THE PENNSYLVANIA STATE UNIVERSITY

The Graduate School

Department of Electrical Engineering

THE EXPLOITATION OF MULTI-LOOK SYNTHETIC APERTURE RADAR AND INVERSE SYNTHETIC APERTURE RADAR IMAGES FOR NON-COOPERATIVE TARGET RECOGNITION

A Thesis in

Electrical Engineering

by

Scott Papson

© 2007 Scott Papson

Submitted in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

August 2007
The thesis of Scott Papson was reviewed and approved* by the following:

Ram M. Narayanan  
Professor of Electrical Engineering  
Thesis Advisor  
Chair of Committee

Robert T. Collins  
Associate Professor of Computer Science and Engineering

Randy L. Haupt  
Senior Scientist

William E. Higgins  
Distinguished Professor of Electrical Engineering

W. Kenneth Jenkins  
Professor of Electrical Engineering  
Head of the Department of Electrical Engineering

*Signatures are on file in the Graduate School
ABSTRACT

Synthetic aperture radar (SAR) and inverse synthetic aperture radar (ISAR) have proven capabilities for non-cooperative target recognition (NCTR) applications. Both sensing modalities have been able to provide operational information in a robust manner. As processing power and communication capabilities on the battlefield increase, new opportunities for exploiting SAR / ISAR systems emerge. Multiple looks of the same target can now be used to increase performance and allow systems to operate at maximum ranges. The research presented in this work outlines a variety of methods for utilizing SAR / ISAR images in the presence of multiple looks. Specific accomplishments include: the development of an automated segmentation method to extract information from SAR images; the development of novel image fusion rules to integrate data from small numbers of independent SAR / ISAR systems; the development of a persistence framework to enhance target features in large, aspect-varying datasets; the development of a shadow classification technique to classify multiple targets based only on their shadow features in SAR imagery; and operational analyses were performed to determine how the algorithms would perform in realistic scenarios. The algorithms are tested on a variety of canonical and real-world datasets.
# TABLE OF CONTENTS

LIST OF FIGURES .......................................................................................................................... vi

LIST OF TABLES ............................................................................................................................. xvi

ACKNOWLEDGEMENTS ...................................................................................................................... xviii

Chapter 1  Introduction .................................................................................................................. 1

1.1 Background .............................................................................................................................. 1

1.2 Data Exploitation ..................................................................................................................... 3

Chapter 2  Radar Imaging Data .................................................................................................... 9

2.1 Radar Imaging .......................................................................................................................... 9

2.2 Canonical Data ......................................................................................................................... 12

2.2.1 2.5-D Ray-Tracing Model ................................................................................................. 13

2.2.2 2.5-D Shadow Model .......................................................................................................... 19

2.2.3 Point Scattering Model with Occlusion .............................................................................. 24

2.3 ERAD’s Slicy Data .................................................................................................................. 29

2.4 DARPA’s MSTAR Data ........................................................................................................... 34

Chapter 3  Data Pre-Processing .................................................................................................. 37

3.1 Problem Overview ................................................................................................................... 37

3.2 Registration ............................................................................................................................... 38

3.2.1 Centroid Registration ......................................................................................................... 39

3.2.2 Jitter Analysis ...................................................................................................................... 44

3.3 Segmentation ............................................................................................................................ 53

3.3.1 Morphology ........................................................................................................................ 56

3.3.2 Mixture Modeling .............................................................................................................. 58

3.3.2.1 Expectation Maximization ............................................................................................... 60

3.3.2.2 Post-EM-processing ......................................................................................................... 62

3.3.2.3 Overview of Results ........................................................................................................ 64

3.3.3 Final Implementation .......................................................................................................... 79

3.4 Conclusions .............................................................................................................................. 89

Chapter 4  Highlight Analysis ....................................................................................................... 92

4.1 Problem Overview ................................................................................................................... 92

4.2 Evaluation Methods .................................................................................................................. 93

4.2.1 Receiver Operating Characteristics .................................................................................... 94

4.2.2 Target Power Scale Space ................................................................................................... 97

