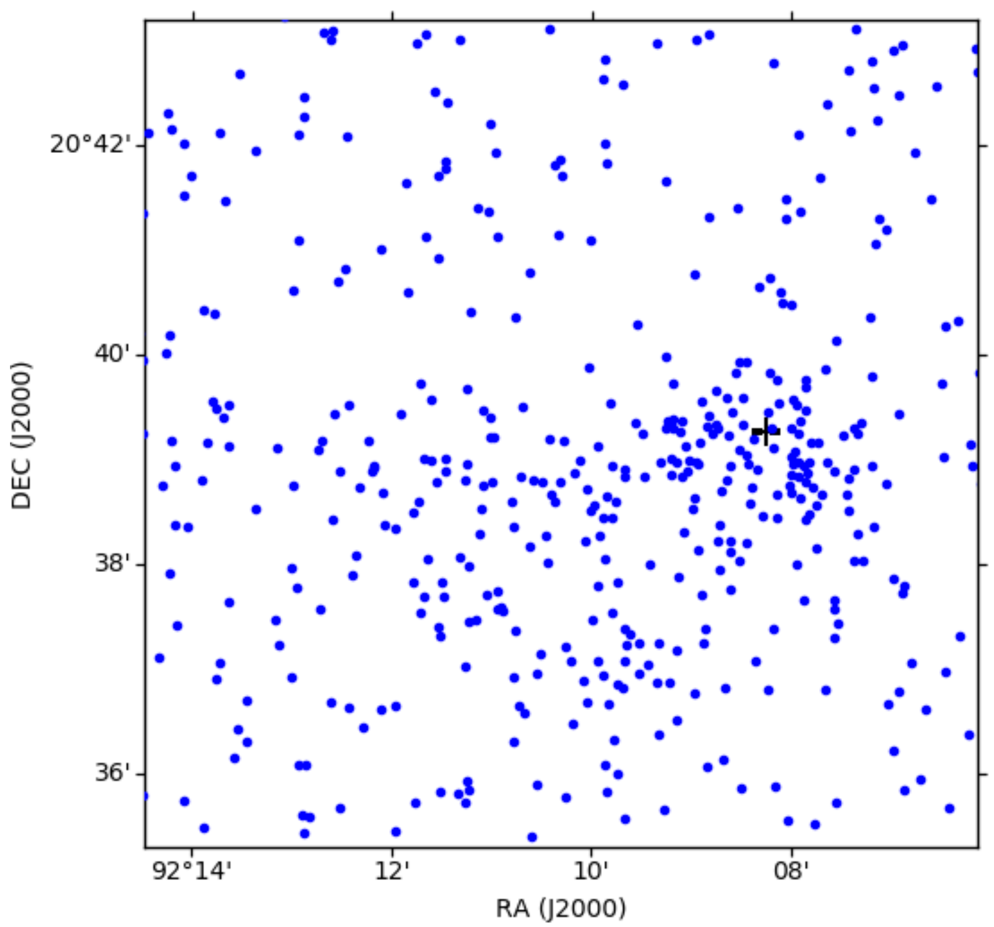
- **Near Infrared Data from UKIDSS**
  K-band ($2.20\mu m$) data from UK Infrared Telescope (UKIRT) Infrared Deep Sky Survey (UKIDSS) 10PLUS Galactic Plane Survey (GPS) archive.

Data Collection

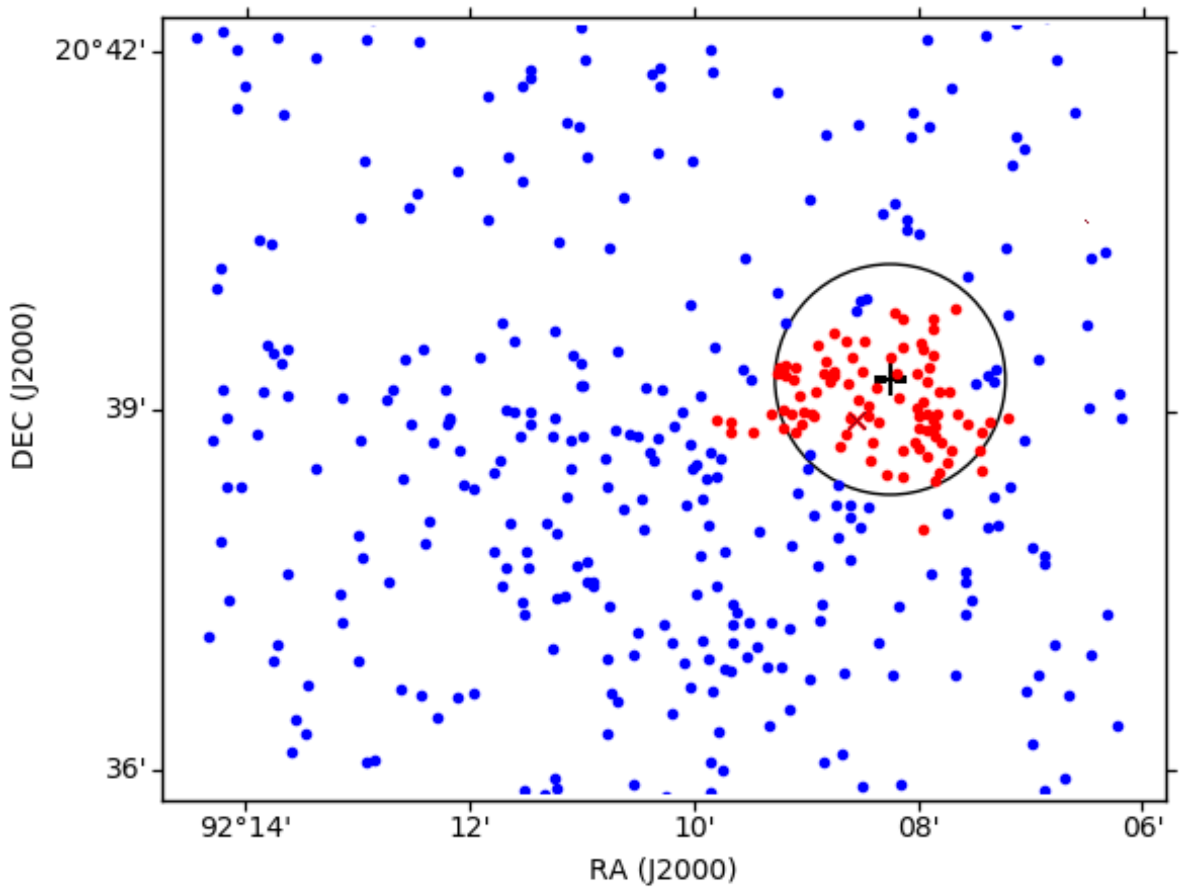
Image Data

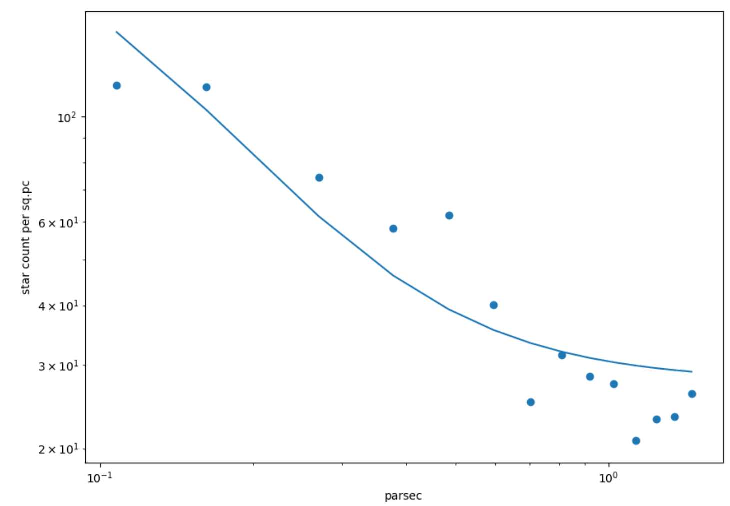


Catalogue Data

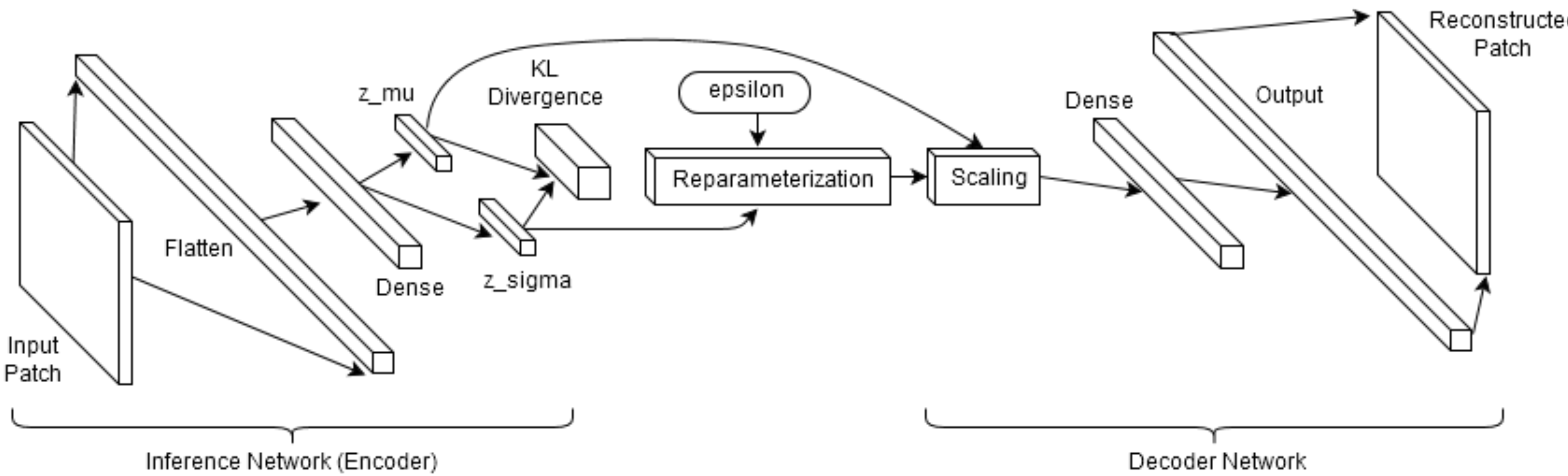


Mean shift algorithm





Deep Variational Autoencoder (DeepVAE) Model



Deep Variational Autoencoder (DeepVAE) Model

Reconstruction

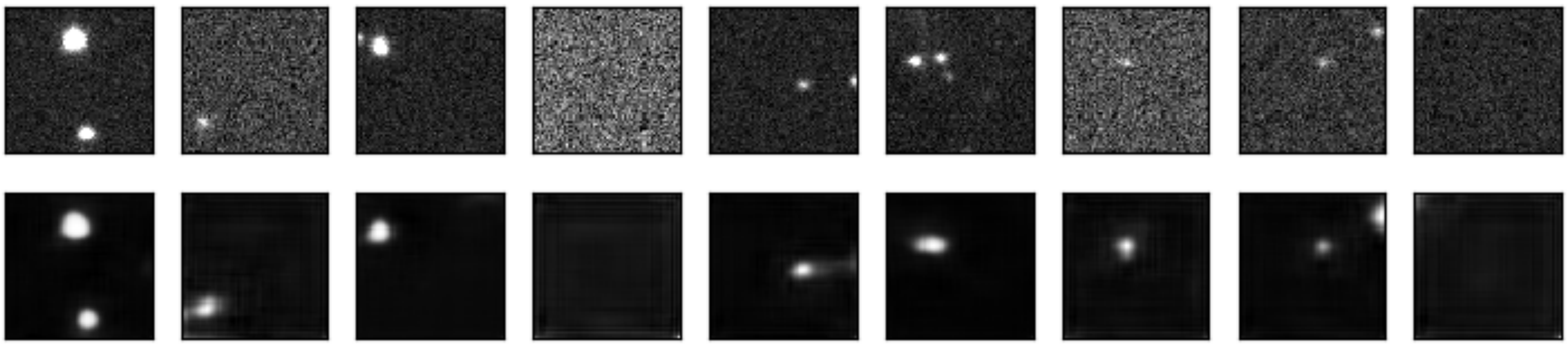


Figure: The reconstructed patches of our DeepVAE model

- It shows that our model performs significant noise removal distinguishing the foreground object from the background noise.

