56th IEEE Conference on Decision and Control
Melbourne, Australia
December 13, 2017

Deep Learning Feature Extraction for Target Recognition and Classification in Underwater Sonar Images

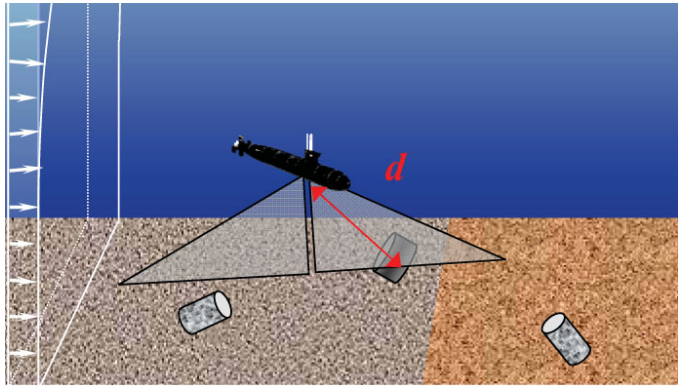
Pingping Zhu, Jason Isaacs, Bo Fu, and Silvia Ferrari
Mechanical and Aerospace Engineering
Cornell University

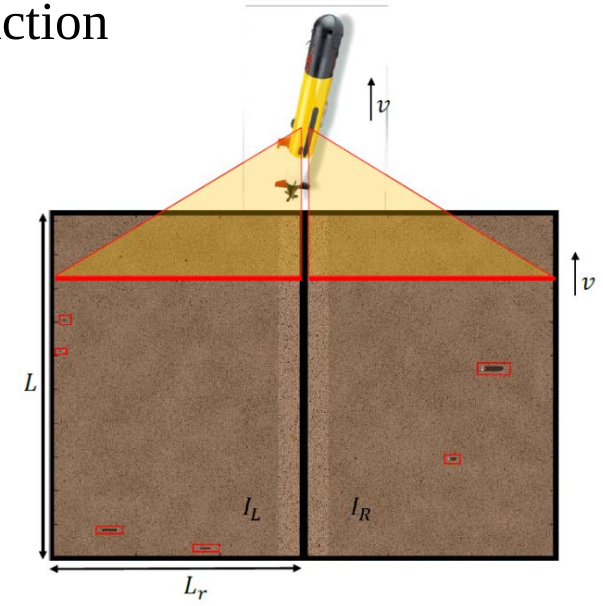
Outline

- Motivation
- Problem Formulation
- UUV-sonar Feature Extraction
- Automatic Target Recognition (ATR)
- Results
- Conclusions and Q&A

Motivation

- Automatic Target Recognition (ATR) and classification are important for a wide range of autonomous systems and applications.
- ATR eliminates manually classifying targets by expert human operators – costly and slow
- For sonar images, handcrafted features may fail to extract meaningful information – noise, clutter, low resolution, and low contrast
- Choose to leverage convolutional neural network's (CNN's) proven ability to effectively perform highly nonlinear feature extraction





Problem Formulation

Consider the problem in which multiple images I are obtained by a mobile UUV, to recognize, segment, and classify one or more objects of interest, each belonging to one of two classes referred to as target c_0 and non-target c_1.

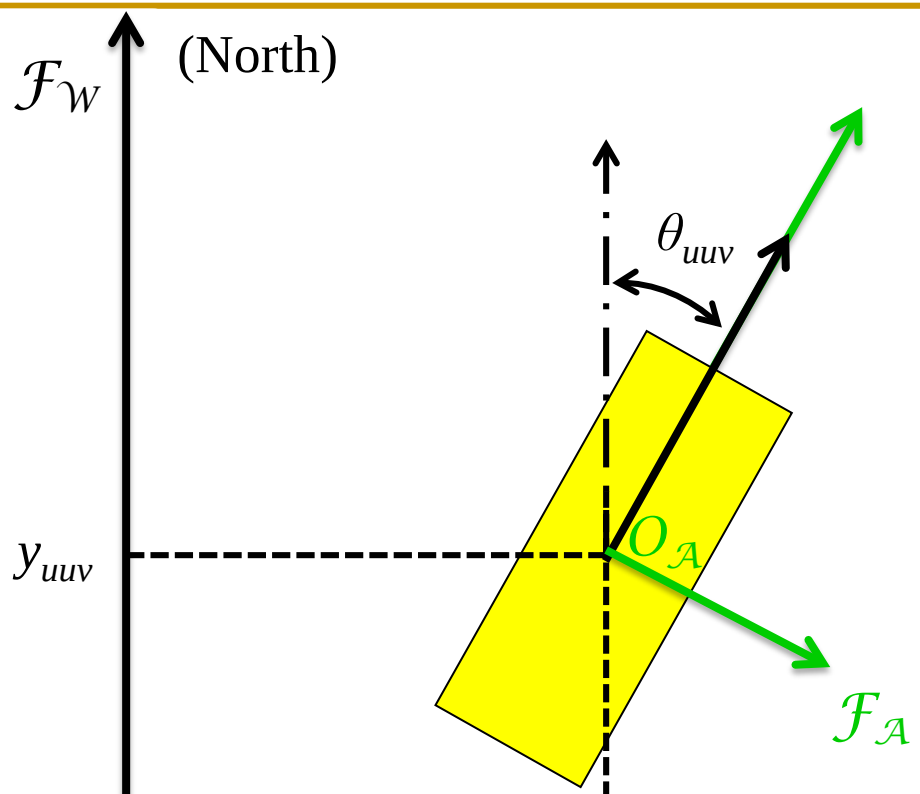
To implement the binary image classification, the following steps include:

1) Pre-image processing is applied to obtain the enhanced image I_{enhance}
2) Image segmentaion is applied to obtain the image segment of interesting objects denoted by K.
3) Extract the features from the image segments K denoted by $\mathbf{z}$
4) Classify the objects by using the extracted features via a classification function: $$f(K) = y : R^{n_{s'} \times n_{t'}} \mapsto \{0,1\}$$

- f is first learned from training dataset,
- f is then used for testing. The function f is applied to determine the class variable u:

$$u = \begin{cases} c_0, & \text{if } y = f(K) = 0 \\ c_1, & \text{if } y = f(K) = 1 \end{cases}$$

UUV Kinematics Frames of Reference



Assume:

v = *constant*

UUV States:

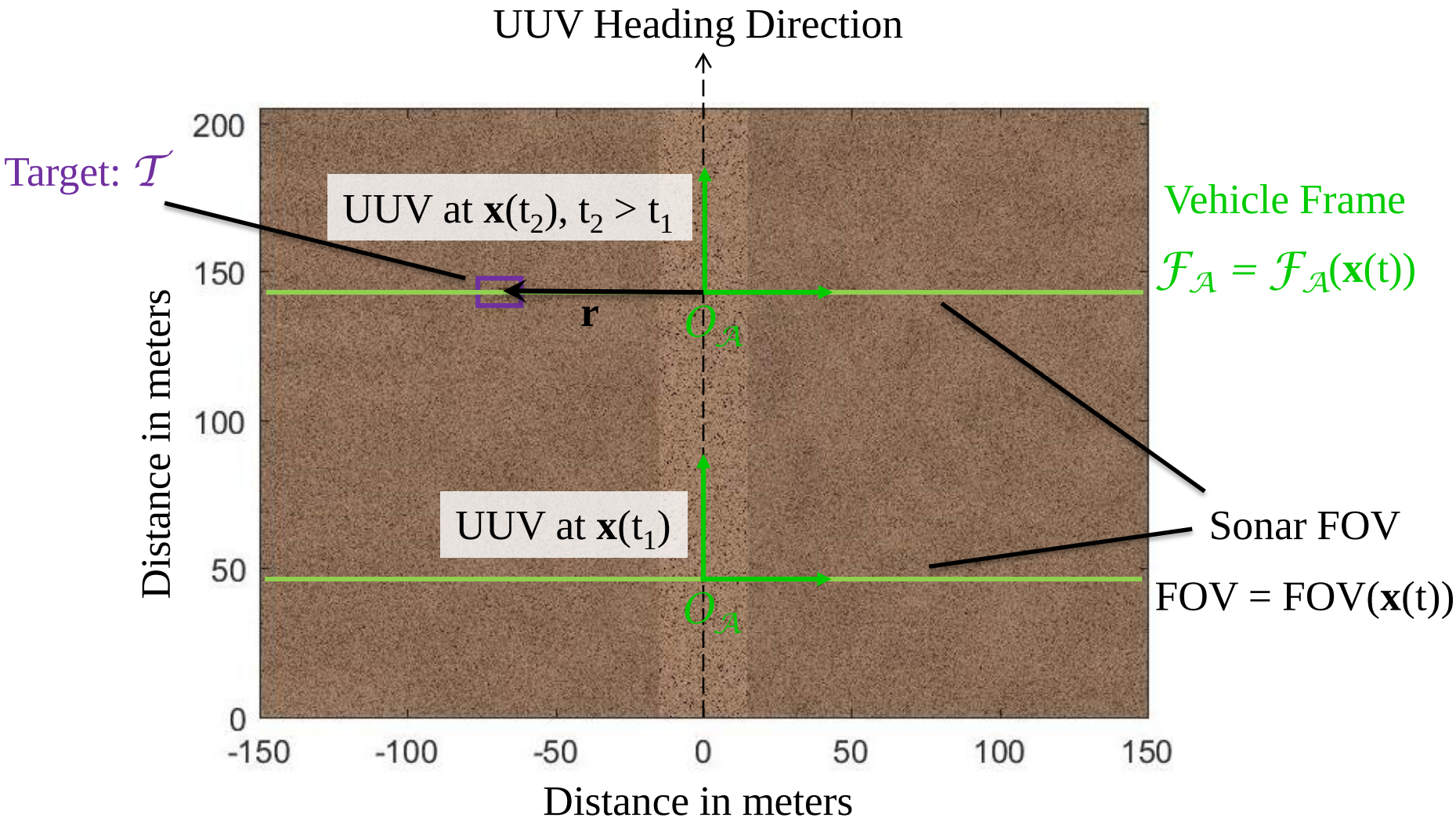
$\mathbf{x}(t) = [x_{uuv}(t), y_{uuv}(t), \theta_{uuv}(t)]^T$

UUV Dynamics:

$$\dot{x}_{uuv} = v \cdot \cos(\theta_{uuv})$$

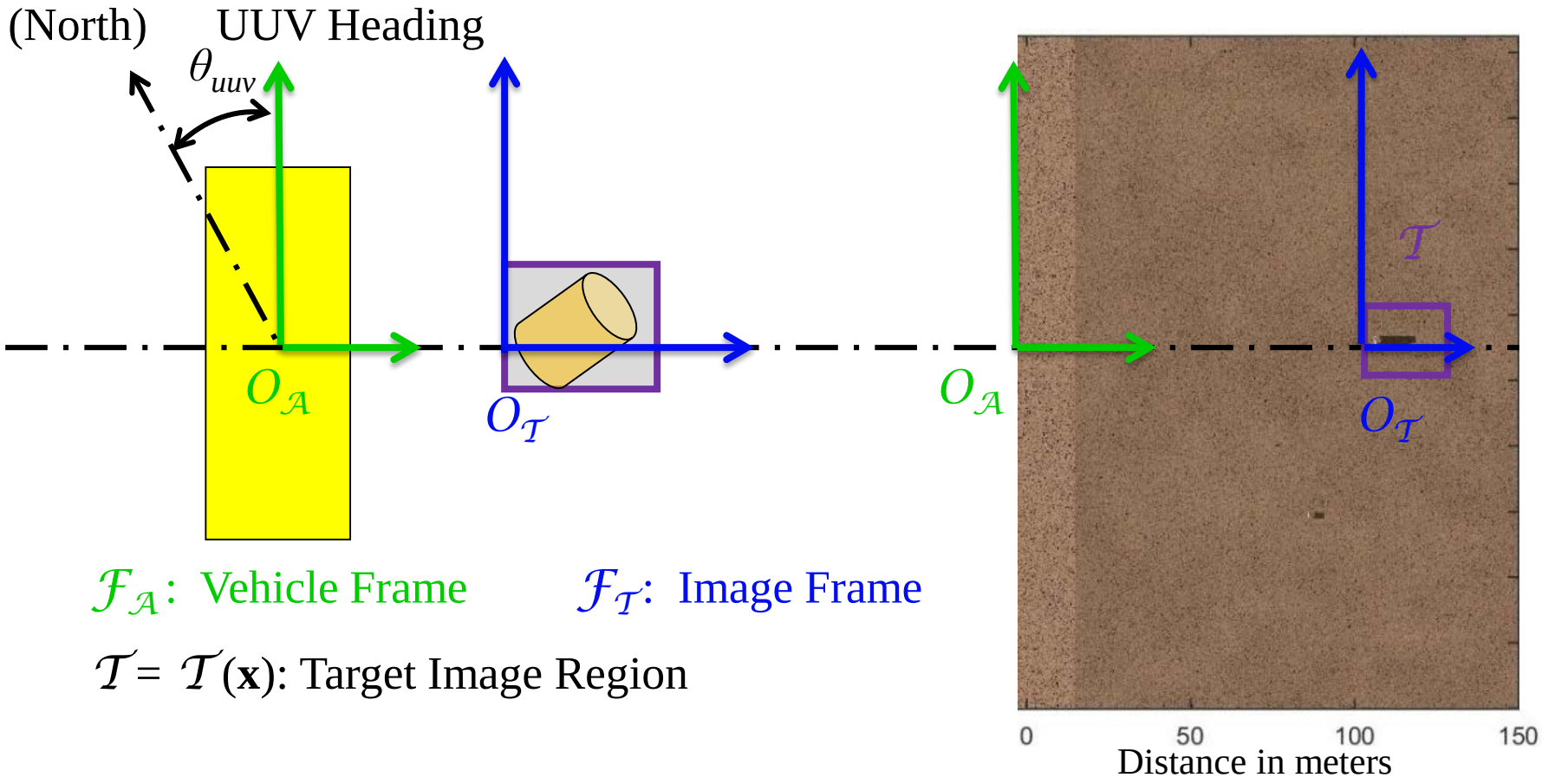
$$\dot{y}_{uuv} = v \cdot \sin(\theta_{uuv})$$

UUV Frames Relative to Sonar Image



Vehicle Frame and Image Frame

• Consider the image segmentation as the target geometry $\mathcal{T}$. Then, the actual target geometry, orientation, and shape are viewed as hidden random features.