4.3 Implementations ....................................................................................................................... 100
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.3.1</td>
<td>Baseline Tests</td>
<td>100</td>
</tr>
<tr>
<td>4.3.2</td>
<td>Fusion Rules</td>
<td>101</td>
</tr>
<tr>
<td>4.3.2.1</td>
<td>Averaging Rules</td>
<td>103</td>
</tr>
<tr>
<td>4.3.2.2</td>
<td>Probabilistic Rules</td>
<td>104</td>
</tr>
<tr>
<td>4.3.2.3</td>
<td>Component Analysis</td>
<td>106</td>
</tr>
<tr>
<td>4.3.2.4</td>
<td>Wavelets</td>
<td>109</td>
</tr>
<tr>
<td>4.4</td>
<td>Results &amp; Analysis</td>
<td>120</td>
</tr>
<tr>
<td>4.4.1</td>
<td>Ray-tracing model experiments</td>
<td>120</td>
</tr>
<tr>
<td>4.4.2</td>
<td>MSTAR (linearity of the boundary)</td>
<td>130</td>
</tr>
<tr>
<td>4.4.3</td>
<td>MSTAR (multi-radar configuration analysis)</td>
<td>142</td>
</tr>
<tr>
<td>4.4.4</td>
<td>MSTAR (advanced rules)</td>
<td>152</td>
</tr>
<tr>
<td>4.4.5</td>
<td>Point-Scatterer data experiments</td>
<td>158</td>
</tr>
<tr>
<td>4.5</td>
<td>Conclusions</td>
<td>167</td>
</tr>
<tr>
<td>5.1</td>
<td>Problem Overview</td>
<td>170</td>
</tr>
<tr>
<td>5.2</td>
<td>Implementation</td>
<td>173</td>
</tr>
<tr>
<td>5.3</td>
<td>Results &amp; Analysis</td>
<td>178</td>
</tr>
<tr>
<td>5.3.1</td>
<td>Visualization</td>
<td>178</td>
</tr>
<tr>
<td>5.3.2</td>
<td>Classification</td>
<td>188</td>
</tr>
<tr>
<td>5.4</td>
<td>Conclusions</td>
<td>197</td>
</tr>
<tr>
<td>6.1</td>
<td>Problem Overview</td>
<td>199</td>
</tr>
<tr>
<td>6.2</td>
<td>Classification Structures</td>
<td>200</td>
</tr>
<tr>
<td>6.2.1</td>
<td>Procrustes Analysis</td>
<td>200</td>
</tr>
<tr>
<td>6.2.1.1</td>
<td>Procrustes Overview</td>
<td>200</td>
</tr>
<tr>
<td>6.2.1.2</td>
<td>Minimum Distance Classifier</td>
<td>203</td>
</tr>
<tr>
<td>6.2.1.3</td>
<td>Gaussian Likelihood Classifier</td>
<td>205</td>
</tr>
<tr>
<td>6.2.1.4</td>
<td>K-Nearest Neighbor Classifier</td>
<td>206</td>
</tr>
<tr>
<td>6.2.1.5</td>
<td>Multilayer Perceptron Neural Network</td>
<td>209</td>
</tr>
<tr>
<td>6.2.2</td>
<td>Hidden Markov Modeling</td>
<td>211</td>
</tr>
<tr>
<td>6.2.2.1</td>
<td>Overview</td>
<td>211</td>
</tr>
<tr>
<td>6.2.2.2</td>
<td>Chain Code Representation</td>
<td>213</td>
</tr>
<tr>
<td>6.2.2.3</td>
<td>Generating Target Models</td>
<td>214</td>
</tr>
<tr>
<td>6.2.2.4</td>
<td>Shadow Classification Structure</td>
<td>216</td>
</tr>
<tr>
<td>6.2.2.5</td>
<td>Model Selection</td>
<td>216</td>
</tr>
<tr>
<td>6.2.2.6</td>
<td>Multiple Looks</td>
<td>220</td>
</tr>
<tr>
<td>6.2.3</td>
<td>Integration with Scattering Center Classifiers</td>
<td>221</td>
</tr>
<tr>
<td>6.3</td>
<td>Small-Scale Tests</td>
<td>223</td>
</tr>
<tr>
<td>6.3.1</td>
<td>Procrustes Analysis</td>
<td>224</td>
</tr>
<tr>
<td>6.3.2</td>
<td>Classification</td>
<td>229</td>
</tr>
<tr>
<td>6.3.3</td>
<td>Markov Modeling</td>
<td>232</td>
</tr>
<tr>
<td>Section</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>---------</td>
<td>-------</td>
<td>------</td>
</tr>
<tr>
<td>6.3.3.1</td>
<td>General Performance</td>
<td>232</td>
</tr>
<tr>
<td>6.3.3.2</td>
<td>Variable Resolution</td>
<td>235</td>
</tr>
<tr>
<td>6.3.3.3</td>
<td>Variable Depression Angle</td>
<td>238</td>
</tr>
<tr>
<td>6.4</td>
<td>Large-Scale Tests</td>
<td>240</td>
</tr>
<tr>
<td>6.4.1</td>
<td>Implementation</td>
<td>240</td>
</tr>
<tr>
<td>6.4.2</td>
<td>Model Selection and Ensemble Averaging</td>
<td>249</td>
</tr>
<tr>
<td>6.4.3</td>
<td>Classification Results</td>
<td>255</td>
</tr>
<tr>
<td>6.4.3.1</td>
<td>Data</td>
<td>255</td>
</tr>
<tr>
<td>6.4.3.2</td>
<td>Single-look Classification</td>
<td>256</td>
</tr>
<tr>
<td>6.4.3.3</td>
<td>Multi-look Classification</td>
<td>259</td>
</tr>
<tr>
<td>6.4.4</td>
<td>Integration</td>
<td>261</td>
</tr>
<tr>
<td>6.5</td>
<td>Shadow Projections</td>
<td>263</td>
</tr>
<tr>
<td>6.5.1</td>
<td>Introduction</td>
<td>263</td>
</tr>
<tr>
<td>6.5.2</td>
<td>Point correspondences</td>
<td>267</td>
</tr>
<tr>
<td>6.5.2.1</td>
<td>Implementation</td>
<td>267</td>
</tr>
<tr>
<td>6.5.2.2</td>
<td>Results</td>
<td>272</td>
</tr>
<tr>
<td>6.5.3</td>
<td>Radar geometry</td>
<td>278</td>
</tr>
<tr>
<td>6.6</td>
<td>Conclusions</td>
<td>284</td>
</tr>
<tr>
<td>7.1</td>
<td>Further Research / Investigations</td>
<td>287</td>
</tr>
<tr>
<td>7.1.1</td>
<td>Applications in Unmanned Aerial Vehicles</td>
<td>287</td>
</tr>
<tr>
<td>7.1.2</td>
<td>Applications to Maritime Targets</td>
<td>290</td>
</tr>
<tr>
<td>7.2</td>
<td>Goals and Accomplishments</td>
<td>294</td>
</tr>
<tr>
<td>References</td>
<td></td>
<td>300</td>
</tr>
<tr>
<td>Appendix</td>
<td></td>
<td>308</td>
</tr>
<tr>
<td>A.1</td>
<td>ERADS Slicy Header</td>
<td>308</td>
</tr>
<tr>
<td>A.2</td>
<td>MSTAR Header</td>
<td>309</td>
</tr>
<tr>
<td>A.3</td>
<td>Small Scale Test Data</td>
<td>310</td>
</tr>
</tbody>
</table>
LIST OF FIGURES