GMM Clustering Results

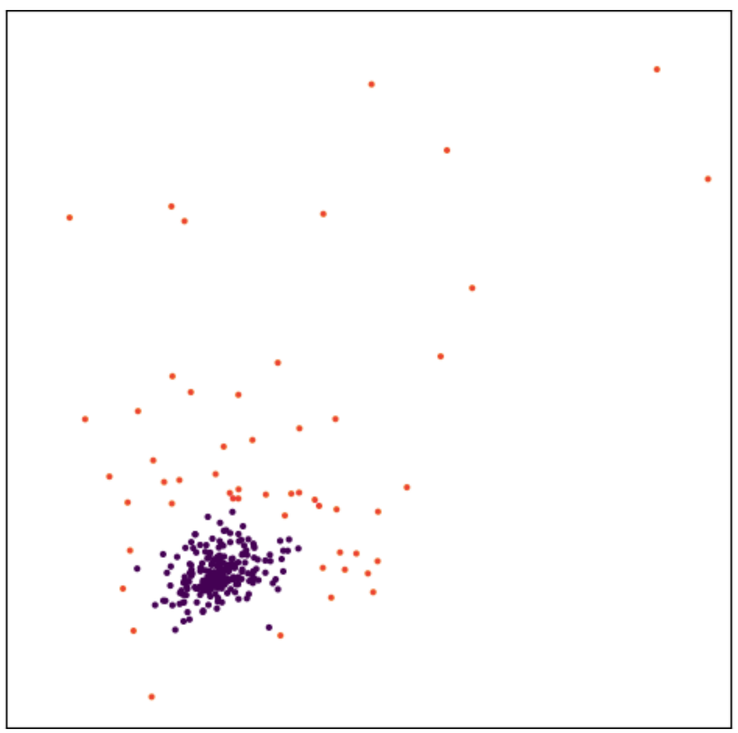


Figure: Visualization of latent embeddings z as classified by GMM for 64 $\times$ 64 patch size

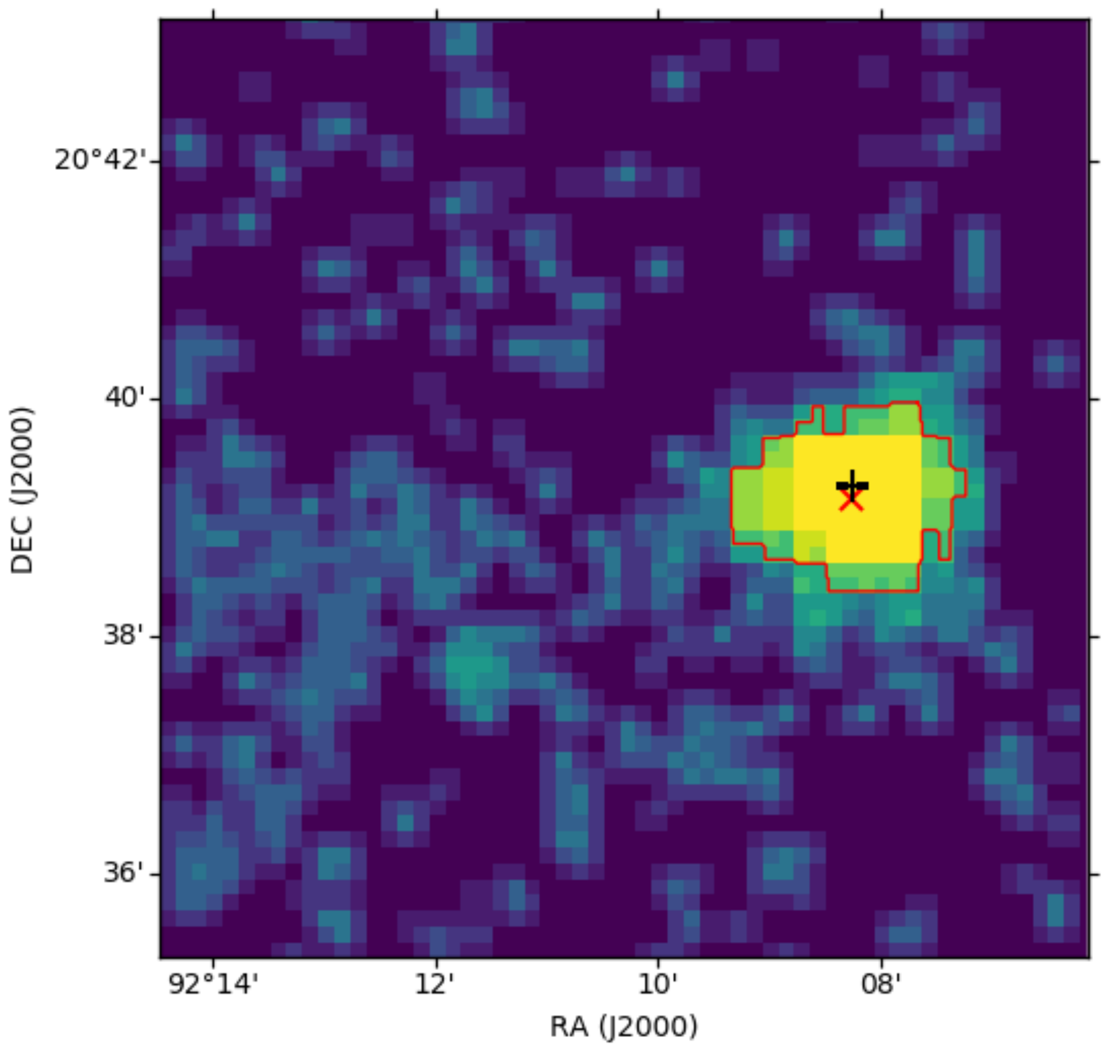
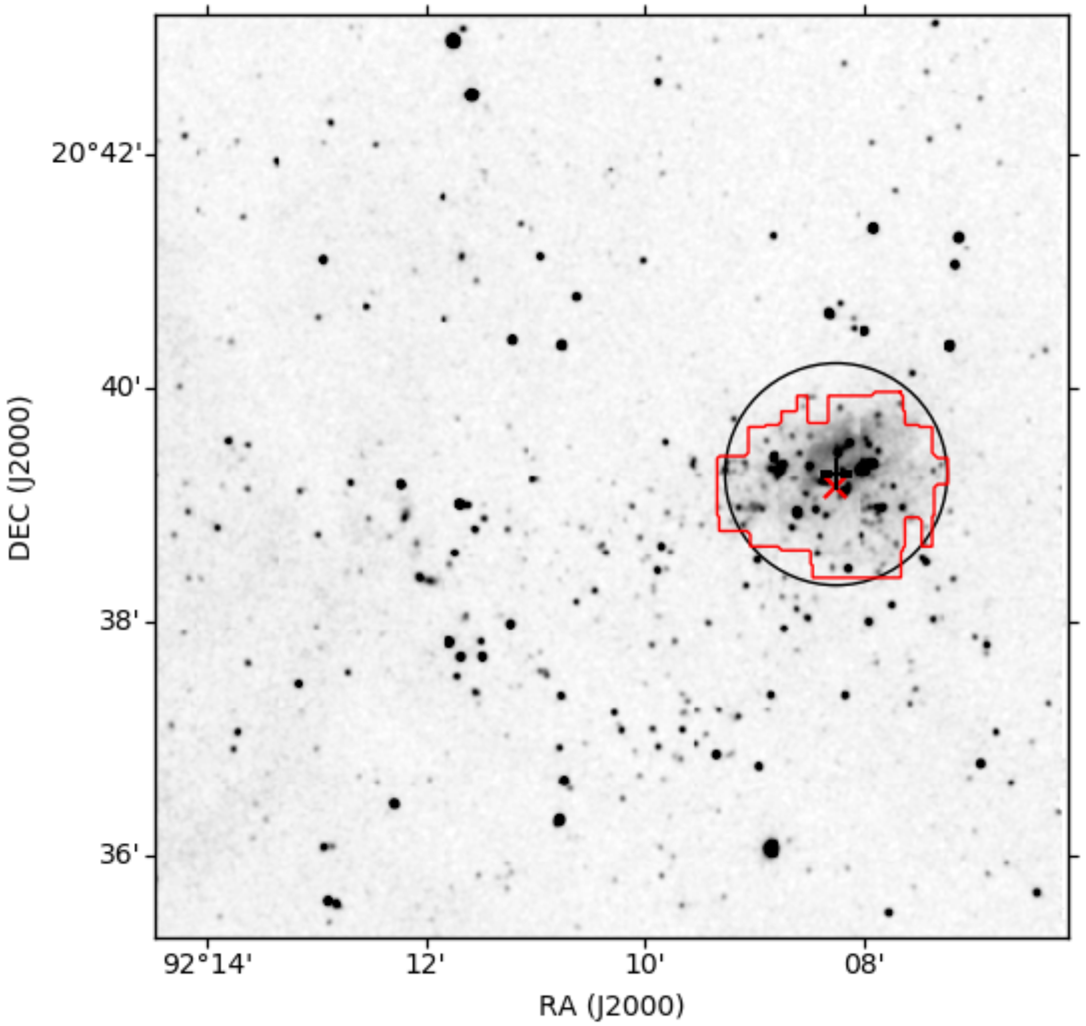


Figure: The heatmap plot combining detections from all 4 different patch sizes

Comparison of Results



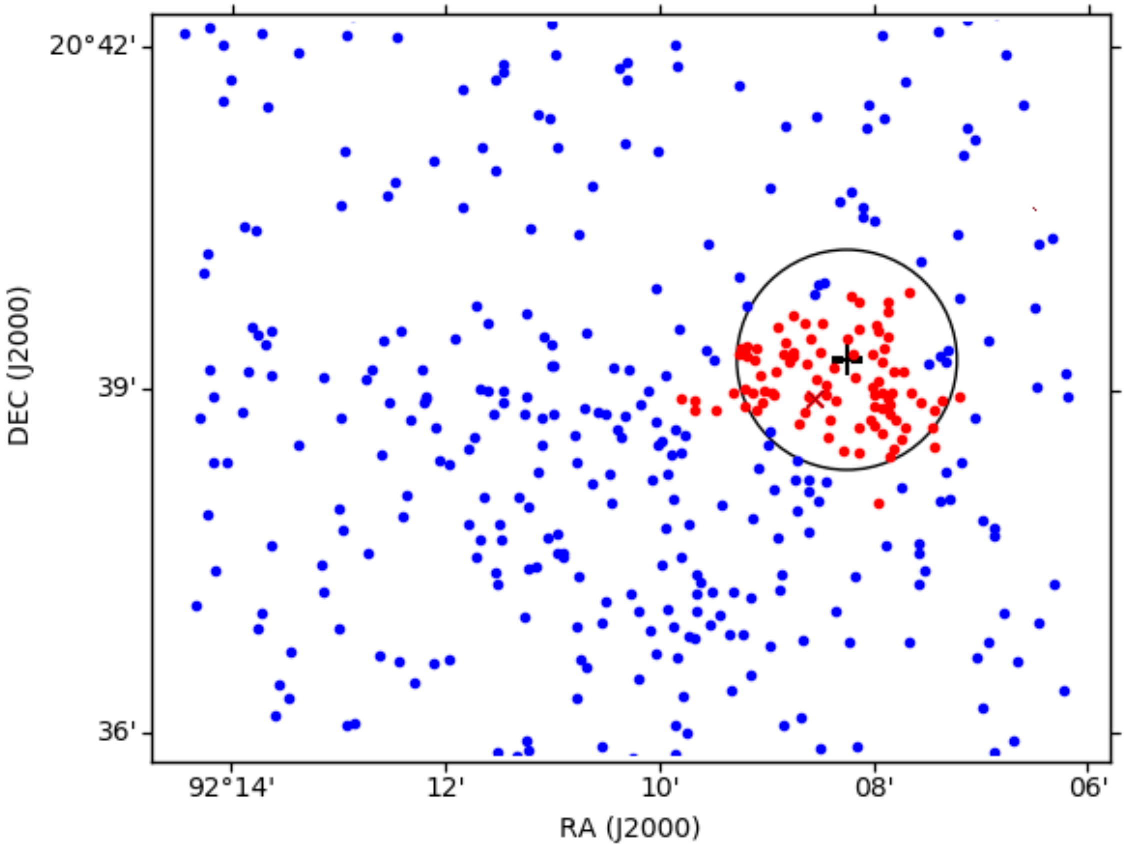


Table: Detected cluster center position and its deviation from IRAS point source, as well as deviation between centers detected by both the methods

IRAS name	IRAS source position		Detected center position (deg)				Deviation (") from IRAS source		Deviation(") catalogue to image
			catalogue		image				
	RA	DEC	RA	DEC	RA	DEC	catalogue	image	
06055+2039	92.1375	20.654	92.1426	20.64886	92.1381	20.6522	13.58	**4.15**	10.15
05274+3345	82.6917	33.799	82.7047	33.79584	82.6953	33.7877	23.97	21.11	22.28
05345+3157	84.45	31.99	84.4487	31.98343	84.4538	31.9826	12.05	14.94	**9.33**
05358+3543	84.7917	35.755	84.7829	35.75929	84.7885	35.7579	17.39	7.32	10.47
05490+2658	88.0542	26.993	88.0577	26.98818	88.0497	26.9947	10.03	9.02	18.62
05553+1631	89.5583	16.533	89.5498	16.53138	89.5703	16.5439	15.63	28.77	43.15
06056+2131	92.1708	21.517	92.1785	21.51992	92.1665	21.526	14.99	18.40	24.15
06061+2151	92.2833	21.844	92.2804	21.8471	92.2745	21.8436	**7.75**	15.85	12.40

Table: Parameters of the star cluster as predicted by both the models

IRAS name	Members			Radius (")			IoU (%)	
	Catalogue	Image	SOTA	Catalogue	Image	SOTA	Catalogue	Image
06055+2039	93	98	119	81.3	77.13	89	91.46	**85.95**
05274+3345	63	56	48	118	89.86	103	**93.63**	79.68
05345+3157	105	98	95	136	101.63	126	81.26	83.27
05358+3543	72	71	53	142	116.74	130	78.69	74.32
05490+2658	89	93	95	121	98.46	122	89.92	82.24
05553+1631	68	96	80	115	123.59	104	76.49	71.43
06056+2131	117	101	132	121	104.79	132	83.42	78.23
06061+2151	114	102	105	159	116.05	145	75.79	69.98

Virtual Reality in Disaster Simulation

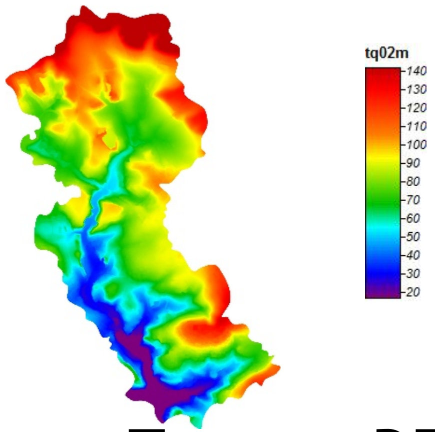
(IIST-NRSC Collaborative project)

The Objective involves

- Study of different open source flood water models, their mathematical modeling, critical analysis etc.
- Selection of apt mathematical model, DEM, soil profile, infiltration model for analysis.
- Flood inundation via a dam outburst or overflow of rivers or rainfall or a combination of all these
- Observing how the flood will spread through the terrain considering the terrain characteristics
- 3D visualization
- Rendering based on Colours
- Implementation of Virtual Reality

Cellular Automata(CADDIES)

- ✓ Cellular automata are computational methods based on the simulation of a discrete space in which a set of universal laws apply
- ✓ The model uses a weight-based system to minimize the use of complex equations like Shallow Water Equations.
- ✓ The parallel model implementation runs on multi-core CPUs and graphics card GPUs.
- ✓ The volume of water transferred between the central cell and the neighbour cells is limited by the Manning's formula
- ✓ The Mannings equation is an empirical equation that applies to uniform flow in open channels and is a function of the channel velocity, flow area and channel slope.
- ✓ Building of CADDIES has been successfully done using CMAKE 3.1 and Visual Studio



