Contract Report: SAR MTI Detection and Tracking

Issue/Revision: 1/0
Task 7: TWSAR Classifier Training and Testing

Kai Wang

Prepared by:

MDA
75 Albert Street, Suite 412, Ottawa, Ontario, Canada K1P 5E7

Project Manager: Paul Lim 604-231-2556
Contract Number: W7714-115147/001/SV
Contract Scientific Authority: Dr. Bhashyam Balaji 613-998-2215

Contract Report

DRDC-RDDC-2017-C046

September 2015

Approved by

Dr. Bhashyam Balaji

Approved for release by

Abstract

This report documented the training and testing results of SVM classifier under Taks #7 of TWSAR Classifier Training and Testing.

The experimental SAR imagery datasets has been collected and processed to generate the metrics for each potential targets under test. These metrics were used to train the SVM classifier to perform human targets identification behind wall structures. The performance of SVM classifier has been documented.

This page intentionally left blank.

Executive summary

Contract Report: SAR MTI Detection and Tracking

Defence Research and Development Canada - Ottawa is investigating 3-D through all SAR imaging form an experimental L-band through -wall SAR prototype. This document summarizes the work done during Taks #7 of TWSAR Classifier Training and Testing.

The metrics for each potential targets has been calculated by using the existing Matlab code. These metrics are feed into SVM classifier to perform human targets detection behind wall structure. The SVM classifier has been trained, cross validated and tested by different sets of data combination from different wall structures. The accuracy of SVM's prediction was calculated as compared to the truth.

This page intentionally left blank.

Table of contents

List of tables

This page intentionally left blank.

1 TWSAR Data Processing

The experimental TWSAR datasets were used to calculate the corresponding metrics, which was used in the following process to train the SVM classifier for human targets detection behind different wall structures. Four different types of input datasets(as shown in Table 1) had been used to generate its corresponding metrics.

The metrics can be calculated automatically by utilized the existing MatLab code `"MainAMC.m"`. The Matalb code has been updated `"MainAMC_Kai_02.m"` to include all 4 different types of input datasets (Table 1). In addition, the power level from the new experimental datasets has been adjusted to match the original datasets. The hard coded metric adjustments has been applied to the new datasets as well.

For each type of input datasets (Table 1), 5 different metric values has been calculated for each potential targets under test. These metrics are the elevation of pixel's maximum intensity (PMI), azimuth maximum intensity, azimuth 3dB bandwidth, range 3dB bandwidth and the volume count.

Table 1: *Processing Parameter File Name*

Variable Name	Comments
SARdata	normalized magnitude of SAR image
newSAR	thresholded magnitude of SAR image
newSAR_noWall	no wall structure
newSAR_noWallnoExt	no wall structure, no exterior voxels

2 SVM Training and Testing

To automatically distinguish human targets behind wall structures from clutter measurements, we trained SVM as a 2 class-classifier. The input of the SVM was the metric values calculated in the previous data processing step. The output of the SVM was the class label corresponding to the input metrics.

In order for SVM to perform classification accurately, the model parameters of SVM has to be optimized. The processes includes SVM training, testing and cross validation.

The SVM training and testing MatLab code has been updated to `"trainingData_Kai_04.m"`

2.1 SVM Training

The training datasets were used to train the SVM. From the ”pattern” between input-output pairs, SVM ”learn” the pattern by adjusting its internal model parameters. In additional to the internal model parameters, external model parameters are available and accessible to fine-tuning the SVM. The resulting SVM model is training datasets dependent. Different training datasets will result in different model parameters (different SVM model).

The training datasets include the input metrics and its corresponding class labels. The input metrics has to be scaled to predefined range (normally to [0 1] or [-1 1]) to avoid greater metric values domination. This scaling factor has to be applied the testing datasets as well before SVM testing starts.

The training accuracy can be calculated by comparing the SVM predicted class labels to its targeting class labels (truth). The training accuracy defined in the existing code as:

$$\texttt{Validation Rate} = \frac{\texttt{number of correct prediction}}{\texttt{total number of prediction}} \tag{1}$$

Since the training datasets collected was out of balance, we manually calculated 2 additional accuracy measure: False Negative rate and False Positive Rate as:

$$\texttt{False Negative Rate} = \frac{\texttt{number of negative prediction (shoud be positive)}}{\texttt{total number of positive (truth)}} \tag{2}$$

$$\texttt{False Positive Rate} = \frac{\texttt{number of positive prediction (should be negative)}}{\texttt{total number of negative (truth)}} \tag{3}$$

Currently the datasets we have including over 99.9% of clutter and only less than 0.1% are human targets. As a result, even SVM ”predict” all the human targets wrong, it still have training accuracy over 99.9%. But with very hight false negative rate (nearly 100%) and very low false positive rate (near 0%).

This can be solved by manually adjusting the external model parameter ”weight” for each class label to ”balance” the data size between 2 classes. In addition, we can put a very high penalty scaler to ”punish” the False Negative rate.

2.2 SVM Cross Validation

Automatic cross validation is available within the existing SVM training code. The process is to sweep 2 external model parameters with very wide range and calculate its cross validation rate. This 2 dimensional model parameters can be plotted against its cross validation rate as contours. The best external model parameters can be determined by the highest cross validation rate.