Figure 1-1: Image-level data fusion ................................................................. 4
Figure 1-2: Feature-level and classifier-level fusion ........................................ 5
Figure 2-1: Target-radar motion allows for two-dimensional imaging .......... 10
Figure 2-2: Linear image formulation method .................................................. 12
Figure 2-3: Setup of the 2.5-D looking radar model ........................................ 14
Figure 2-4: Initial test target database .............................................................. 15
Figure 2-5: Simulated tank-like objects ............................................................ 16
Figure 2-6: Simulated imagery from tank-like objects ...................................... 17
Figure 2-7: Pixel distribution as a function of radar height ............................... 18
Figure 2-8: Image sub-regions ........................................................................ 19
Figure 2-9: Illustration of 2.5-D ray-tracing shadow model .............................. 20
Figure 2-10: An example ray intersecting the target ........................................ 23
Figure 2-11: Aspect dependent shadows for a two-tiered square object .......... 24
Figure 2-12: Setup for point scattering model with occlusion .......................... 25
Figure 2-13: Point scatterer model with points of equal height ....................... 27
Figure 2-14: Point scatterer model with points of unequal height .................. 27
Figure 2-15: Illustration of long, thin shadowing properties ............................ 28
Figure 2-16: Illustration of short, thick shadowing properties ......................... 28
Figure 2-17: Point scatter target model ............................................................ 29
Figure 2-18: Slicy target ................................................................................. 30
Figure 2-19: X-Band RCS plots ..................................................................... 31
Figure 2-20: Data for a single ISAR image .................................................... 32
Figure 2-21: Processing of ERADS data ......................................................... 33
Figure 2-22: Multiple polarizations for a single aspect angle.................................33
Figure 2-23: Varied image sizes for X-Band data ..................................................34
Figure 2-24: T-72 SAR chips for varying aspects ...................................................36
Figure 3-1: Typical SAR image [31] .....................................................................39
Figure 3-2: Alignment using target centroid..........................................................40
Figure 3-3: Misadjustment of the centroid for two threshold levels.......................41
Figure 3-4: Centroid position during registration ...................................................42
Figure 3-5: Resulting mean images of X-Band data ..............................................42
Figure 3-6: Method of registration for the entire dataset from a single template ......45
Figure 3-7: Registration results for target 1 under low noise ...............................46
Figure 3-8: Registration performance for target 1 under low noise.......................47
Figure 3-9: Registration results for target 1 under high noise ..............................48
Figure 3-10: Registration performance for target 1 under low noise.....................48
Figure 3-11: Comparisons of absolute errors.......................................................49
Figure 3-12: Error as a function of configuration...................................................50
Figure 3-13: De-jitter in the SLICY data...............................................................51
Figure 3-14: Normalized cross-correlation for translation corrections..................52
Figure 3-15: Method of registration for an entire dataset using multiple templates...53
Figure 3-16: SAR images of T-72 illustrating variable region statistics ................54
Figure 3-17: Detection of shadow region .............................................................58
Figure 3-18: EM segmentation algorithm.............................................................59
Figure 3-19: Sample result of class labeling.........................................................65
Figure 3-20: Processing of the shadow region.......................................................66
Figure 3-21: Processing of the highlight region ....................................................66
Figure 3-22: Final labeling...........................................................................................67
Figure 3-23: Sample segmentations.............................................................................68
Figure 3-24: Error images from EM segmentation......................................................69
Figure 3-25: Performance metric .............................................................................70
Figure 3-26: Histogram for a manually segmented target .......................................72
Figure 3-27: Results for target segmentation.............................................................72
Figure 3-28: Results for target segmentation.............................................................73
Figure 3-29: Performance of 1D estimations...............................................................74
Figure 3-30: Class assignments for various model orders.........................................75
Figure 3-31: Number of test instances that had an overlap factor over the desired level.......................................................................................................................76
Figure 3-32: Mean overlap factor .............................................................................77
Figure 3-33: Edge perturbations in the extracted shadow .......................................80
Figure 3-34: Overview of segmentation method.......................................................81
Figure 3-35: Diffusion process ...............................................................................82
Figure 3-36: Smoothing over multiple iterations.......................................................83
Figure 3-37: Rapid diffusion.....................................................................................83
Figure 3-38: Effects of smoothing in the turret region of the T-72 .........................84
Figure 3-39: Shadow segmentation results for various combinations of diffused input and output probabilities. (diffused input / diffused output).................85
Figure 3-40: Effects of smoothing on the posterior probability .............................86
Figure 3-41: Effect of spatial variance constraints on shadow class .....................87
Figure 3-42: Shadow segmentation for various MSTAR targets.............................89
Figure 4-1: Areas of interest for image evaluation .................................................96
Figure 4-2: Concept of multiple operating points for an intensity-valued image ....96
Figure 4-3: Concept of ROC curve .................................................................................................97
Figure 4-4: Rectangular scale-spaces kernels for 3 arbitrary point scatterers ..................99
Figure 4-5: Linear weighting scheme ..........................................................................................104
Figure 4-6: 2-D wavelet decomposition [51] ..............................................................................110
Figure 4-7: Daubechies 3 wavelet & corresponding filters ..................................................111
Figure 4-8: Mapping procedure for wavelet fusion .................................................................112
Figure 4-9: 2-D wavelet reconstruction [51] ..............................................................................112
Figure 4-10: Binary fusion rules ...............................................................................................113
Figure 4-11: Effects of fusion .....................................................................................................115
Figure 4-12: Disparity/Morphology as a measure of rule assignment ..................................117
Figure 4-13: Data pre-processing overview .............................................................................118
Figure 4-14: Data fusion rules .................................................................................................119
Figure 4-15: Data post-processing ...........................................................................................120
Figure 4-16: Pixel distributions ...............................................................................................121
Figure 4-17: ROC for 4 total sub-regions ..............................................................................122
Figure 4-18: Sub-block distributions .......................................................................................122
Figure 4-19: ROC for 16 total sub-regions .............................................................................124
Figure 4-20: Sub-block distributions .......................................................................................124
Figure 4-21: Images with various SNR values .......................................................................126
Figure 4-22: Improvement in the probability of detection for various SNR values .... 126
Figure 4-23: Sample images for data being fused .................................................................127
Figure 4-24: Improvement in probability of detection for various inter-radar angles ..............128
Figure 4-25: Improvements as a function of false alarm for target 2 .................................129
Figure 4-26: Pixel distributions ...............................................................................................131
Figure 4-27: Fusion analysis