Torquay DEM

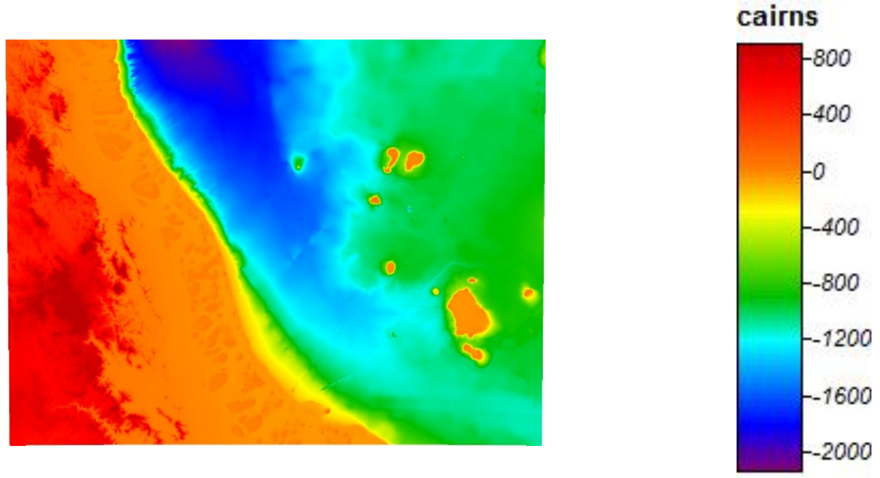
<u>Real World Test-Case(Torquay)</u>

- The test case consists of a steep slope in the upper catchment and gentle slope in the lower catchment
- A design rainfall of 40mm/h was applied to the whole area for one hour
- The catchment was set to be 100% impervious
- The modelling of the drainage system was not included since this is not implemented in the WCA2D model
- The full simulation time was 12 h.
- The terrain elevation ranges from 0 to 180 m The terrain does not contain any rivers or channels and the main slope direction is from north to south where the Torquay harbour is located
- A constant Manning roughness of 0.015 $(m^{-1/3}s)$ was applied to the whole area

[redacted]

ANUGA

✓ Developed by Australian National University and GeoScience Australia
✓ Based on full dynamic shallow water equations
✓ Frictional resistance implemented using Manning's formula
✓ Finite volume analysis on triangular mesh has been used.
✓ Operating language is Python.

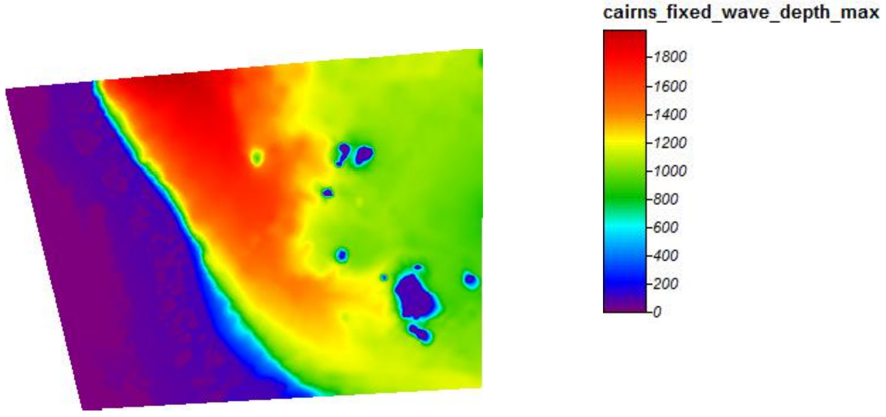


Cairns DEM

Real World Test-Case(Cairns)

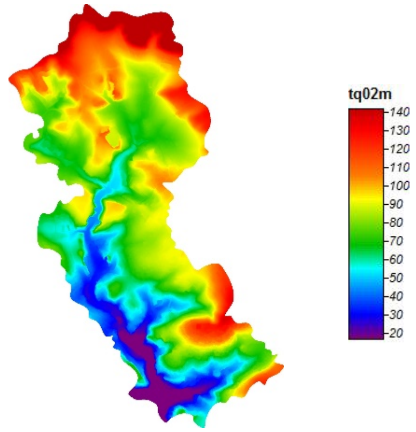
- A hypothetical scenario using real-life data has been simulated.
- A huge 50m wave starting after 60 seconds and lasting 1 hour has been simulated.
- No infiltration has been considered

DEM showing the maximum water depth after simulation



LISFLOOD-FP

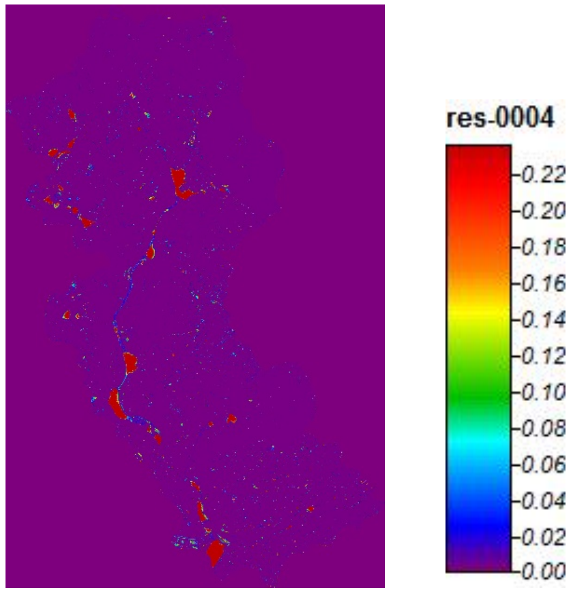
- ✓ A two-dimensional hydrodynamic model specifically designed to simulate floodplain inundation
- ✓ Capable of simulating grids up to 10^6 cells for dynamic flood events
- ✓ Mathematical framework is based on the inertial approximation(neglects convective acceleration) of shallow water equations
- ✓ Solution is implemented using Finite difference method on staggered grid



Torquay DEM

Real World Test-Case(Torquay)

- The test case consists of a steep slope in the upper catchment and gentle slope in the lower catchment
- A design rainfall of 40mm/h was applied to the whole area for one hour
- The catchment was set to be 100% impervious
- The full simulation time was 12 h.
- The terrain elevation ranges from 0 to 180 m .
- A constant Manning roughness of 0.06 $(m^{-1/3}s)$ was applied to the whole area



Torquay water depth after 40000s.

Satellite Image Classification using Deep Learning

- **Satellite Data Specification:**
  - Sentinet-1 sat is at about 680km above earth
  - Equivalent to RISTSAT(an Indian Remote Sensing Satellite)
  - Pings H polarized signals and receives H & V polarized back scattered signals, namely HH & HV
  - The back scatter coefficient is given by,
    - $\sigma_0(dB)=\beta_0(dB)+10\log10[\sin(ip)/\sin(ic)]$

    where,

    1. ip = Incidence angle for a particular pixel
    2. ic = Incidence angle for center of the image
    3. K = constat

Satellite Image Classification using Deep Learning

- **Satellite Data Specification:**
  - σ_o varies with the surface on which the signal is scattered from.
  - For a particular angle of incidence, it varies like:

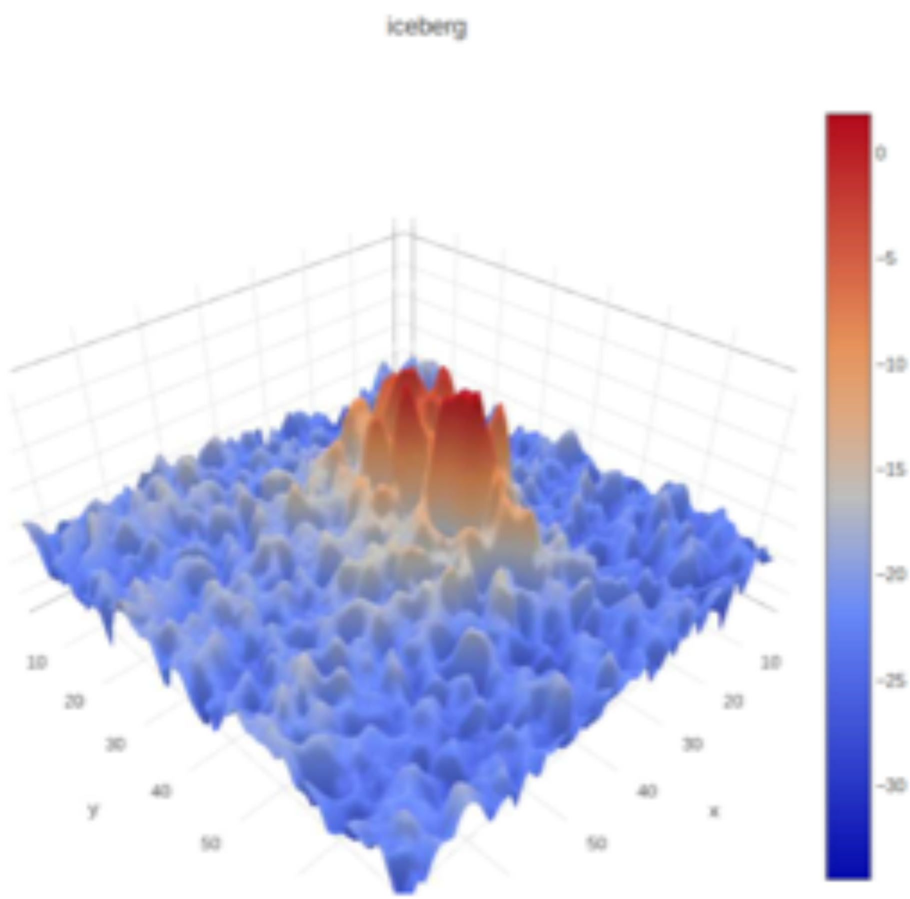
Surface	Water	Settlement	Agriculture	Barren
HH	-27.0010	2.70252	-12.7952	-17.2579
HV	-28.0350	-20.2665	-21.4471	-20.0190

  - Objective is to classify iceberg and ships. However, it can be easily extended to other classes.

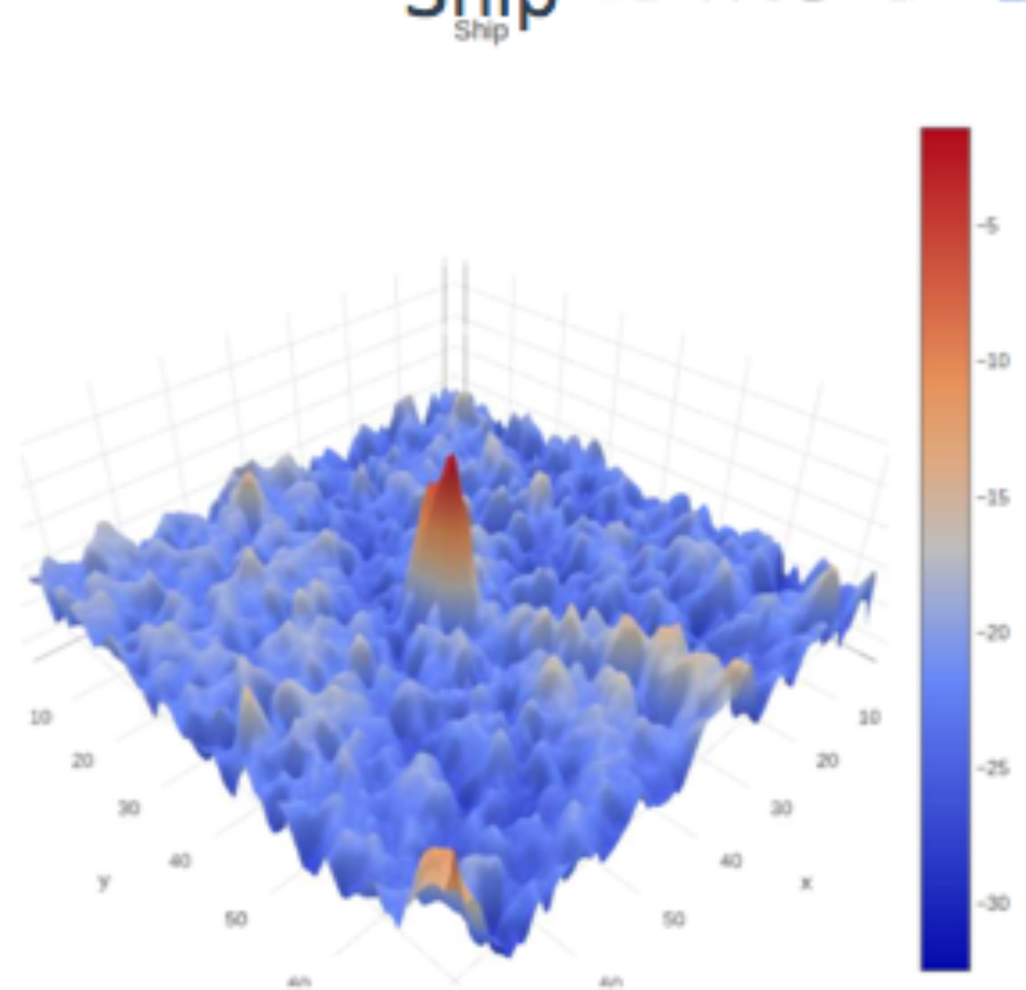
Satellite Image Classification using Deep Learning