$\mathcal{F}_{\mathcal{A}}$: Vehicle Frame $\mathcal{F}_{\mathcal{T}}$: Image Frame

$\mathcal{T} = \mathcal{T}(\mathbf{x})$: Target Image Region

Image Frame and Target Frame

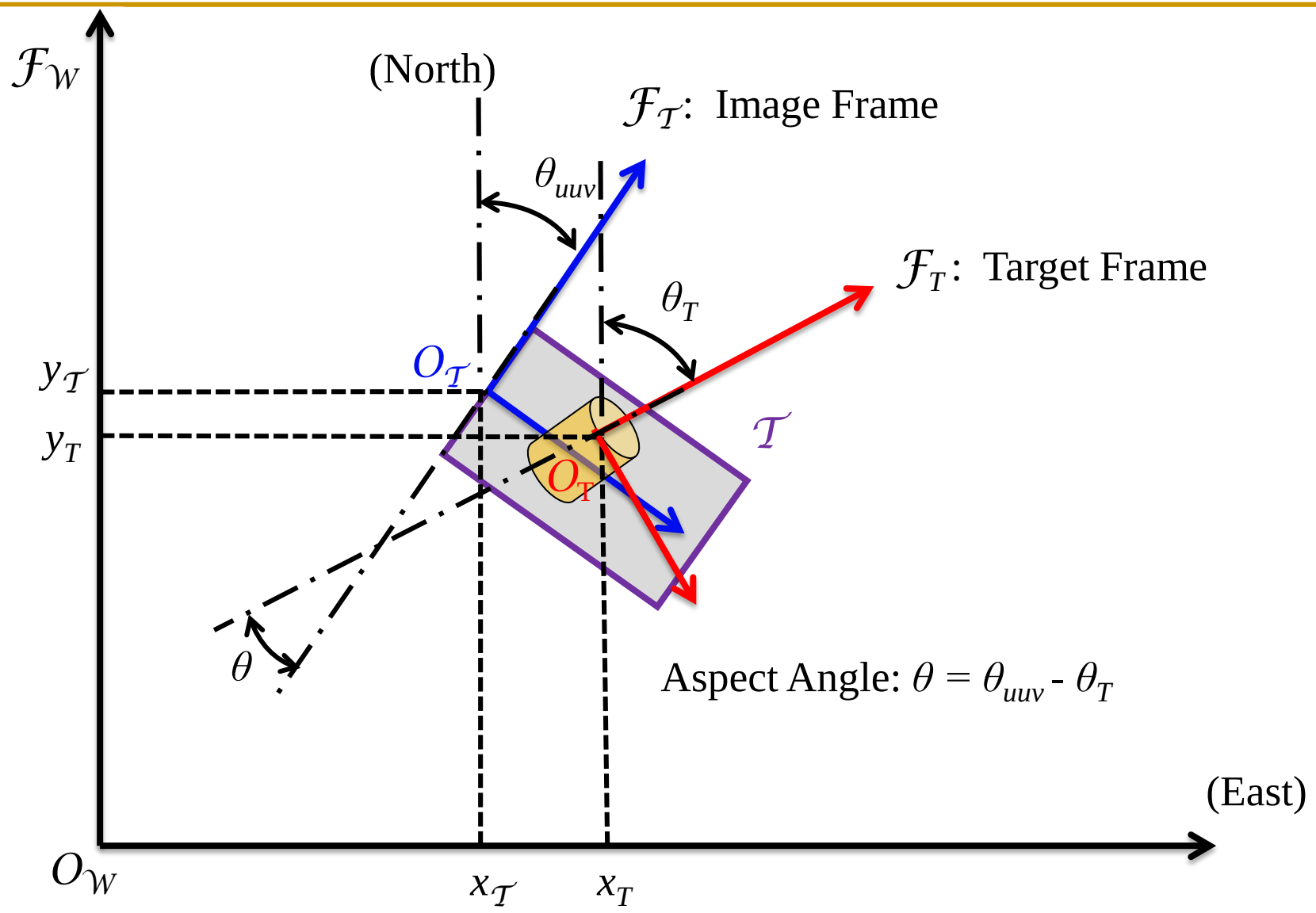
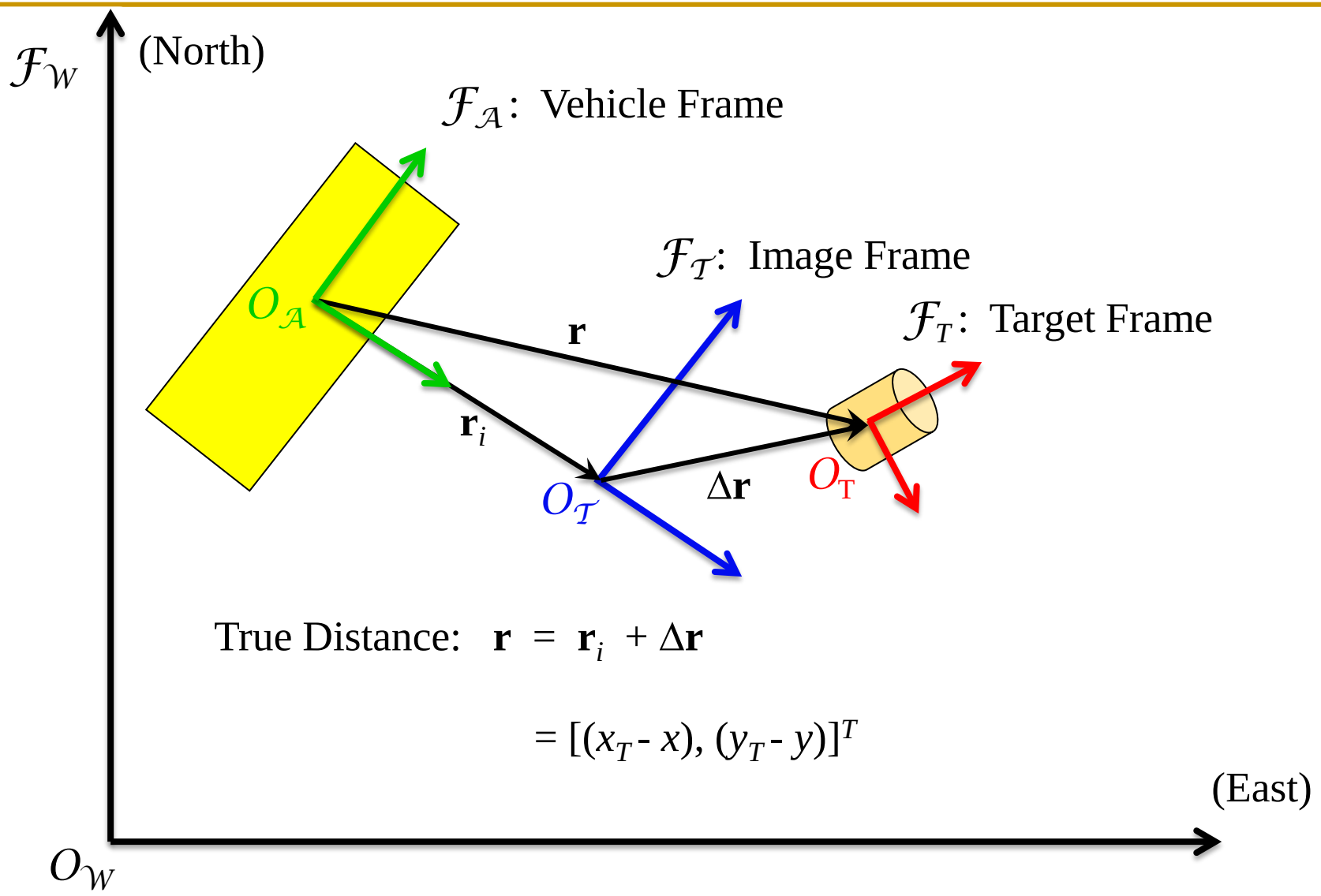
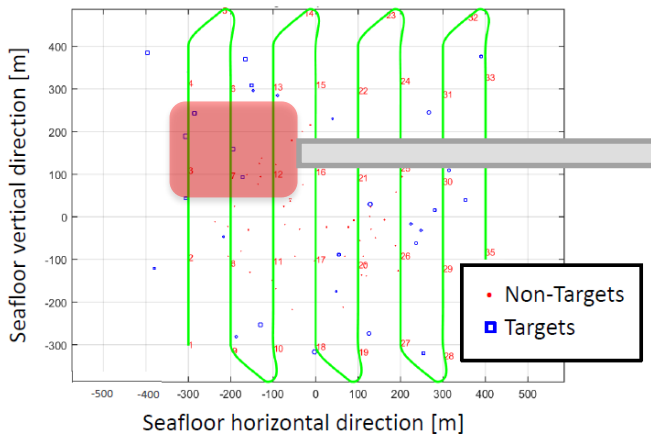
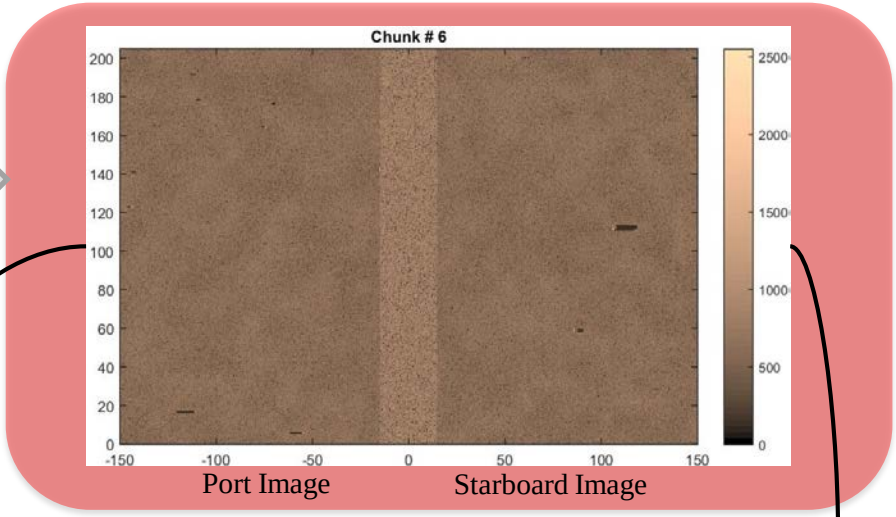