Figure 4-28: Improvement in probability of detection

Figure 4-29: 2D pixel distributions

Figure 4-30: Decision boundaries

Figure 4-31: Theoretical ROC curves

Figure 4-32: Input test images

Figure 4-33: Decision boundaries

Figure 4-34: ROC for linear and conditional risk boundaries

Figure 4-35: Evaluation of weighted averaging

Figure 4-36: Two-look fusion with images of equal information content

Figure 4-37: Two-look fusion with images of unequal information content

Figure 4-38: Evaluation of weighted average for a two-look system

Figure 4-39: Two-look fusion with images of severely disparate information content

Figure 4-40: Improvement in the probability of detection as a function on intra-radar angle for a 2-look system

Figure 4-41: Analysis framework for a 3-look systems

Figure 4-42: Neyman-Pearson criteria

Figure 4-43: Generating configuration diversity for the 3-look analysis

Figure 4-44: Configuration dependent performance measures

Figure 4-45: Equally spaced radar configuration #1

Figure 4-46: Equally space radar configuration #2

Figure 4-47: Varying performance for equally spaced radar configurations

Figure 4-48: Outline of genetic algorithm implementation

Figure 4-49: Evolution of the genetic algorithm
Figure 4-50: Performance evaluation.................................................................151
Figure 4-51: Mean performance of multi-look systems ........................................151
Figure 4-52: Persistence weighted average........................................................153
Figure 4-53: Test input images ...........................................................................153
Figure 4-54: Component analysis......................................................................154
Figure 4-55: Evaluation of fusion.................................................................155
Figure 4-56: Test input images ...........................................................................156
Figure 4-57: Component analysis of ICA..........................................................156
Figure 4-58: Evaluation of fusion.................................................................157
Figure 4-59: Fusion of set A ............................................................................159
Figure 4-60: Fusion of set B ............................................................................159
Figure 4-61: Variable aspect for fusion set A....................................................160
Figure 4-62: Variable aspect for fusion set B....................................................161
Figure 4-63: Improvement for random point configurations..............................162
Figure 4-64: Single test image with occlusion point ..........................................163
Figure 4-65: Wavelet rules applied to test image with minimal occlusions ..........163
Figure 4-66: Generating wavelet map...............................................................164
Figure 4-67: Morphological wavelet fusion mapping ........................................165
Figure 4-68: Test images with multiple occlusion points.....................................166
Figure 4-69: Morphological wavelet fusion mapping ........................................166
Figure 5-1: Typical phenomenology for a target/radar configuration..................170
Figure 5-2: Illustration of small and large scale angular perturbations ...............171
Figure 5-3: Radar image ..................................................................................173
Figure 5-4: Persistence modeling......................................................................177
Figure 5-5: Initial implementation of persistence model.............................................178
Figure 5-6: Full data and mappings for D7...............................................................178
Figure 5-7: Intensity plots for all pixel locations......................................................179
Figure 5-8: Coefficients for cubic persistence model..............................................180
Figure 5-9: Modeled persistence data and corresponding error..............................181
Figure 5-10: Maximum modeling errors..................................................................181
Figure 5-11: Persistence model images of D7 ..........................................................182
Figure 5-12: Persistence model images of T-72 ......................................................183
Figure 5-13: Eigenvalues for T-72 datasets..............................................................184
Figure 5-14: Dimensionality of the angularly dependent data..................................185
Figure 5-15: Dimensionality of the persistence data ..............................................186
Figure 5-16: PCA compilation images ....................................................................187
Figure 5-17: Templates..............................................................................................189
Figure 5-18: Thresholded templates ........................................................................190
Figure 5-19: Classification performance as a function of the number of templates ....191
Figure 5-20: Classification performance as a function of the number of neighbors ...192
Figure 5-21: Average classification rate on the training data using a single template.................................................................................................................194
Figure 5-22: PCA basis function classification structure .........................................196
Figure 6-1: Key point feature selection .................................................................201
Figure 6-2: Minimum distance classifier ..................................................................204
Figure 6-3: Shape descriptors in the shape space ....................................................205