- **Visualizing Data**

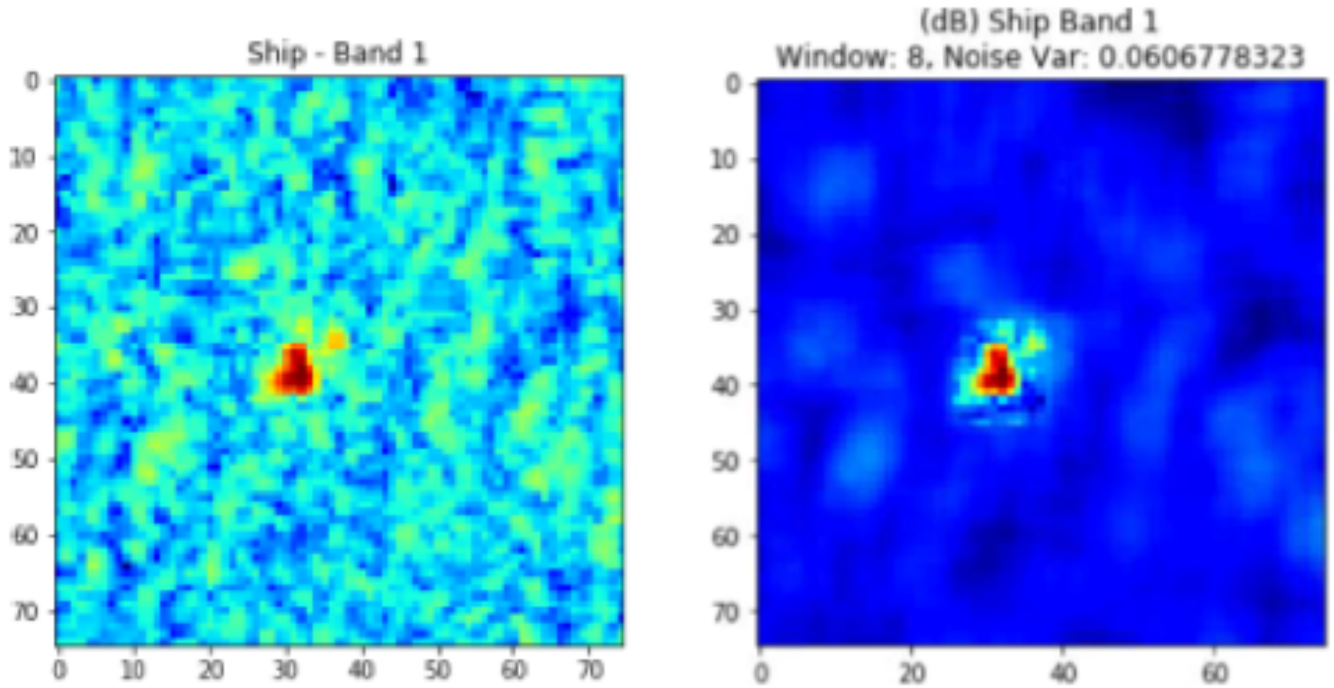
Iceberg



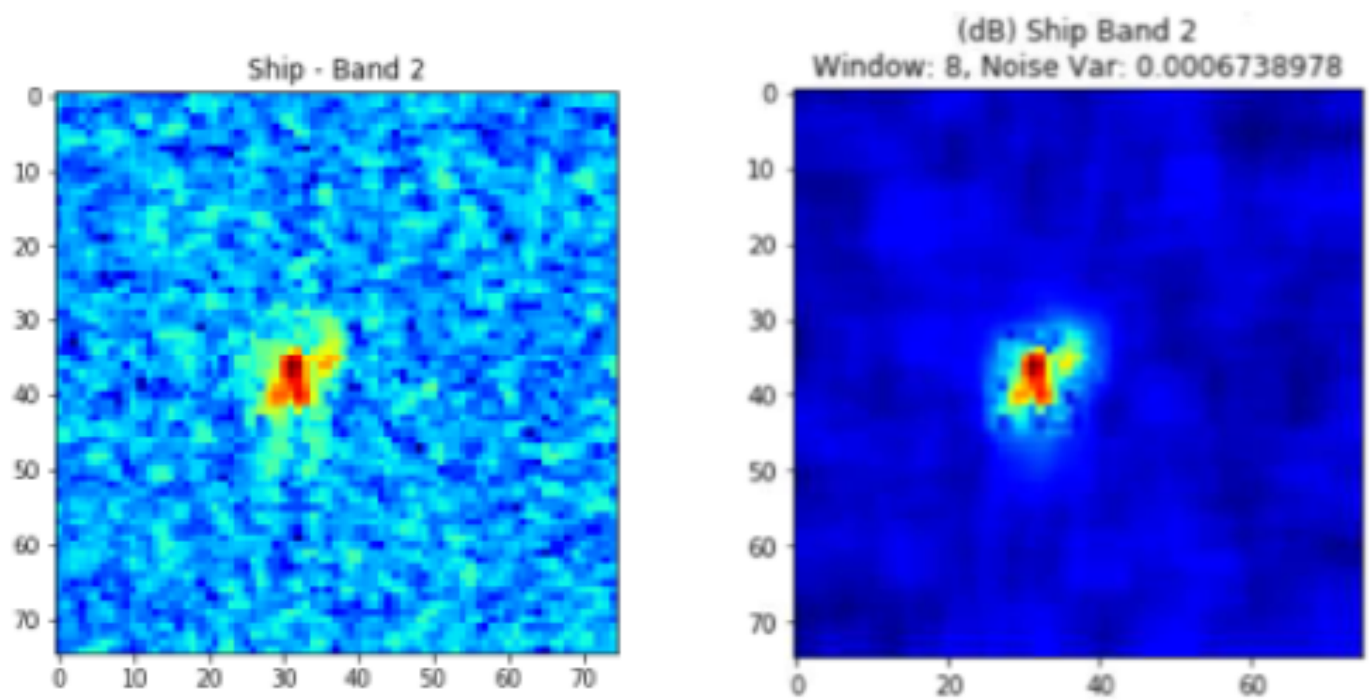
Ship



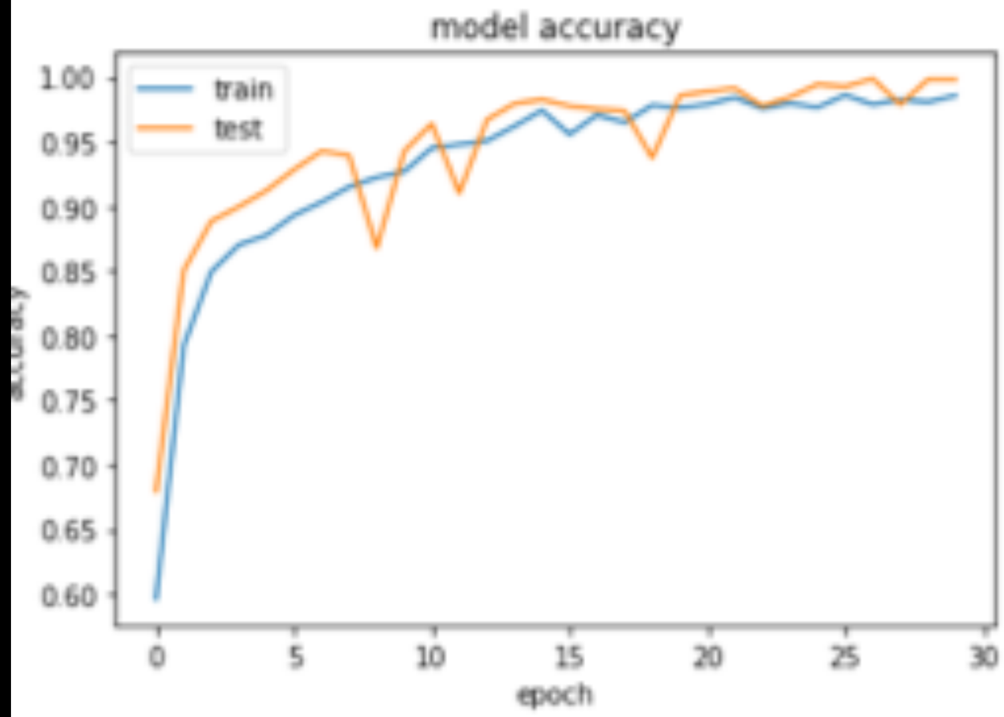
- **Speckle Filter Ship Data: Band1(HH)**

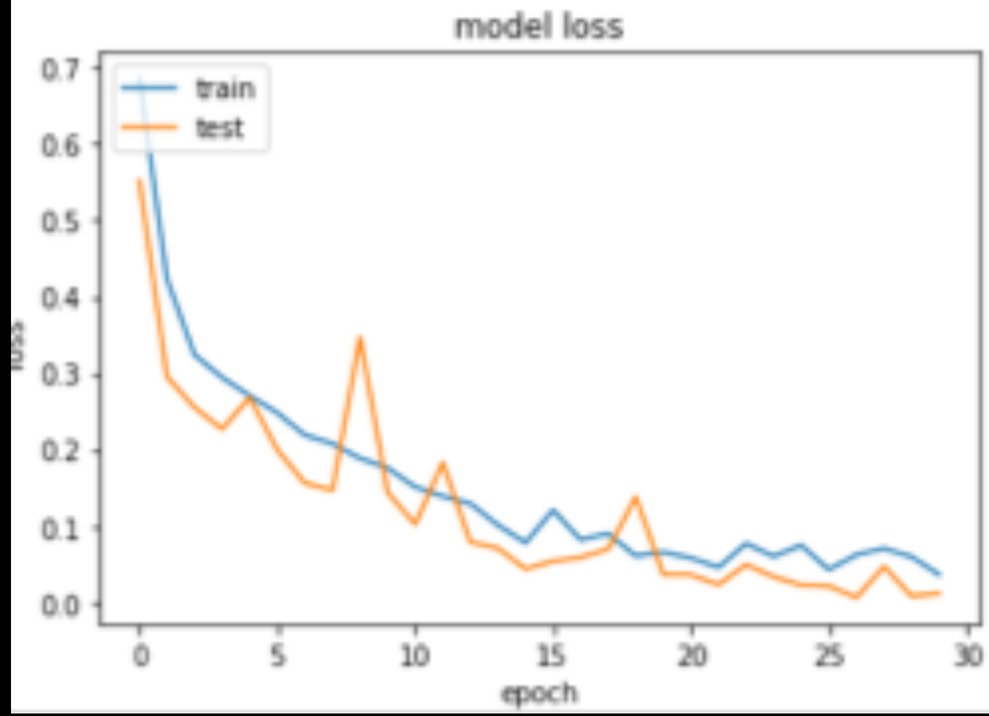


- **Speckle Filter Ship Data: Band2(HV)**



ACCURACY /RESULTS





- **Deep Learning Architecture**

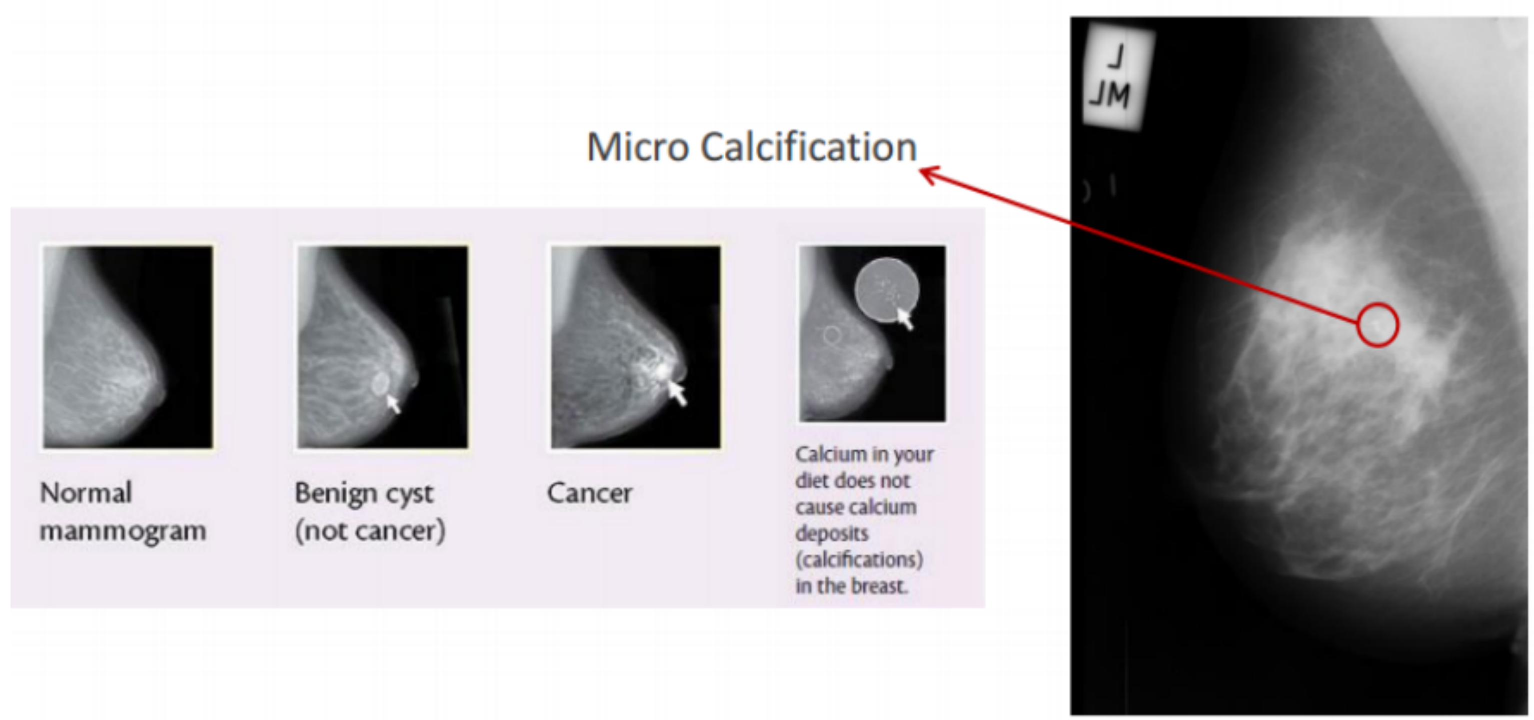
- Training Accuracy : 98.76%
  Log Loss : 0.07