Because the current SVM code use cross validation rate and our datasets is "unbalanced", we got over 99.9% validation rate even we have a very bad prediction (high Flase Negative Rate). To take the advantage of the automatic cross validation, it will require a major change of the existing SVM code to use false negative and/or false positive rate instead of cross validation rate currently in use. Furthermore, since we have to apply the "weight" for 2 class differently, the 2 dimensional contour will become 3 dimensional. As a result, we did this cross validation process manually.

2.3 SVM Testing

The testing datasets was used to test the SVM's classification accuracy. The testing datasets has the same data type and structure. The only difference is that SVM never "see" the testing datasets in the entire training process. As a result, the testing datasets will not change any model parameters of SVM model. The SVM model is independent of testing datasets but training datasets dependent.

3 SVM Classification Performance

The Performance of SVM classification has been documented. The datasets has been labelled as shown in Table 2 for simplicity. The datasets label 1-4 was corresponding to drywall building #1. The datasets label 5-10 was corresponding to drywall building #2. The datasets label 11-12 was corresponding to concrete building.

The SVM training and testing accuracy for different input datasets (Table 1) was shown in Table 3 to Table 6.

Table 2: *SVM Performance*

Dataset Name	Dataset Label	number of clutters	number of targets
ts_back_s8_04_dataMetrics.mat	1	3486	2
ts_back_s9_02_dataMetrics.mat	2	3486	2
ts_back_s10_02_dataMetrics.mat	3	3486	2
ts_back_s13_01_dataMetrics.mat	4	3486	1
ts_one_few_01_dataMetrics.mat	5	2877	5
ts_standB_01_dataMetrics.mat	6	3447	2
ts_standA_01_dataMetrics.mat	7	3447	2
ts_sitA_01_dataMetrics.mat	8	3447	1
ts_sitB_01_dataMetrics.mat	9	3447	1
ts_sitC_01_dataMetrics.mat	10	3447	1
br_humanA_01_dataMetrics.mat	11	3486	3
502_hfar_01_dataMetrics.mat	12	3486	1

Table 3: *SVM Performance (SARdata)*

Training Datasets	Testing Datasets	Training Time (sec)	Training Accuracy (%) Accuracy/FN/FP	Testing Accuracy (%) Accuracy/FN/FP
1-2	3-4	0.042	99.35/0/0.65	99.23/0/0.77
5-7	8-10	0.871	96.12/0/3.88	99.15/0/0.85
5-10	1-4	1.26	97.81/0/2.19	93.12/0/6.88
3-10	1-2	4.85	95.64/0/4.36	93.17/0/6.83
1-10	11-12	12.40	93.42/0/6.58	97.81/25/2.18

Table 4: *SVM Performance (newSAR)*

Training Datasets	Testing Datasets	Training Time (sec)	Training Accuracy (%) Accuracy/FN/FP	Testing Accuracy (%) Accuracy/FN/FP
1-2	3-4	0.021	99.97/0/0.03	99.91/33.3/0.07
5-7	8-10	0.701	93.90/0/6.11	98.45/0/1.55
5-10	1-4	0.888	98.54/0/1.46	97.14/14.29/2.86
3-10	1-2	8.71	95.08/0/4.92	91.54/0/8.47
1-10	11-12	14.49	91.43/0/8.57	98.02/25/1.97

Table 5: *SVM Performance (newSAR_noWall)*

Training Datasets	Testing Datasets	Training Time (sec)	Training Accuracy (%) Accuracy/FN/FP	Testing Accuracy (%) Accuracy/FN/FP
1-2	3-4	0.016	99.84/0/0.16	97.74/33.3/0.24
5-7	8-10	0.627	94.23/0/5.72	98.71/0/1.29
5-10	1-4	1.064	98.69/0/1.31	97.08/14.29/2.91
3-10	1-2	7.27	95.69/0/4.31	92.35/0/7.65
1-10	11-12	14.67	90.94/0/9.06	97.93/25/2.05

Table 6: *SVM Performance (newSAR_noWallnoExt)*

Training Datasets	Testing Datasets	Training Time (sec)	Training Accuracy (%) Accuracy/FN/FP	Testing Accuracy (%) Accuracy/FN/FP
1-2	3-4	0.024	91.81/0/0.19	99.71/33.3/0.27
5-7	8-10	0.666	94.05/0/5.95	98.73/0/1.27
5-10	1-4	1.31	97.61/0/2.39	94.34/14.29/5.65
3-10	1-2	6.93	95.64/0/4.36	92.49/0/7.51
1-10	11-12	15.66	90.91/0/9.09	97.63/25/2.35

References

[1] B. Chan, P. Sevigny and D.J. DiFilippo, ”Analysis of metrics for human detection behind walls using experimental 3-D synthetic aperture radar imagery,” *Scientific Report*, DRDC-RDDC-2014-R140, December 2014.

[2] C. Chang and C. Lin, ”LIBSVM – A Library for Support Vector Machines,” ¡http://www.csie.ntu.edu.tw/ cjlin/libsvm/¿