Figure 6-4: Template generation for K-nearest neighbors.......................................207
Figure 6-5: K-nearest neighbors in solitary shape spaces .......................................208
Figure 6-6: K-nearest neighbors in a multi-space reference....................................209
Figure 6-7: Multilayer Perceptron neural network classification structure ..........210
Figure 6-8: Statistical shadow modeling.................................................................212
Figure 6-9: Chain coding .......................................................................................214
Figure 6-10: Integration of classifiers ....................................................................222
Figure 6-11: Point selection ...................................................................................224
Figure 6-12: Unregistered segmentation points ....................................................224
Figure 6-13: Registered shadows and the corresponding mean shapes ...............226
Figure 6-14: Eigenvalues for shadow points .........................................................227
Figure 6-15: 15-dimensional shadow shape space ...............................................227
Figure 6-16: First mode of variation ....................................................................228
Figure 6-17: Second mode of variation .................................................................229
Figure 6-18: Classification rates using leave-one-out testing...............................230
Figure 6-19: Markov transition matrices ...............................................................233
Figure 6-20: HMM likelihood outputs ..................................................................235
Figure 6-21: Sample image resolutions ...............................................................236
Figure 6-22: Performance analysis for a fixed training scale factor .....................237
Figure 6-23: Performance analysis for a matched training scale factor ...............237
Figure 6-24: Performance analysis for a fixed 15° depression angle .................238
Figure 6-25: Performance analysis for a matched depression angle scale factor ...239
Figure 6-26: Initialization of the HMM probability matrices ..................................242
Figure 6-27: HMM performance as a function of the number of hidden states .....243
Figure 6-28: Performance results for varying number of hidden states .............245
Figure 6-29: Performance analysis .......................................................................248
Figure 6-30: Evaluation metrics on validation data for model selection .............249
Figure 6-31: Evaluation metrics on validation data for model selection .................. 250
Figure 6-32: Class outputs ..................................................................................... 257
Figure 6-33: Target color scale ............................................................................. 257
Figure 6-34: Classification rate for T72 ............................................................... 258
Figure 6-35: Classification rate of multi-look scenarios ....................................... 259
Figure 6-36: Performance of a 2-look system as a function of configuration ......... 261
Figure 6-37: Confusion matrices ........................................................................... 262
Figure 6-38: Projective transformation ................................................................. 264
Figure 6-39: Shadow object composition ............................................................. 268
Figure 6-40: Shadow projection ............................................................................ 269
Figure 6-41: Illustration of 3-to-1 point correspondence ....................................... 271
Figure 6-42: Evaluation of homographic transformation as a function of aspect ...... 272
Figure 6-43: Shadows of varying depression angles ............................................. 273
Figure 6-44: Point correspondences ..................................................................... 273
Figure 6-45: Shadow transformations ................................................................. 274
Figure 6-46: Test object ....................................................................................... 276
Figure 6-47: Overlap factor as a function of aspect for simple target .................... 276
Figure 6-48: Overlap factor as a function of aspect for tank-like target .................. 277
Figure 6-49: Projection of complex target with hidden features ......................... 278
Figure 6-50: Shadow geometry .......................................................................... 279
Figure 6-51: Variable shadow lengths from varying depression angles ............... 280
Figure 6-52: Cross-range geometry .................................................................... 281
Figure 6-53: Reference SAR shadow at 15° depression ...................................... 283
Figure 6-54: Projection of 15° shadow to varied depressions .............................. 283
Figure 7-1: Directed UAV flight paths and sensing locations.................................288
Figure 7-2: UAV flight trajectories.........................................................................289
Figure 7-3: Data capture framework for maritime ISAR........................................291
Figure 7-4: ISAR images using all available data [77]...........................................292
Figure 7-5: ISAR fusion images using sub-apertures and fusion [77]......................293
Figure 7-6: Variable rotation of ISAR for a ship target [78]....................................294
Figure A1-1: SAR chips of T-72 ............................................................................310
Figure A1-2: SAR chips of BTR60 ........................................................................311
LIST OF TABLES