- Testing Accuracy : 98.83%
  Log Loss : 0.06

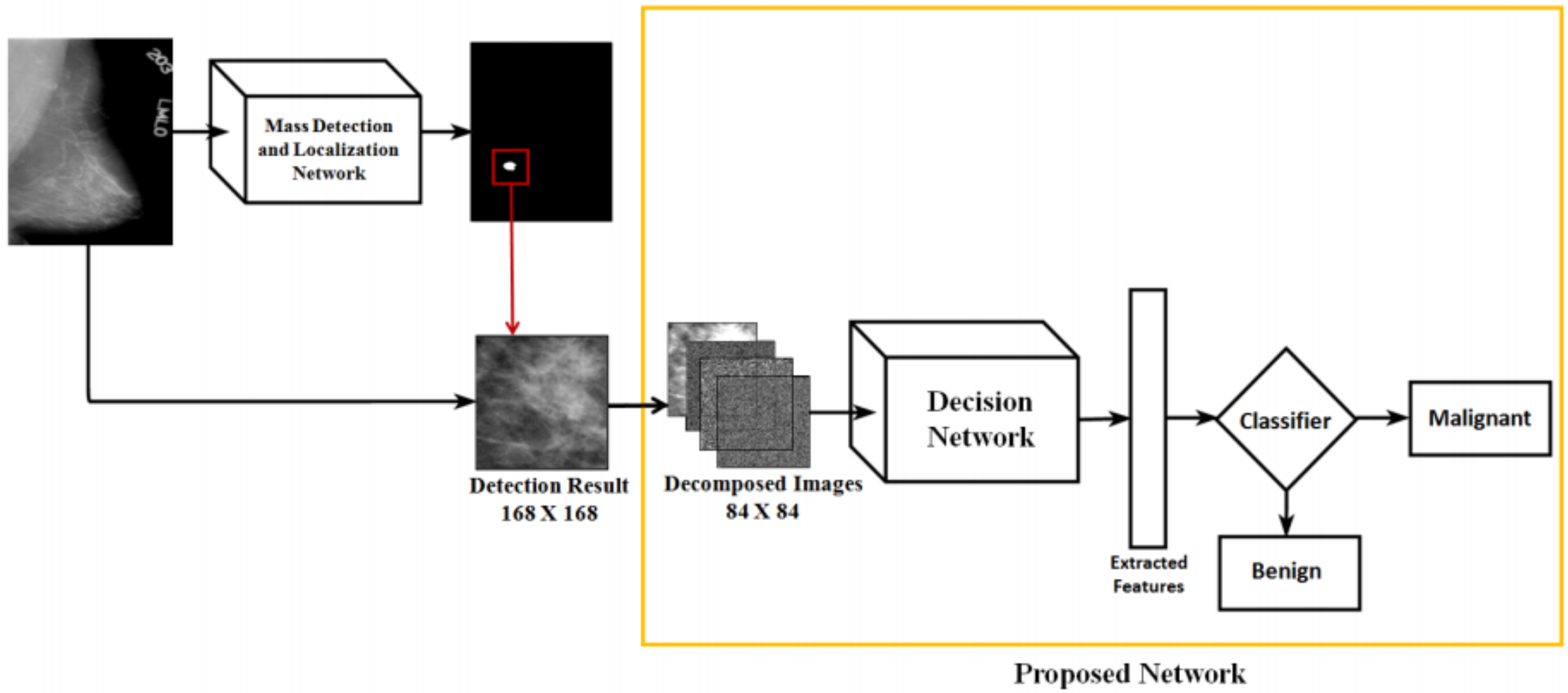
- Total Parameters : 420,837
- Trainable Parameters : 420,791
- Non-trainable Parameters: 46

Deep Learning in Computer Vision and Bio-medical Applications

- Problem Statement:-
- Breast cancer is one of the major causes for the increase in mortality among women, especially in developed and under developed countries
- Due to very small non-palpable micro-calcification clusters/masses, approximately 1% to 20% of breast cancer is missed by radiologists.
- Due to the difficulties of Radiologists to detect micro-calcification clusters a Computer Aided Diagnosis (CAD) system is much needed.
- Objective:-Automatic mass classification from breast mammograms



Architecture of a complete breast mass classification frameworks



- Extraction of Region of Interests:-
- we evaluated our model on pre-segmented ROIs.
- Each ROIs are resized to 168 × 168 to reduce the number of network parameters also we used four rotations rot = {0, 90, 180 , 270 } (in degrees)and increased the training data three times.

Feature Extraction

- Once the ROI is extracted, 2D DWT (Daubechies wavelet) is applied to each ROIs.
- DWT decomposes the mammographic ROI into a number of sub-images in different resolution levels preserving the high and low frequency information
- We train the proposed CNN network with normalized sub-images along with the normalized and approximated ROI as the input.

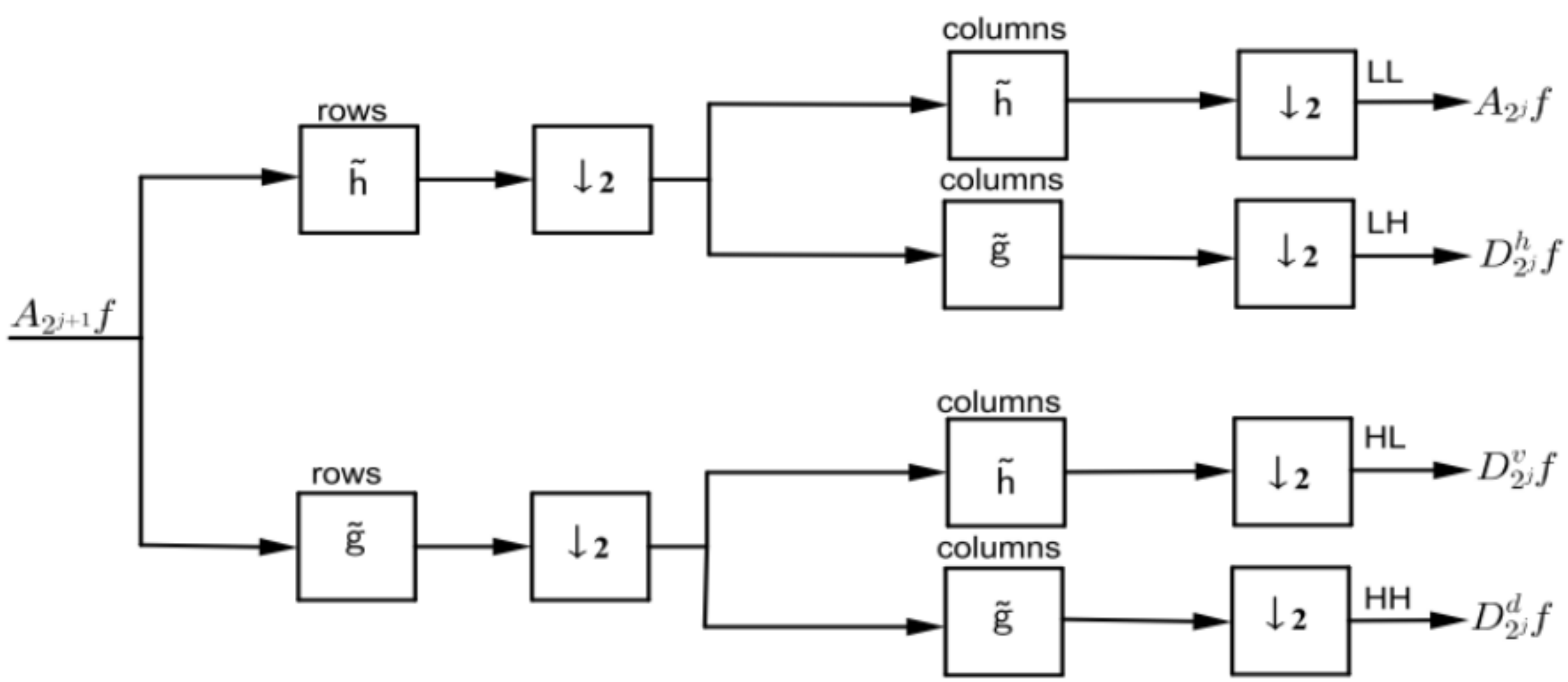


Fig:- Wavelet decomposition using filter banks

Classification

- We have used our trained CNN network as a deep feature extractor and Support Vector Machine (SVM) as the classifier

- The deep feature is a 1000 × 1 dimensional feature vector. We used SVM with Radial Basis Function (RBF) kernel

- A grid search method is employed to find the best hyperparameters w.r.t RBF kernel within a range

Name	Filter size	Depth of Filter	Dropout
Conv1	11	32	
ReLU1	1		
Pooling1	3		
Conv2	11	64	
ReLU2	1		
Dropout1			0.5
Pooling2	3		
Conv3	7	96	
ReLU3	1		
Dropout2			0.15
Pooling3	2		
Conv4	5	128	
ReLU4	1		
Dropout3			0.25
Pooling4	2		
Fc5	1	5000	
ReLU5	1		
Dropout4			0.5
Fc6	1	1000	
ReLU6	1		
Fc7	1	2	
Softmax	1		

CNN Architecture

Comparison of classification performances

- Validated the proposed method using two publicly available databases i.e. Mammographic Image Analysis Society (MIAS)and CBIS-DDSM.
- Within 2620 scanned cases based on the magnitude of abnormality, the abnormal class is divided into two more classes, benign and malignant. We have taken total 273 benign cases and 273 malignant cases from CBIS-DDSM
- From 322 cases of mini-MIAS database 64 benign and 51 malignant cases are there in the abnormal class.

References	Techniques	Database	Classification performance
Xie et. al.[20]	Gray level features, textural features	MIAS DDSM	96.0% 95.7%
Jiao et. al.[21]	High & medium level, deep features	DDSM	96.7%
Arevalo et. al.[22]	CNN, SVM	DDSM	96.7%
Beura et. al.[23]	2D-DWT, GLCM	MIAS DDSM	98.0% 98.8%
Ours	**Segmented ROIs, CNN**	MIAS DDSM	**99.174%** **99.542%**

Visual Object Detection & Tracking

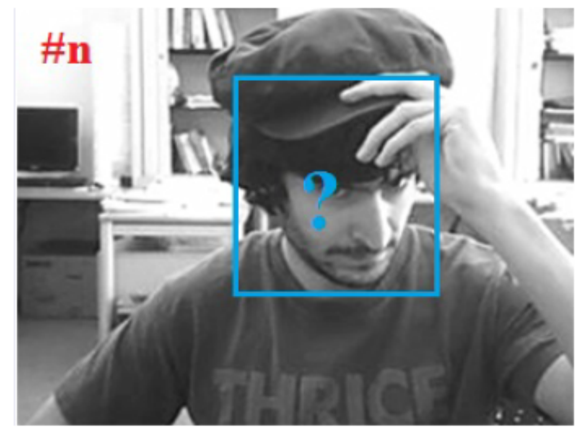
- Single Object Tracking (SOT)

Input: sequence of N images and initial bounding box of target

Objective: estimate target state over time (“track the target”)

initialization in 1st frame

. . .



estimated states in nth frame

- Multiple Object Tracking (MOT)

Input: set of object detections in every frame of the video sequence

Objective: association of state (id) for each detections (find the trajectory)

Single Object Tracking

- Single object, single camera
- Model free:
  - Nothing but a single training example is provided by the bounding box in the first frame
- Short term:
  - Tracker does not perform re-detection • Fail if tracking drifts off the target
- Subject to Causality:
  - Tracker does not use any future frames

Single Object Tracking

- Protocol:

  Setup tracker
  Read initial object region and first image
  Initialize tracker with provided region and image
  loop
      Read next image
      if image is empty **then**
          Break the tracking loop
      end if
      Update tracker with provided image
      Write region to file
  end loop
  Cleanup tracker

Visual Object Tracking

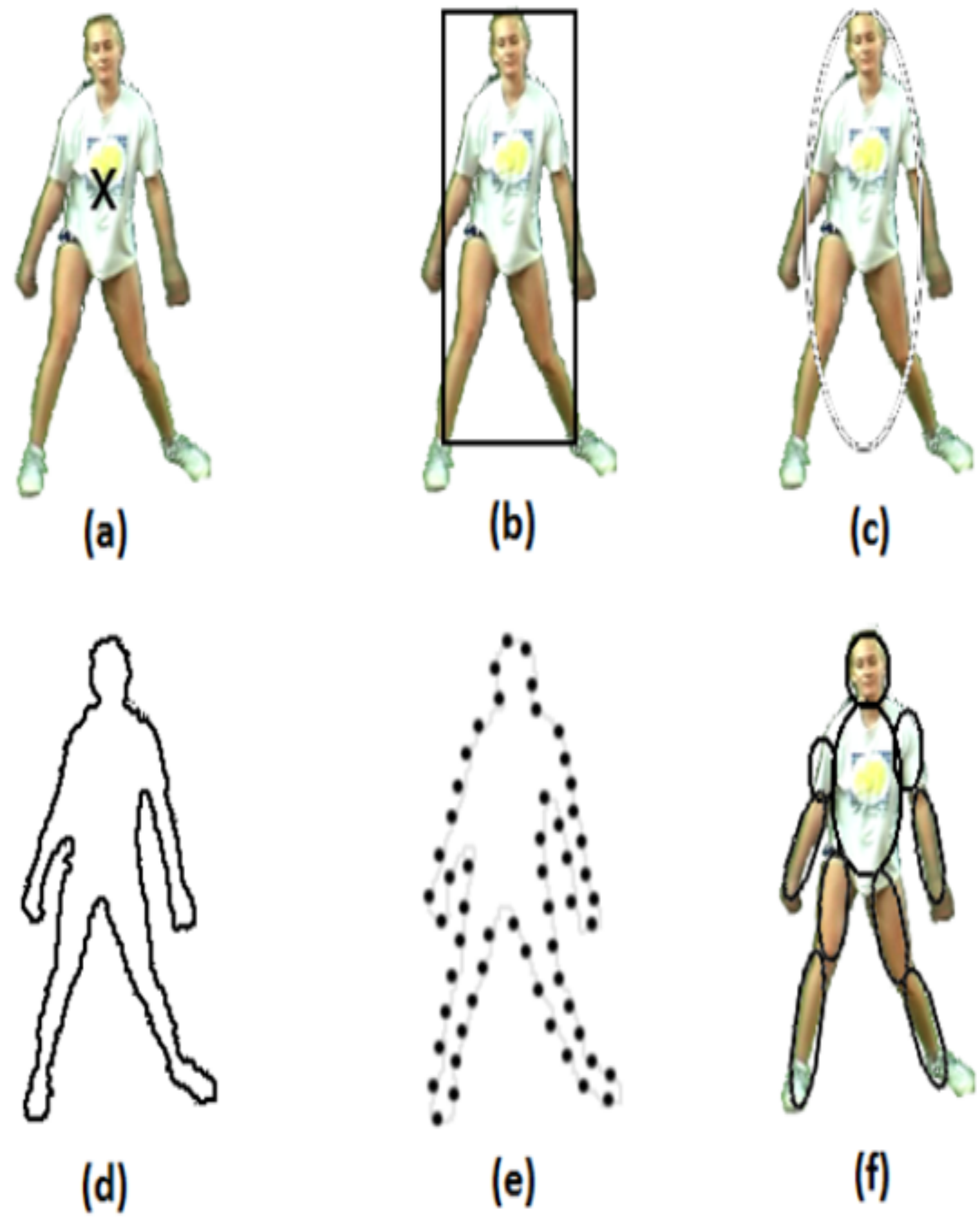
Single target tracking (model-free)