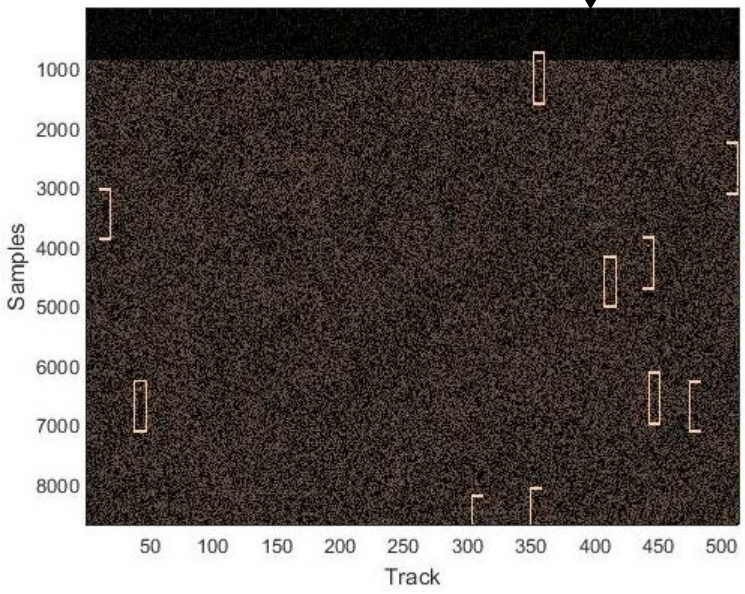
Image and Target Distance Vectors



Sonar Image Data







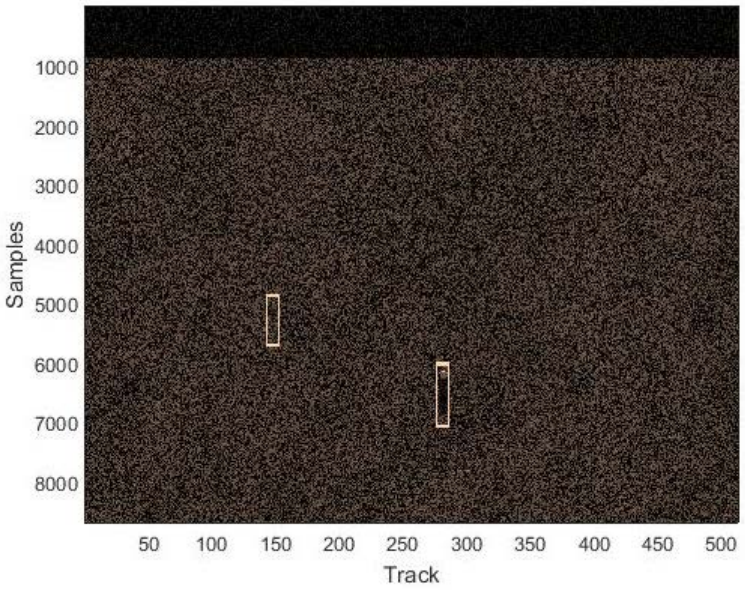


Image Data Description

Images matrix:

Data structure of the sonar image

(8684, 512)	(8683, 512)	…	(2, 512)	(1,512)	(1,512)	(2,512)	…	(8683, 512)	(8684, 512)
(8684, 511)	(8683, 511)	…	(2,511)	(1,511)	(1, 511)	(2, 511)	…	(8683, 511)	(8684, 511)
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
(8684, 2)	(8683, 2)	…	(2,2)	(1,2)	(1,2)	(2,2)	…	(8683, 2)	(8684, 2)
(8684, 1)	(8683, 1)	…	(2,1)	(1,1)	(1,1)	(2,1)	…	(8683, 1)	(8684, 1)

Port Image

Starboard Image

Track-axis:

- 512 tracks
- 0.4 m interval distance
- Range: $0.4 \times 512 = 204.8$ m

Sample-axis:

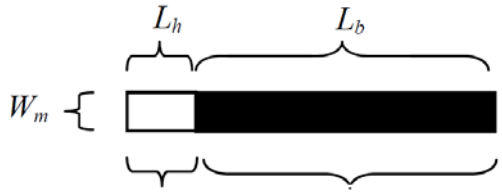
- Resolution: 8685 samples
- Range: 150 m

Object Detection – Matched Filter

To recognize and segment object, the following steps are implemented:

1. Normalize linearly the down-sampled image I_d to obtain the grayscale image I_g.

$$I_g(i,j) = \frac{I_d(i,j)}{\max\limits_{i,j}\left[I_d(i,j)\right]}$$



2. Convert the image I_g to the binary image I_b

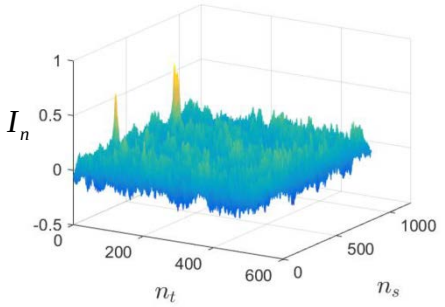
$$I_b(i,j) = \begin{cases} 1, & \text{if } I_g(i,j) \geq \theta_b \\ -1, & \text{if } I_g(i,j) < \theta_b \end{cases}$$

3. A matched filter is applied to detect the object.

$$I_n(i,j) = \frac{\sum_{\iota=1}^{W_m}\sum_{\zeta=1}^{L_h+L_b} I_b(i+\iota, j+\zeta) I_m(\iota,\zeta)}{\sum_{\iota=1}^{W_m}\sum_{\zeta=1}^{L_h+L_b} I_m^2(\iota,\zeta)}$$

where I_m is the matched filter defined by

$$I_m(i,j) = \begin{cases} 1, & \text{for } i \in [1, W_m] \text{ and } j \in [1, L_h] \\ -1, & \text{for } i \in [1, W_m] \text{ and } j \in [L_h+1, L_h+L_b] \end{cases}$$

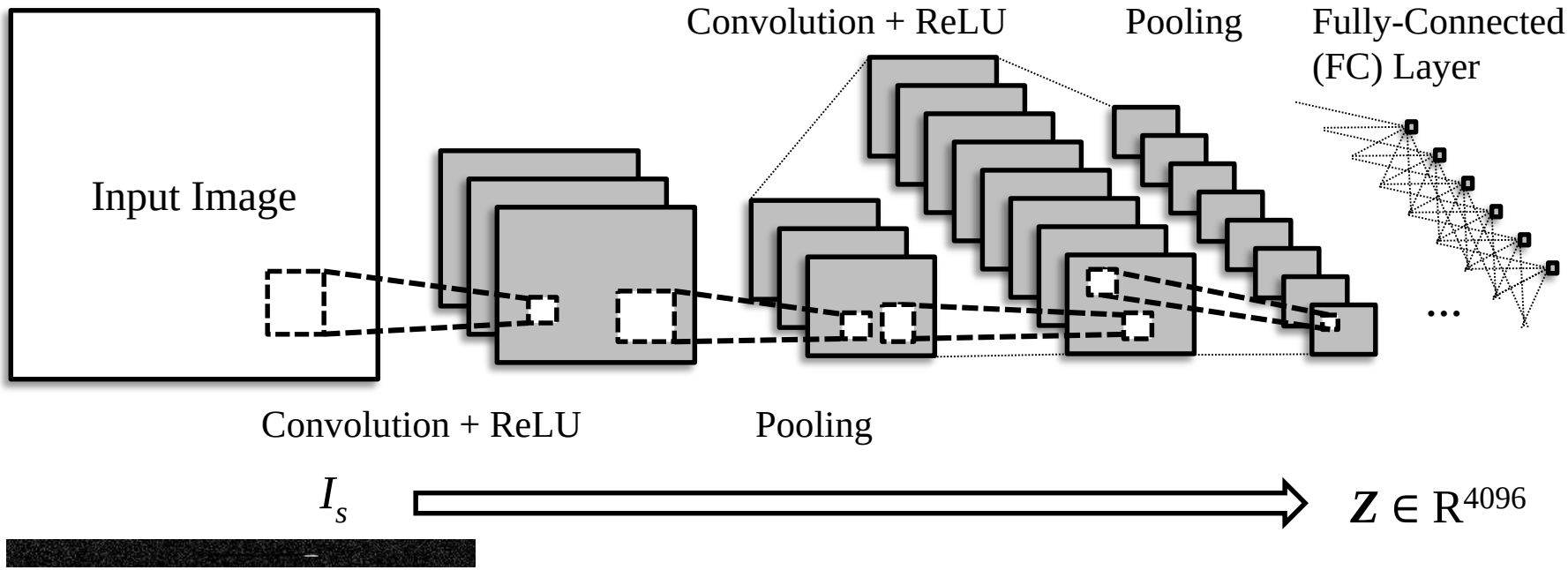


5. Points selected by the kth matched filter are denoted by $\boldsymbol{\sigma}(k) = \{(i,j) \mid I_n(i,j) \geq \theta_m\}$ and the combined set of points selected by all k_0 matched filters is denoted by

$$\boldsymbol{\sigma}_{tot} = \boldsymbol{\sigma}(1) \cup \ldots \cup \boldsymbol{\sigma}(k) \cup \ldots \cup \boldsymbol{\sigma}(k_0)$$