Table 2-1: Slicy experimentation parameters ..............................................................30

Table 2-2: Target and image characteristics for the MSTAR database at 2 depression angles..................................................................................................35

Table 3-1: Performance statistics.................................................................................71

Table 3-2: Initialization parameters for the EM algorithm...........................................87

Table 3-3: Trim statistics for shadow segmentation....................................................88

Table 5-1: Global simulation parameters.....................................................................189

Table 5-2: Classification using minimum distance classifier with persistence templates...........................................................................................................195

Table 5-3: Classification in reduced template spaces....................................................197

Table 6-1: Classification comparison ..........................................................................231

Table 6-2: Training time ..............................................................................................243

Table 6-3: Model performance for 15 hidden states....................................................244

Table 6-4: Model performance for 25 hidden states....................................................245

Table 6-5: Vernacular for KS testing...........................................................................247

Table 6-6: Type Caption Here .....................................................................................252

Table 6-7: Loud-mouth models classification results...................................................253

Table 6-8: Single-target cautious models classification results....................................253

Table 6-9: Global, single-target cautious classification results ....................................253

Table 6-10: Global, single-target loud-mouth ............................................................254

Table 6-11: Ensemble classification rate ......................................................................255

Table 6-12: Database trim statistics.............................................................................256

Table 6-13: Classification rates for multi-look scenarios ............................................260

Table 6-14: Joint probability space of shadow-only and highlight classification ......263
Table 6-15: RANSAC performance

275
ACKNOWLEDGEMENTS

First, I would like to thank our sponsors M.A. Pollock and G.D. McNeal of the Office of Naval Research (ONR) for their support and feedback under Contract #N00014-04-1-0307. I have appreciated the guidance and support from my advisor Ram Narayanan throughout the entire research process. I would also like to thank all the teachers and professors from my academic career for helping me get to where I am today. Finally, I would like to thank the High Performance Computing Group of the Pennsylvania State University for use of their computation resources.
Chapter 1

Introduction

1.1 Background

Although advancing rapidly, a challenging problem for current surveillance and weapon systems is the rapid and reliable detection of air, ship, and ground targets at maximum ranges. Imaging at maximum ranges is important for both the safety of the operators and the safe return of the sensing hardware. This scenario presents a unique challenge in that the data is not necessarily as reliable as in other mission scenarios. It is therefore essential that techniques developed for this environment are robust and efficiently exploit all available information [1].