Initialization in the 1st frame

……

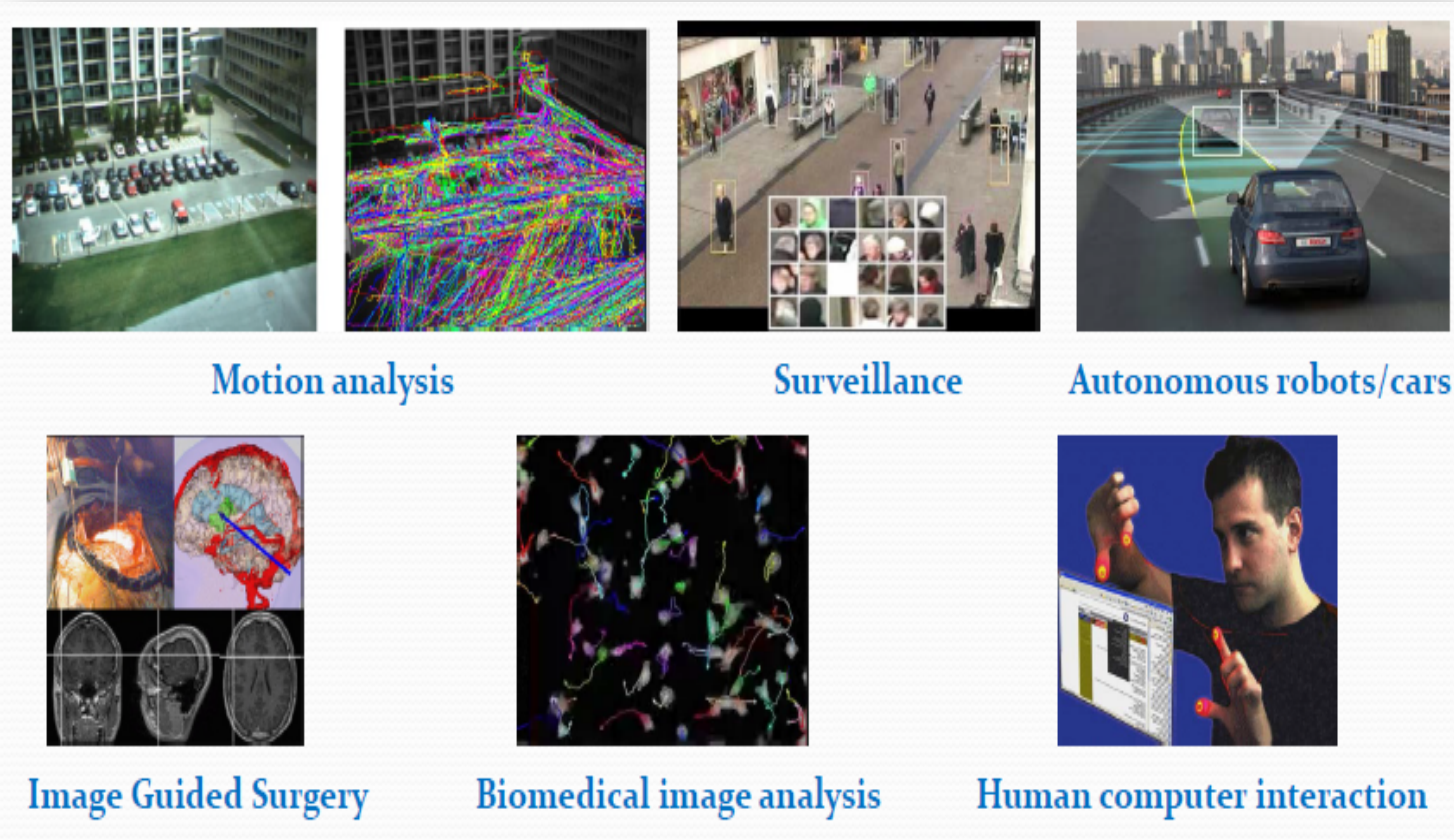
Estimation in the Nth frame

Object Representation



(a) Centroid (b) Rectangular (c) Elliptical (d) Object Contour
(e) Control points on the contour (f) Articulated objects

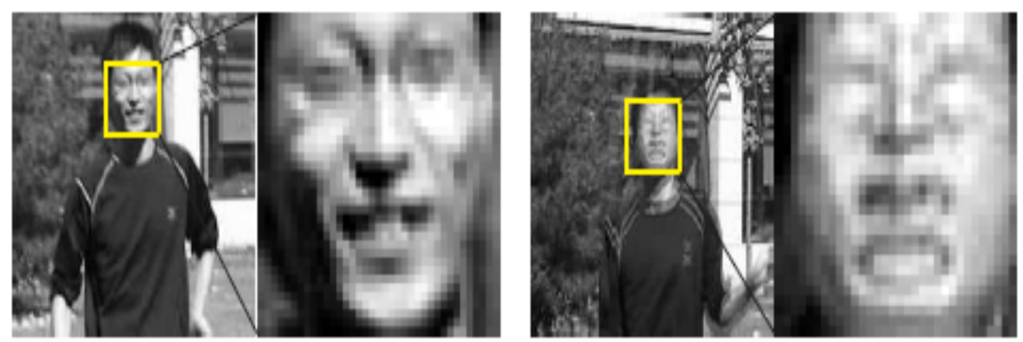
Tracking Applications



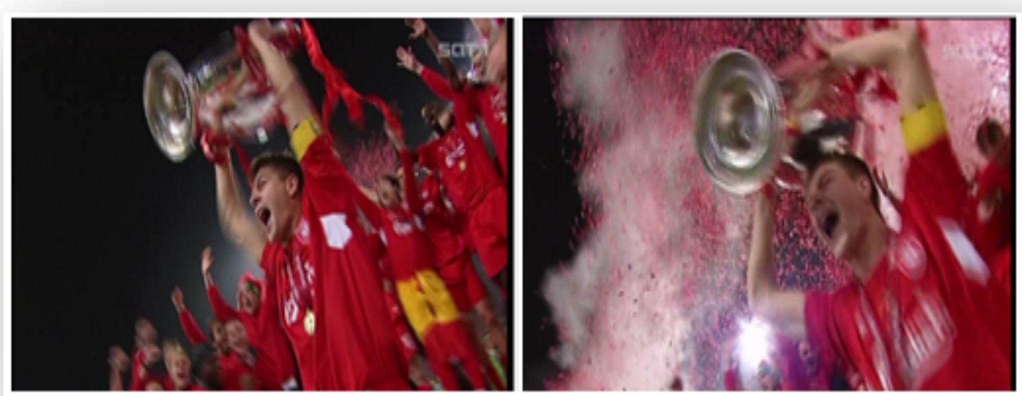
Challenges in Tracking

➢ Illumination Variations :

➢ Motion Blur :

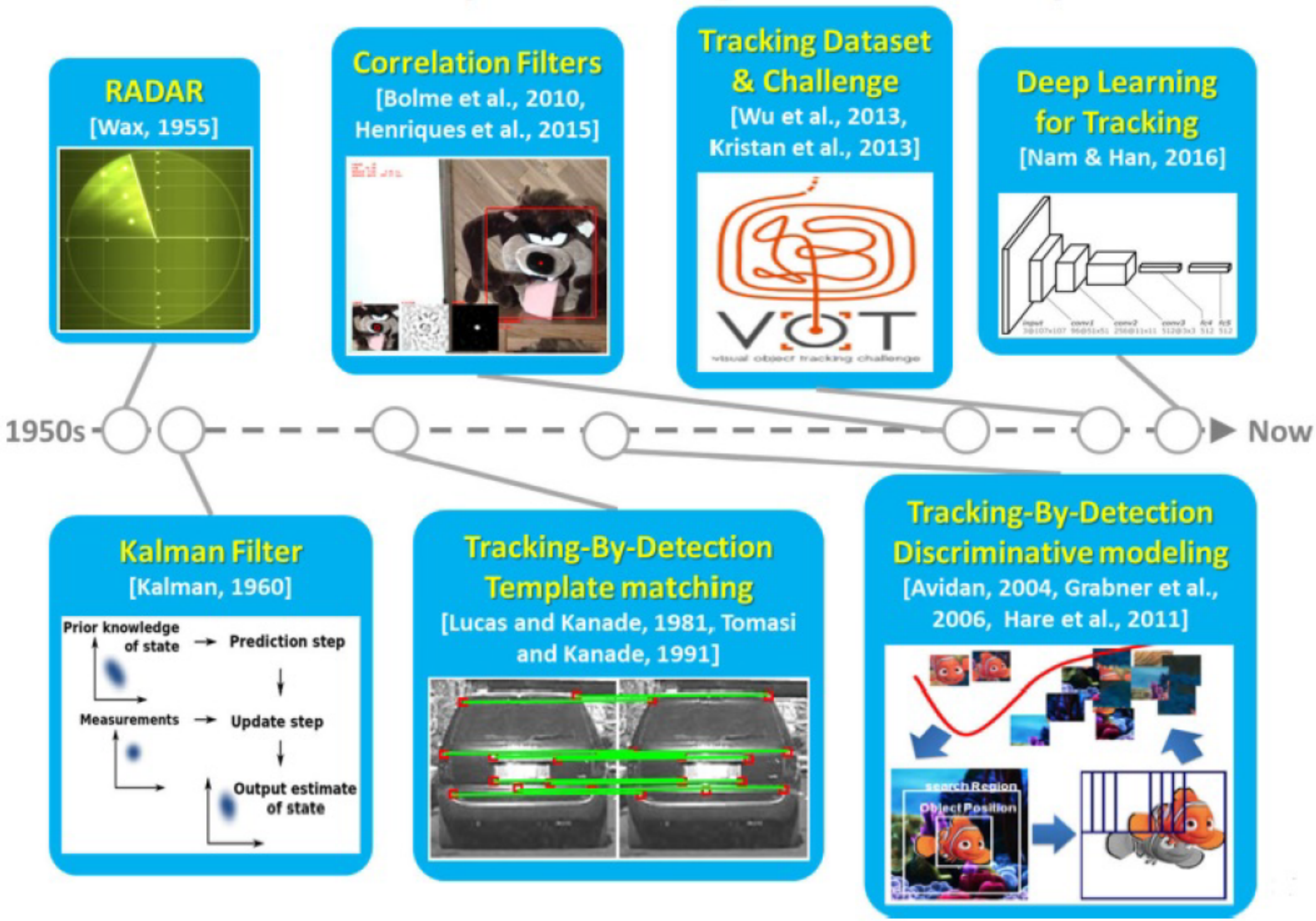


➢ Background Clutter :



Challenges in Tracking

Tracking: A Brief History



- # Single Object Tracking

Objectives

- Comprehensive study on Visual Object Tracking and its applications
- Critically analyze various state-of-the-art trackers and identify their shortcomings
- Augment state-of-the-art trackers with own contributions to tackle these issues
- Develop a robust visual object tracking algorithm on deep learning frame work

Contributions

- **GoogLeNet Tracker** : used pretrained GoogLeNet CNN architecture for feature representation; fast and computationally efficient tracker
- **SiameseFC-DSR & CFNet-DSR Tracker**: augmentation of displacement (**D**) and scale (**S**) consistency and rotation (**R**) adaptiveness methods in SiameseFC and CFNet tracker framework
- **RIDF-ECO and RIDF-SRDCF tracker:** augmentation of rotation (**R**) adaptiveness, illumination (**I**) elimination, displacement consitency (**D**) and false positive (**F**) rejection methods in ECO and SRDCF tracker framework.
- **WAET tracker: W**eighted **A**ggregation with **E**nhancement **F**ilter for enhancement of visual information

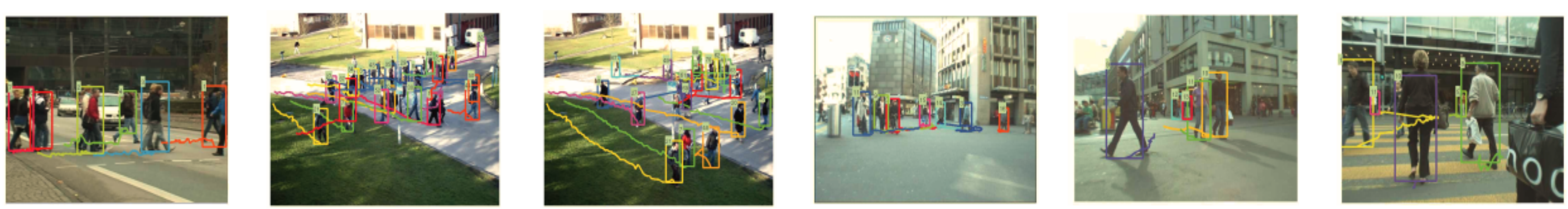
• Multiple Object Tracking

Objectives

- robustly associate noisy object detections on a new video frame with previously tracked objects
- Incorporate the concept learn-to-track (add learning capabilities to MOT)

Contibutions

- **MDP-MOT tracker**: formulate the data association in online multi-object tracking problem as decision making in Markov Decision Processes (MDPs)
- analyze the performance based on generative tracking (eg: TLD) and



TUD-Crossing #31 PETS09-S2L2 #68 PETS09-S2L2 #111 ETH-Jelmoli #82 ETH-Linthescher #51 ETH-Crossing #97

AVG-TownCentre #52 ADL-Rundle-1 #232 ADL-Rundle-3 #183 Venice-1 #235 KITTI-16 #90, KITTI-19 #281

INCORPORATING ROTATIONAL INVARIANCE IN CNN

- For tracking applications (video surveillance, spacecraft tracking…), first the object should be detected and given to the tracker as a ground truth.
- Deep learning based algorithms are being popularly used for object detection (Object recognition, object detection…).
- This object detection algorithms are desired to be invariant to transformation such as scaling, rotation, shift…