UUV-Sonar Feature Extraction

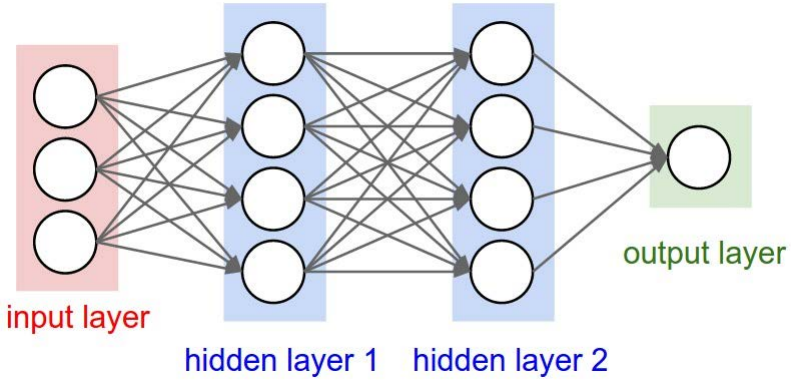
UUV-Sonar Feature Extraction



- Automatic segmentation is performed via matched filter
- Matched filter provides better performance than Markov random fields (MRFs)
- Deep learning features extracted from segmentation by Pre-trained AlexNet
- AlexNet CNN provides better performance than other features extraction techniques such as (Histogram of oriented gradients) HOG and Local binary pattern (LBP).

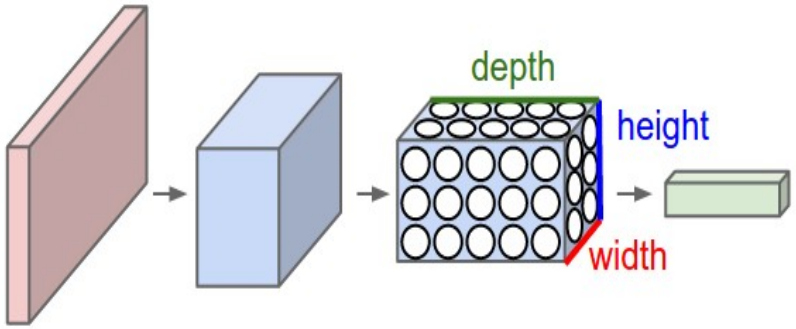
Convolutional Neural Network

Comparison between Conventional and Convolutional Neural Networks



Architecture of the conventional neural network

- 1D input vector
- 1D output scalar/vector
- Fully connected NN is applied in CNN



Architecture of the convolutional neural network

- 3D input volume
- 3D output volume
- Different types of layers

Convolutional Neural Network

Types of layers to build CNN architecture:

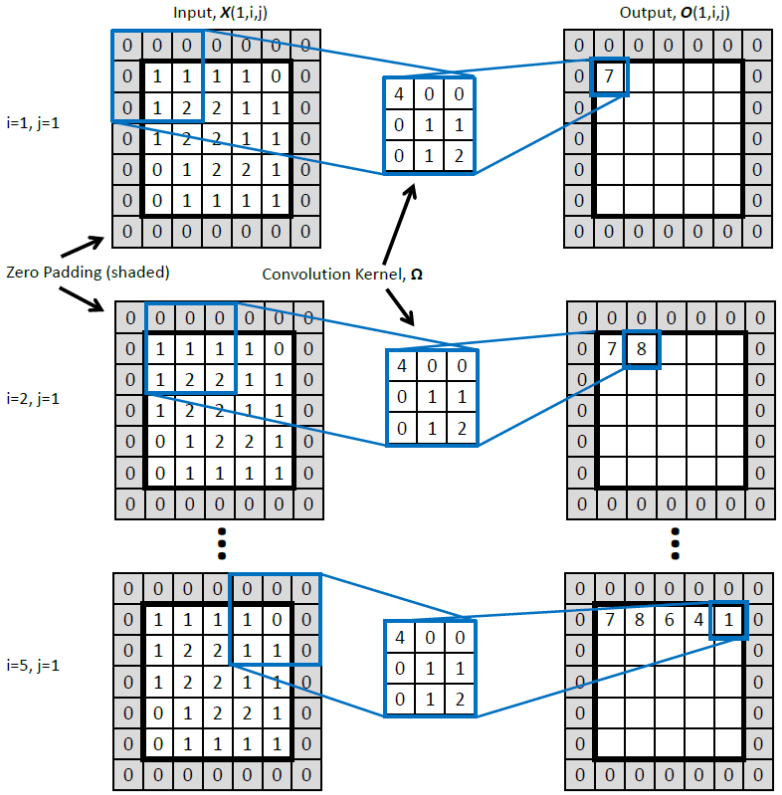
- **Input layer:** The raw pixel values of the image are input in 3-D volumes.
- **Conv layer:** The output of neurons are computed that are connected to local regions in the input, each computing a dot product between their weights and small region they are connected to in the input volume.
- **RELU layer:** An element-wise activation function is applied, such as the max(0,x) thresholding at zero.
- **POOL layer:** A downsampling operation is performed along the spatial dimensions (width, height), resulting in volume.
- **Fully-connected (FC) layer:** The class scores are computed to implement classification. As with ordinary Neural Networks and as the name implies, each neuron in this layer will be connected to all the numbers in the previous volume.

Convolutional Neural Network

Convolution operation by Linear Filter:

The convolution operator is defined as

- Filter spatial extent: $F = 3$
- Source pixel: $X(d,i,j)$
- Convolution kernel parameter: $\omega_{k,d,\iota,\varsigma}$
- Stride size: $S = 1$
- Zero padding: $P = 1$

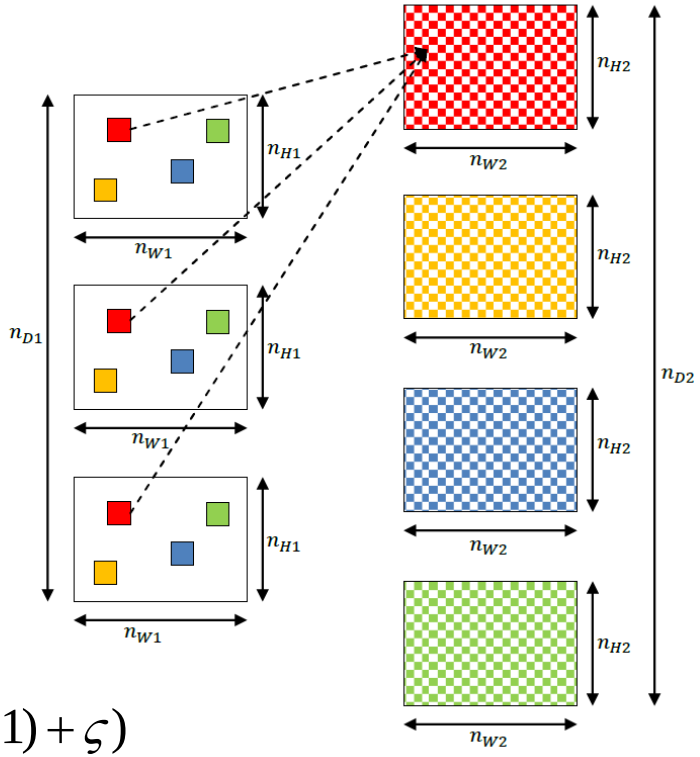


Convolutional Neural Network

Convolutional Layer:

The process of convolutional layer is demonstrated:

- Left side volume of size: $n_{D1} \times n_{W1} \times n_{H1}$
- Right side volume of size: $n_{D2} \times n_{W2} \times n_{H2}$



Using P and S, the input and output dimensionalities of the convolutional layer are related by

$$n_{W2} = (n_{W1} - F + 2P)/S + 1$$

$$n_{H2} = (n_{H1} - F + 2P)/S + 1$$

The convolutional operation is defined by

$$O(k,i,j) = \sum_{d=1}^{n_{D1}} \sum_{\iota=1}^{F} \sum_{\varsigma=1}^{F} \omega_{k,d,\iota,\varsigma} X'(d, i(S-1)+\iota, j(S-1)+\varsigma)$$

Convolutional Neural Network

Rectified Linear Units (ReLU) Layer:

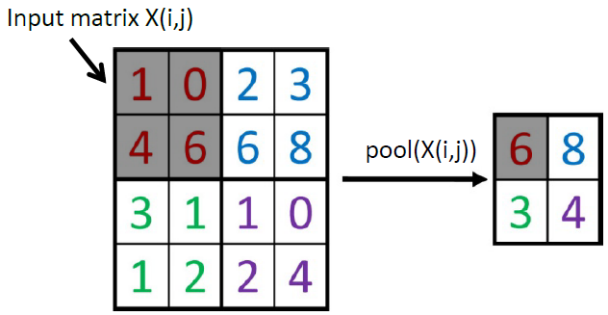
- This is a layer of neurons that applies the non-saturating activation function, which increases the nonlinear properties of the neural networks.

$$f(x) = \max(0, x)$$

- Other function are also applied to increase nonlinearity, such as
  - Saturating hyperbolic tangent function: $f(x) = \tanh(x)$, and $f(x) = |\tanh(x)|$
  - Sigmoid function: $f(x) = \left(1 + e^{-x}\right)^{-1}$

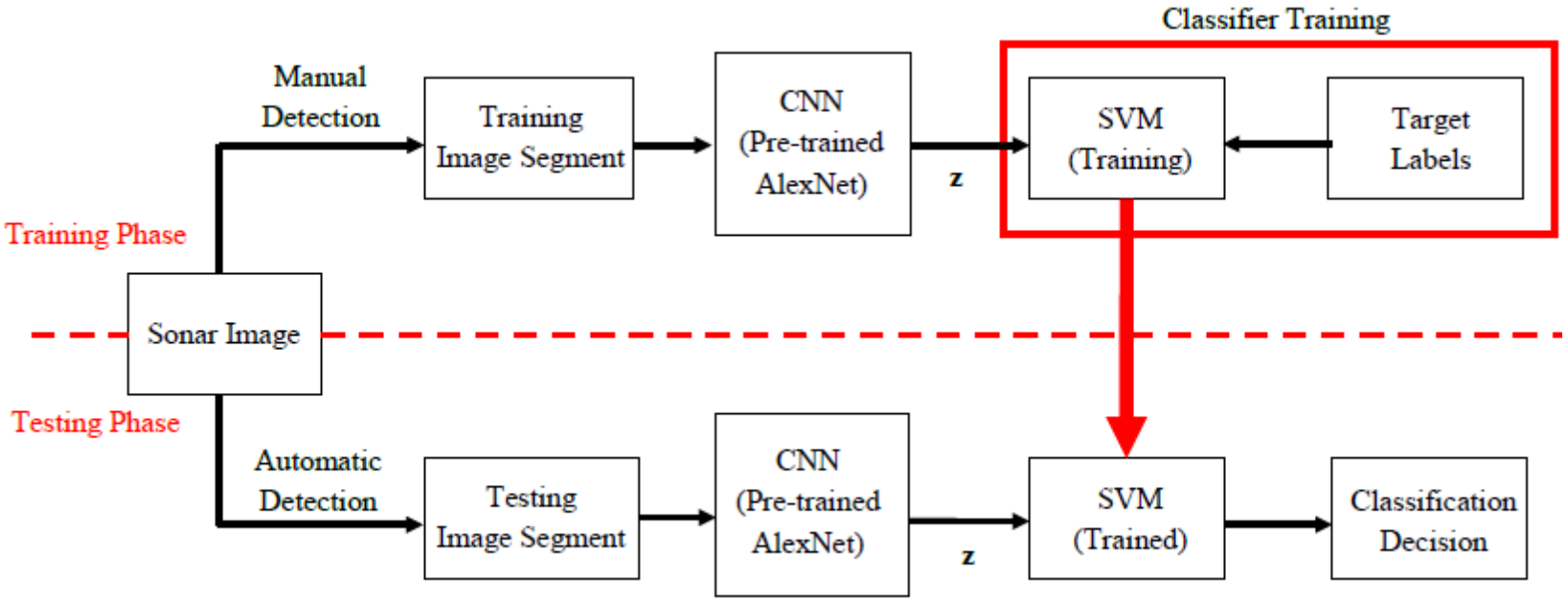
POOL Layer:

- This layer is a form of non-linear down-sampling.
- An example of the max pool with a 2×2 filter and stride $S = 2$



Automatic Target Recognition (ATR)

Automatic Target Recognition an Detection



- **Training Phase:** Train Support vector machine (SVM) classifier with Sonar Image I_s and true classification Y_T
- **Testing Phase:** Matched filter for image segmentation and CNN+SVM for target classification

Classification using Support Vector Machine (SVM)

•Support vector machine (SVM) is used for classification, which is expressed by

$$y = f_{SVM}(\mathbf{z}) = \Phi(\mathbf{w}^T\mathbf{z} + b)$$

where $\mathbf{z}$ is the high dimenstional feature.