Such opportunities for exploiting systems emerge because processing power and communication capabilities on the battlefield are increasing. The deployment of multiple sensing networks or the availability of multiple sensing modalities is quite common. Moreover, multiple looks of the same target can be obtained from different aspect angles to allow for different views of the same target. The following dissertation outlines various techniques that take advantage of multiple images of a single target. Specifically, synthetic aperture radar (SAR) and inverse synthetic aperture radar (ISAR) are used to image, classify, and characterize various targets.

Solutions to the target location and identification problem can be broadly categorized into one of two main areas – cooperative techniques and non-cooperative
techniques. Identification friend-or-foe (IFF) is one example of a cooperative technique currently employed in many radar systems. IFF operates by allowing friendly craft to respond to an authorized interrogation using a set of predetermined coded signals. When the correct signals are exchanged, a friendly aircraft can be positively identified; however neutral and hostile targets cannot be identified via this method. In such cases, non-cooperative target recognition (NCTR) techniques are employed. NCTR techniques are primarily based on comparing the information available for a given target with a database of known identifications. SAR and ISAR are two techniques that have proven capabilities for NCTR [2] [3].

Although the direct goal of NCTR is automated target recognition (ATR) there are a number of functions beyond classification that the algorithms should aim to address. A system operator’s visual display of data from multiple sensors is a simple example where fusion employed for NCTR is not directly related to identification. Moreover, being able to visually or automatically resolve and detect various features within an image is also logical extension of NCTR. An example would be determining the articulation of a tank’s turret; although the ID of the target may be known, there is still useful sensing information that can be obtained. The algorithms investigated and developed in this work attempt to solve problems encompassing many of the various aspects of NCTR for multi-location SAR / ISAR systems.
1.2 Data Exploitation

In cases where a single sensing modality images a target multiple times, in this case radar, there are a number methods in which the data can be integrated [4] [5]. Homogenous data-level fusion is a distinction often used to characterize methods that would concurrently operate directly on the returns from the various radar systems. A higher-level approach would be to operate on the radar images after processing from the individual systems; this is one of the techniques proposed. Finally, it is proposed to perform integration of the data in a feature or classifier-space as well. Fusion systems can also operate in high levels of the mission-space as task-level / operational fusion techniques; these high-level techniques are beyond the scope of this research.

One of the main advantages of performing image level fusion comes from the fact that SAR / ISAR imaging systems have well established and proven algorithms. For example, many SAR systems account for a non-linear flight path or target motion. A data-level approach would require incorporating these concepts into a multi-look system. An image-level approach can concentrate on the multi-look features since the radar systems output images that are already corrected for these non-linearities. Additionally, image-level approaches allow for a clearer extrapolation to other sensing modalities. Many image-level algorithms can easily be ported between SAR and ISAR images, but data-level algorithms cannot.

The idea of multi-location imaging using an image-level fusion approach is illustrated in Figure 1-1 below. In this approach, \( N \) radar images are generated for \( N \) radar locations. The method of image generation is tied to the individual radar systems.
The multiple input images are then fused together to obtain a single composite image that ideally reflects the information of all $N$ original images.

---

Figure 1-1: Image-level data fusion

The figure shows how individual SAR systems are used in conjunction with image-level fusion to reduce the data data-space from $N$ individual images to a single fusion image.

Not all applications of SAR / ISAR images can be performed with an image-level approach, however. For example, if a target is being classified by the radar image it may be more prudent to perform a feature or classifier level fusion. This is precisely what is proposed for the shadow features described in Chapter 6. In that context, an image-level fusion is not meaningful; a classifier-level approach however allows multiple images to be used to enhance the probability of correct classification. Figure 1-2 illustrates where feature-level and classifier-level fusion fit into the NCTR paradigm.
Some of the methods that have been previously investigated for radar fusion can be broadly categorized into: broadband multi-aspect processing [6] [7] [8] [9]; multi-aspect pattern classification [10] [11]; and statistical analysis [12]. Broadband multi-aspect processing focuses on generating an image by minimizing the image dependency on the look angle of individual radars. Typically, the returns from extended targets are highly dependent on aspect angle as the images exhibit the effects of highlighting and shadowing. By steering the main beam of a synthesized aperture or collecting data from multiple locations, an image can be generated that minimizes these unwanted effects; this additional information should allow for target identification with a suitable confidence level. These techniques are instances of data-level approaches.