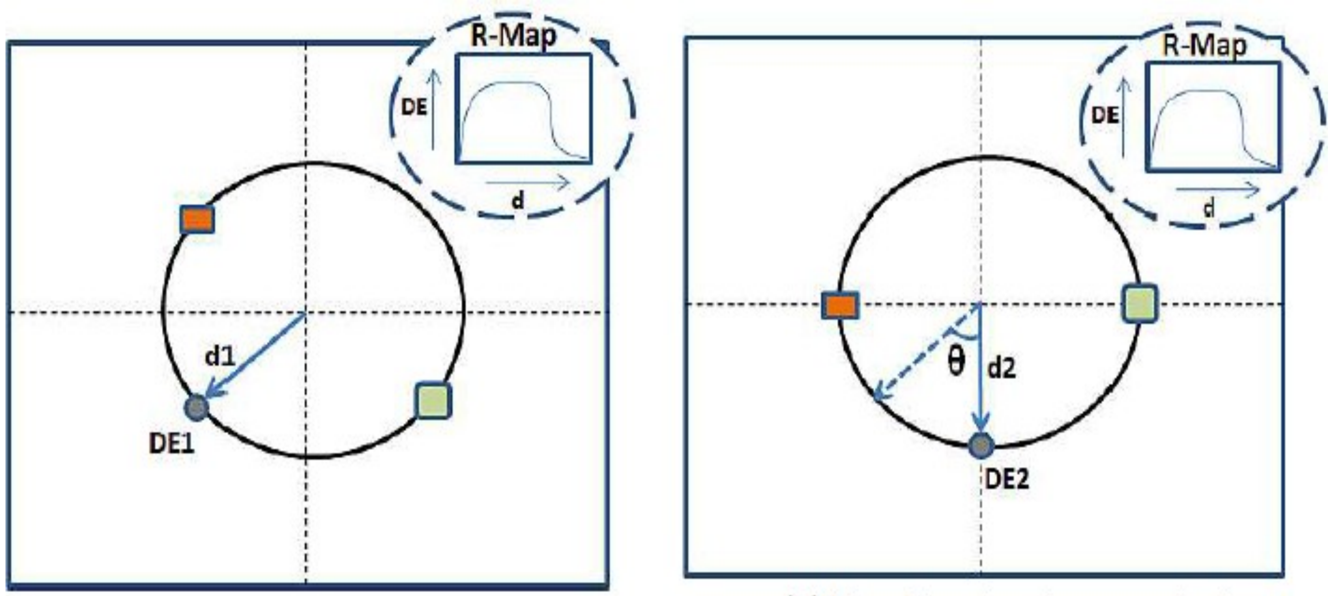
1. RICNN (Rotational Invariant CNN)

- A new layer for rotation-invariance is introduced

- **Idea:** Even after orientation of the sample from the centre of the image changes, following parameters remain invariant:
  - The current pixel's differential excitation with respect to neighbours (DE)
  - Distance of each pixel from the centre of the image (d)

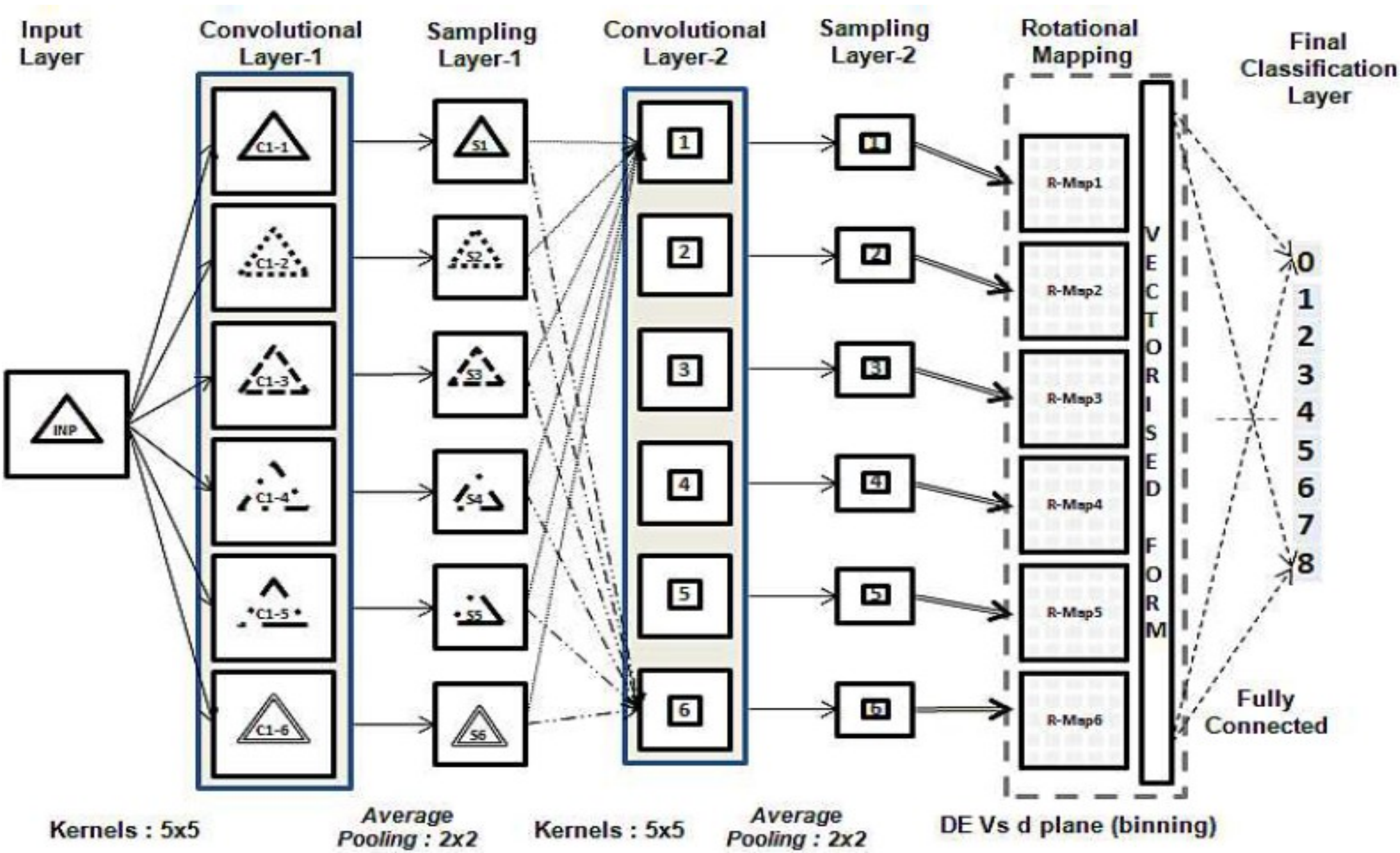
- The new layer consists of rotation-maps, which are formed by using above two metrics

R-Map

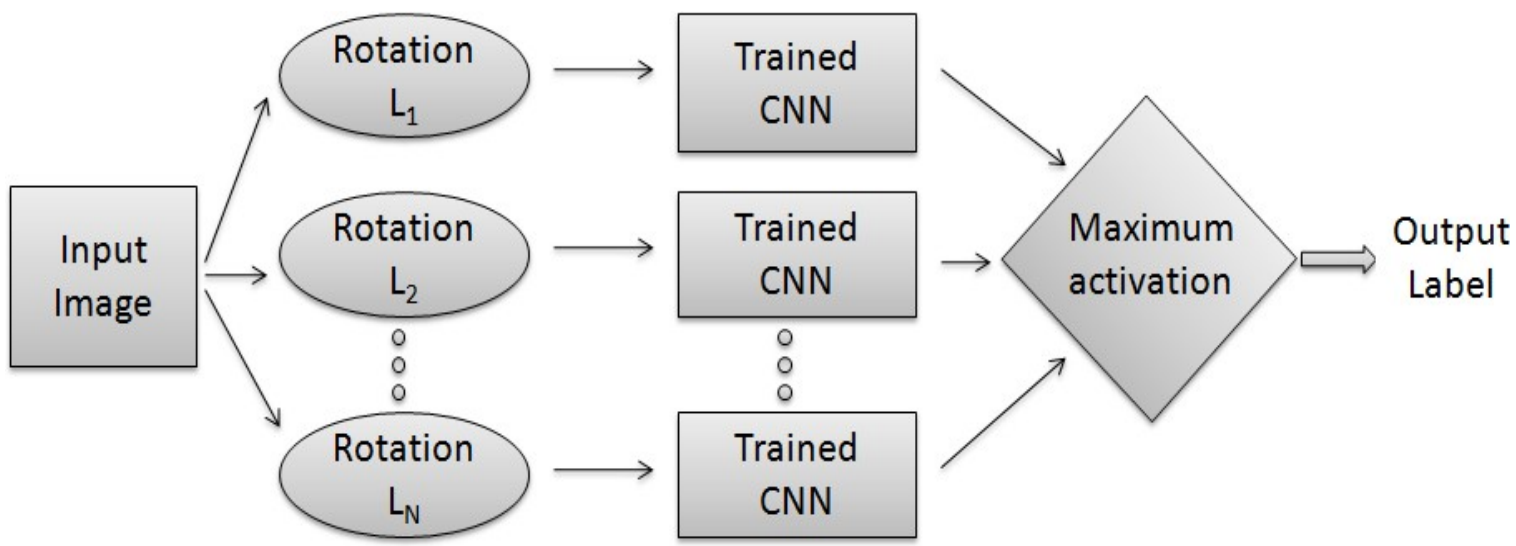


(a)Before Rotation (b)After Rotation by an angle θ

RICNN : Complete Architecture



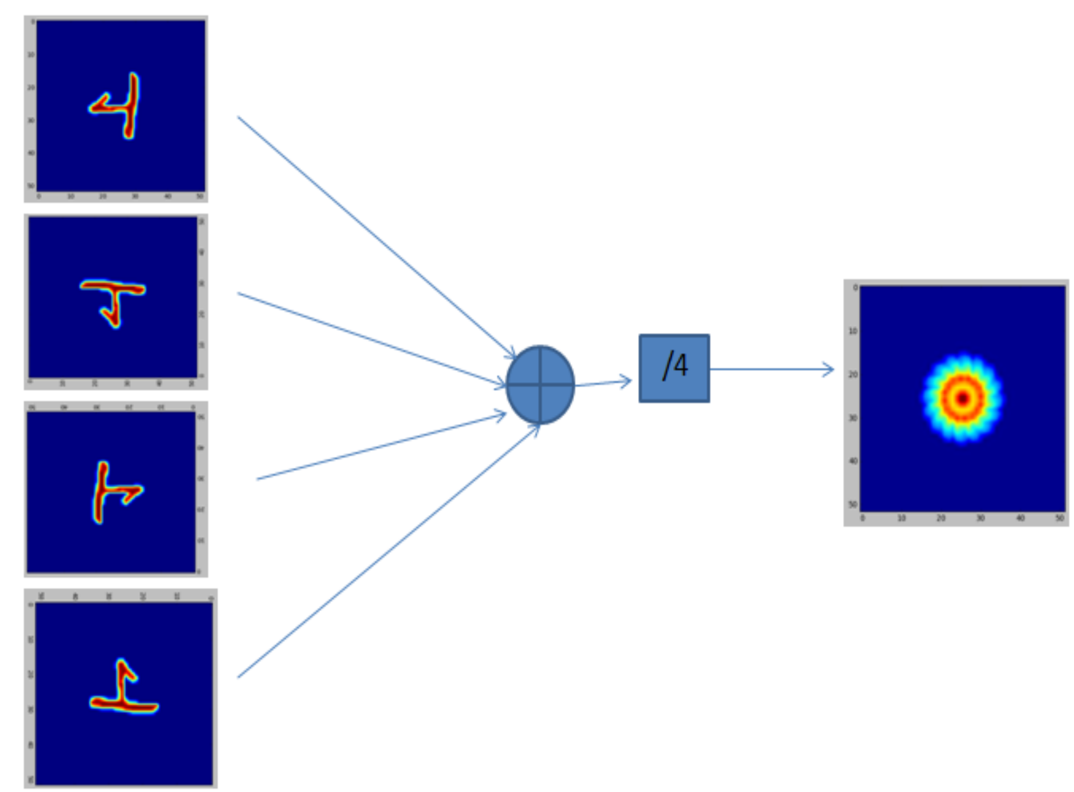
2.RIMCNN (Multiple Instance Testing)



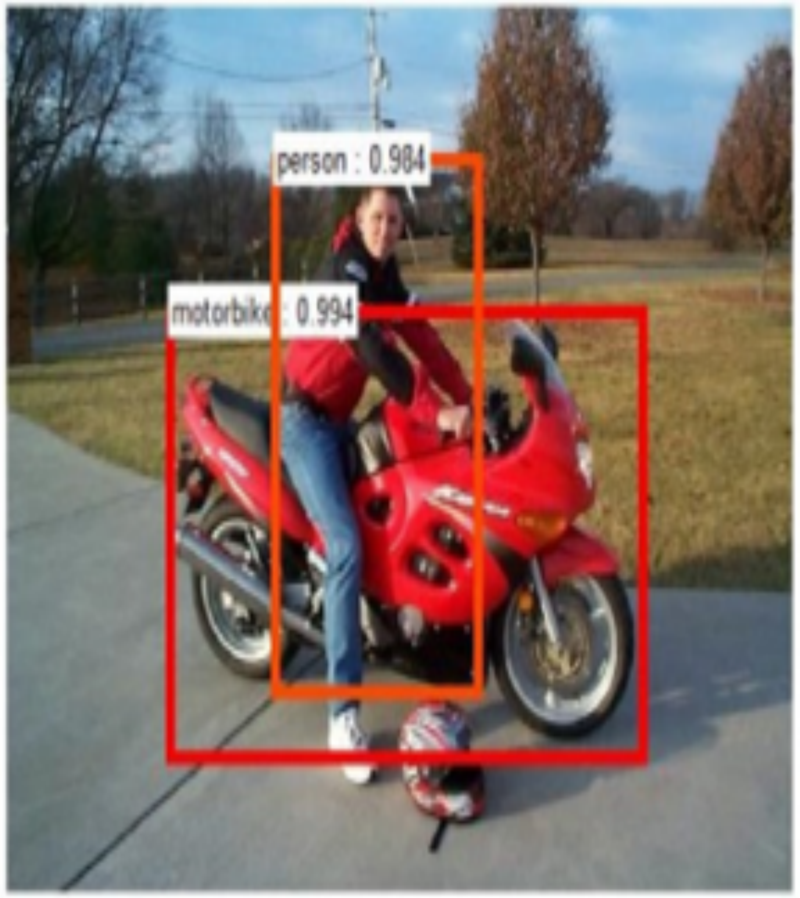
- Pre trained network can be used, without any modification.
- In testing different rotated copies of test images are used.