The parameters $\mathbf{w}$ and b of the function f_{SVM} can be learned from the training dataset by solving the following optimization problem:

$$\min_{\mathbf{w},b,\xi} \frac{1}{2}\|\mathbf{w}\|^2 + c\sum_{n=1}^{N_{tr}} \xi_n$$

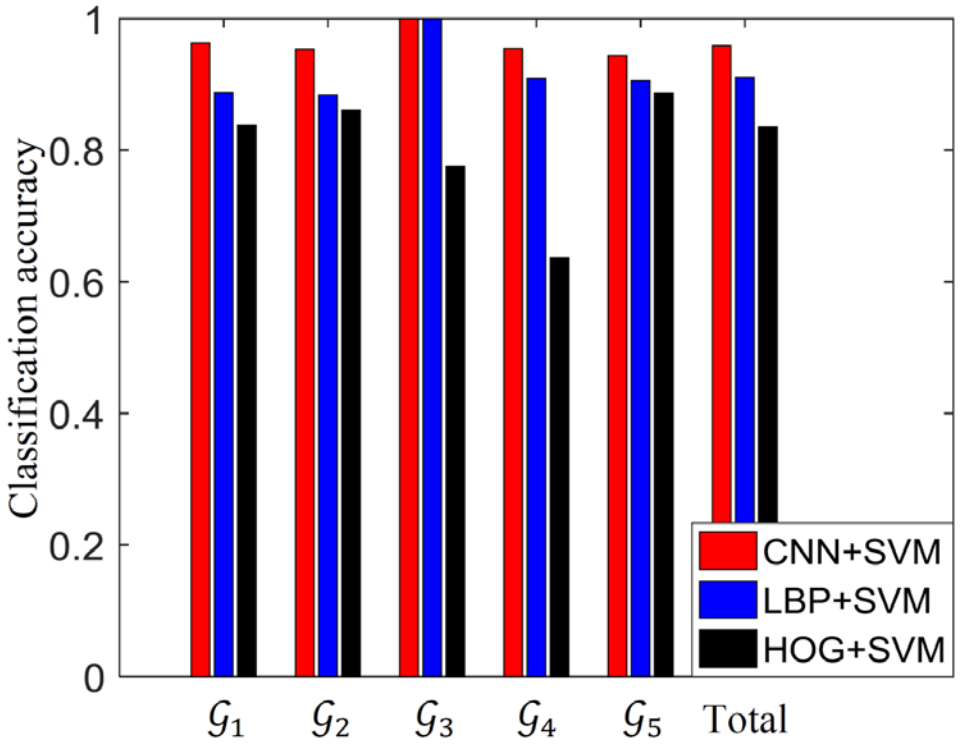
$$s.t. \quad \xi_n \geq 0, \quad y_n(\mathbf{w}^T\mathbf{z}_n + b) \geq 1 - \xi_n$$

$$n = 1,\ldots, N_{tr}$$

where $\boldsymbol{\xi} = [\xi_1,\ldots,\xi_n]$ are the slack variables. They represent the degree that each data sample lies inside the margin. The user-defined parameter $c>0$ controls the trade-off between the slack variable penalty and the margin.

Results

CNN Performance: Classification Accuracy



Legend:

- TP: True Positive
- FP: False Positive
- TN: True Negative
- FN: False Negative
- $\mathcal{G}_i$: Image Group i

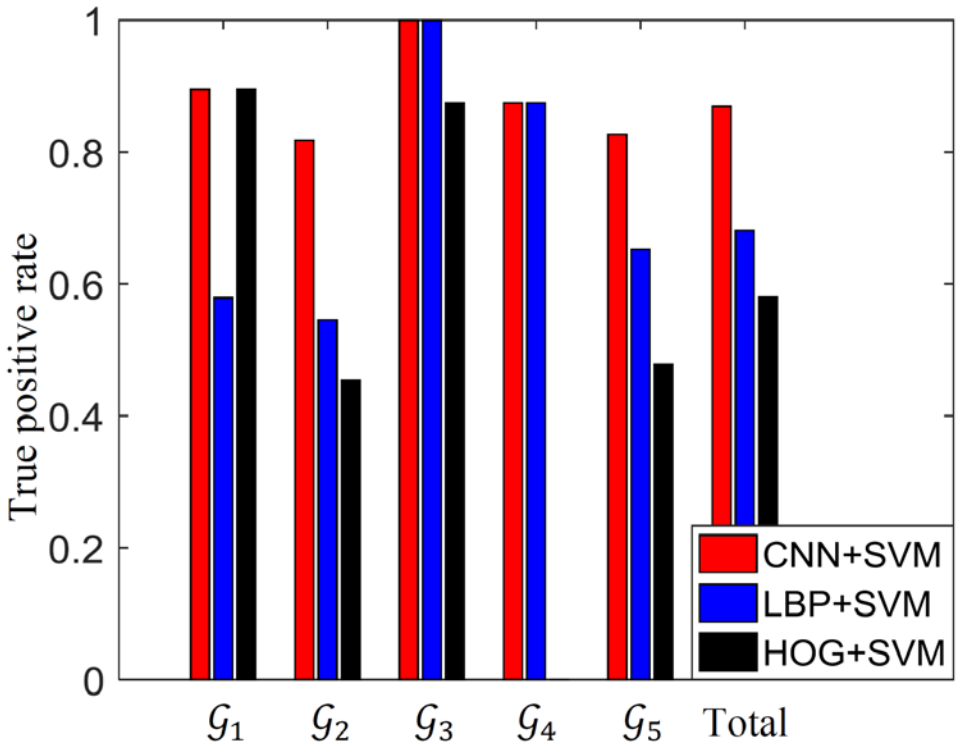
Classification Accuracy:

$$CA = \frac{n_{TP} + n_{TN}}{n_{TP} + n_{FN} + n_{FP} + n_{TN}}$$

Probability of detection (Matched Filter performance): 88.31%

* Zhu, P., Issacs, J., Fu, B., Ferrari, S., "Deep Learning Feature Extraction for Target Recognition and Classification in Underwater Sonar Images" 56th IEEE Conference on Decision and Control, Melbourne, Australia (Accepted).

CNN Performance: True Positive Rate



Legend:

- TP: True Positive
- FP: False Positive
- TN: True Negative
- FN: False Negative
- $\mathcal{G}_i$: Image Group i

True Positive Rate (TPR):

$$TPR = \frac{n_{TP}}{n_{TP} + n_{FP}}$$

Probability of detection (Matched Filter performance): 88.31%

* Zhu, P., Issacs, J., Fu, B., Ferrari, S., "Deep Learning Feature Extraction for Target Recognition and Classification in Underwater Sonar Images" 56th IEEE Conference on Decision and Control, Melbourne, Australia (Accepted).

Conclusions and Future work

Conclusions:

- Demonstrate automatic target recognition (ATR) and classification using deep learning features extraction for underwater sonar images
- Proposed method using a CNN+SVM structure for classification outperforms other methods such as HOG and LBP

Future Work:

- Improve algorithm robustness for sonar images taken in different environmental conditions
- Develop sonar-driven path planning for autonomous UUV