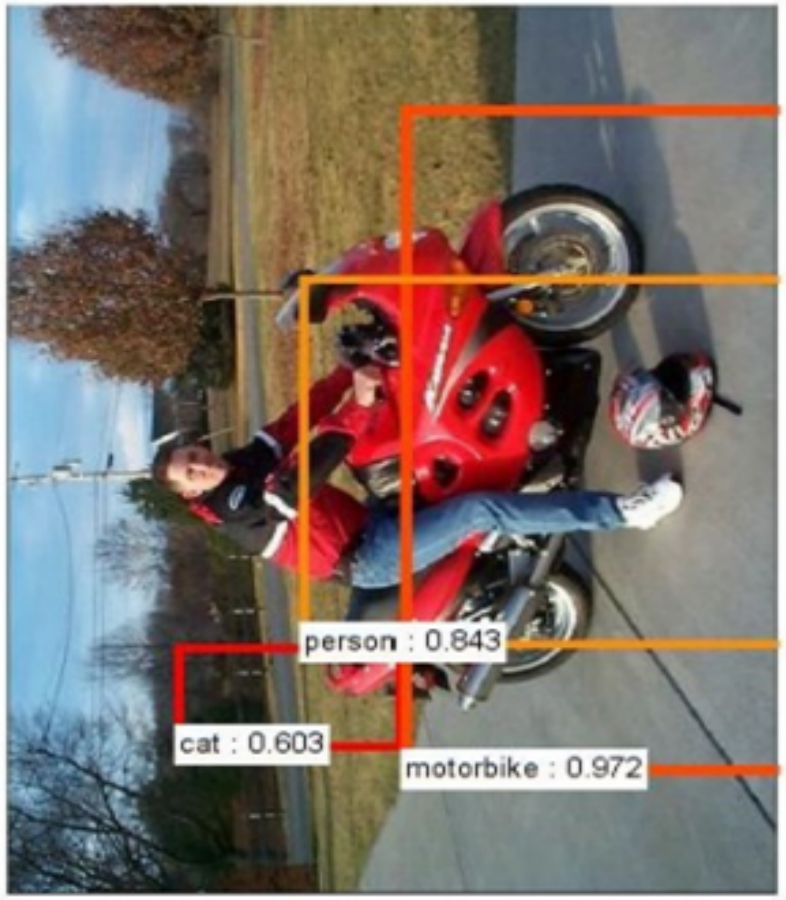
3. Rotational Symmetric Patterns (RSL Method)

- Mapping images to rotational invariant patterns by simple averaging.
- Networks learns rotation invariant features.



Demonstration of rotation invariance of RIMCNN.





No rotation

Rotation: Wrong detection

Faster RCNN

Rotation: correct detection

RIMCNN- Faster RCNN

4. Oriented Networks

Test image with unknown orientation and scale

Target train / test image (orientation corrected)

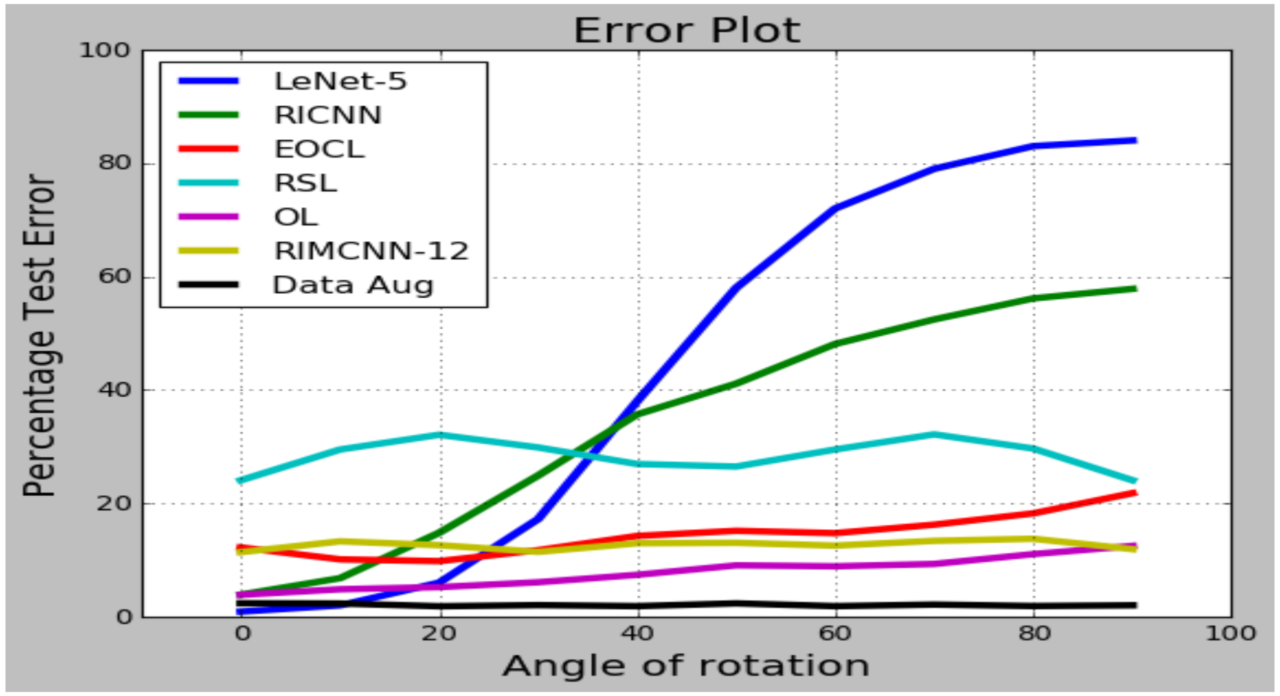
CNN

Rotation correction by finding PC and aligning to a unique direction

Scaling correction by rescaling according to eigen values in different directions

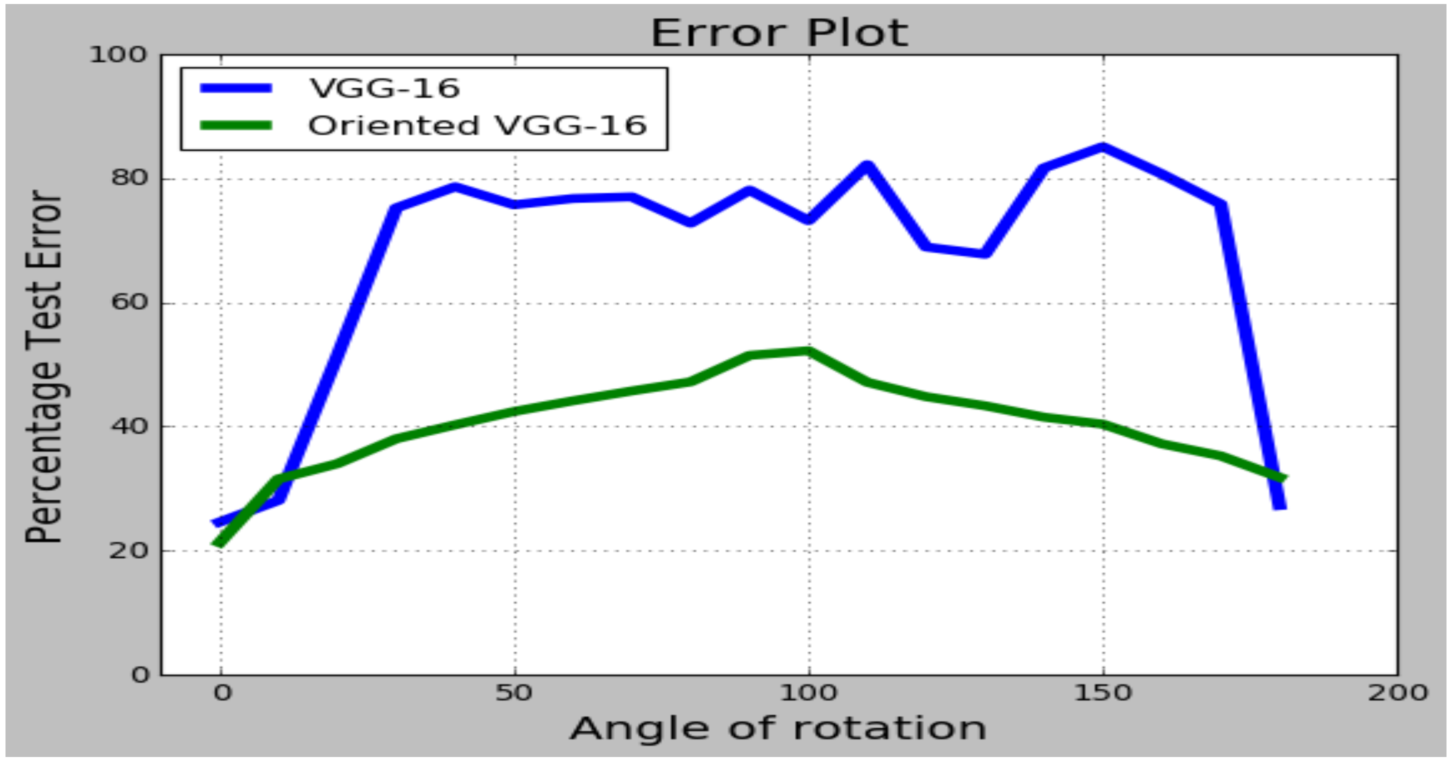
Used to correct the scale and orientation of test images.
CNNs can be trained with images of single orientation and scale and classify images of any scale and rotation.

Performance Analysis



Digits - MNIST

English Alphabets - NIST



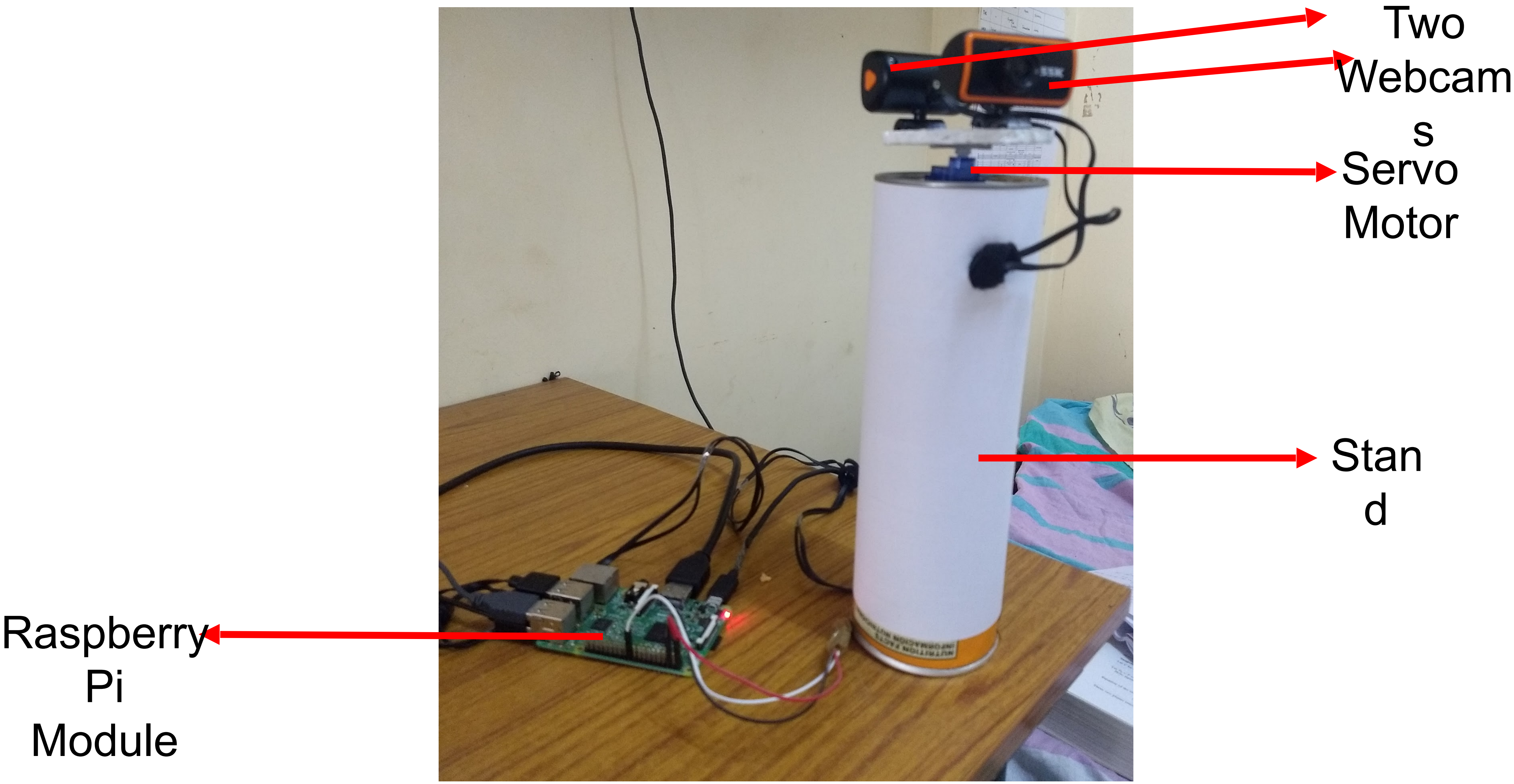
CIFAR – 10 dataset

360 Degree Field of View Camera

Goal

- To develop a 360 Degree Field of View Camera, using two inexpensive webcams. The two cameras are mounted on a stand which is rotated by a servo motor and multiple images are taken. These images are then stitched to form a panorama.
- Software :
  - Interfacing cameras using Raspberry Pi and Python
  - Stitching the multiple images using OpenCV library
  - Development of unique and efficient algorithm for image stitching, to get better result.
  - Using image stabilization techniques for enhanced quality of images.
- Hardware:
  - Development of robust stand for the camera to avoid destabilization, while the servo motor is rotating.
  - Development of portable module which houses camera, battery pack and Raspberry Pi.

Current Prototype



Preliminary Results

Stitched Image:

Boundary between the 2 Images

Image 1

Image 2

Thank you !!!