Figure 1-2: Feature-level and classifier-level fusion

The image shows how multiple radar images can be used for classification using feature-level and classifier level fusion. The output of these systems is a classification result that should be more robust because of the presence of multiple input images.
Multi-aspect pattern classification and statistical analyses typically focus on feature and classifier-level fusion. A feature that is most commonly used in radar images is the location and strengths of the targets’ various scattering centers. The scattering centers of the test target are extracted from multiple images and subsequently classified. Hidden Markov models (HMMs) and predictive networks are two forms that have been effectively used for such classification problems [13] [14] [15]. These techniques involve learning a mathematical model from a set of training data. Other classification techniques use a physical target model and aspect-dependencies to assess the test data [16] [17] [18] [19]. More specifics of these and similar research topics will be subsequently discussed in the appropriate chapters.

The proposed techniques operate in a scenario where multiple images of a single target have been obtained. The goal of the algorithms is to therefore advantageously exploit the available data; the problem is one of data integration, fusion, and utilization. These algorithms should classify the images, resolve features in the image, or otherwise enhance the overall process of NCTR. The breakdown of the remaining chapters is given below:

**Chapter 2 – Data**: The main focus of this chapter is SAR and ISAR data. A brief overview of the radar systems is given, and the imaging procedures are outlined. Next, the datasets that were used for experimentation are presented. The datasets offer a mixture of canonical data, laboratory data, and real-world data.

**Chapter 3 – Data Pre-Processing**: The pre-processing chapter deals with registration and segmentation. Both steps are required before the fusion or classification
algorithms can be applied. The algorithms are tailored for the particular types of data being investigated.

**Chapter 4 – Highlight Analysis:** The highlight analysis chapter deals with the fusion of small numbers of individual radar images. More specifically, processing focuses on evaluating the scattering center information within the images. Both traditional and novel fusion rules are tested over a variety of datasets. The performance of fusion is investigated as a function of radar configurations. The development of image evaluation metrics is also given in this chapter.

**Chapter 5 – Persistence Modeling:** In this chapter, a model is developed to analyze and fuse large numbers of images. The goal of the persistence model is to transform the aspect dependencies of the images to target specific features. The model is implemented and tested for the purposes of visualization and classification. The visualization problem amounts to displaying large amounts of information to a user in a logical and useful way. Secondly, the persistence model is tested for use as a classification structure that utilizes a reduced number of training templates.

**Chapter 6 – Shadow Analysis:** In the presence of multiple angular looks, the shadow features within SAR images are shown to be features that can be reliably used for classification. Multiple classification structures are tested on a reduced set of targets. The final classification structure is more rigorously tested over its range of operating parameters on a larger dataset, and the performance is analyzed as a function of radar configuration and target type. Finally, the integration of the shadow classification technique with traditional classification techniques is examined.
Chapter 7 – Conclusions: The conclusion chapter offers a compilation of the important results from the previous chapters. Recommendations for implementations of the algorithms and further research are also presented.
2.1 Radar Imaging

Both SAR and ISAR systems use the relative target-radar motion to actively image a target. The distinction between SAR and ISAR is the motion that induces the synthetic aperture. For SAR systems, the radar is moving and the target is stationary. The motion of the radar allows for multiple collections and multiple views of the scene in a narrow aspect. This information can subsequently be processed to form an image. In ISAR, the radar is stationary; it is the motion of a maneuvering target that allows for the formulation of an image. Varying Doppler shifts are induced from the target’s rotational motion for individual point scatterers [20]. It is the motion of the target, not the radar, which allows for two-dimensional imaging in ISAR. Simple representations of SAR and ISAR systems are given in Figure 2-1.
SAR and ISAR are both fairly established methods for imaging targets. Specifically, there has been significant work done in the field of signal processing for both of these systems. In realistic scenarios, a SAR system’s acquisition path will not be straight as theoretically developed. The targets within the scene may be moving as well. Many researchers have developed techniques for various forms of motion compensation [21][22]. For ISAR, rotating targets will not have uniform pitch, roll, or yaw and targets are also likely to have a translation component. ISAR algorithms correct for these effects in an attempt to yield a more accurate two-dimensional representation of a target. For both systems, super-resolution algorithms have also been developed to optimize the necessary signal transformations [23][24][25][26]. All of the above techniques operate on the radar signals or are directly incorporated into the radar hardware; these specific

---

**Figure 2-1:** Target-radar motion allows for two-dimensional imaging

In a SAR system the radar is in motion and the target is stationary. In an ISAR system the radar is stationary and the target has a rotational motion component.
phenomena and algorithms are beyond the scope of this work. However, it is still important to have a fundamental understanding of how these systems generate the respective imagery. The remainder of this section outlines the traditional linear transformations used in SAR/ISAR systems.

The matrix in Eq. 2.1 represents the radar data matrix, \( D \). The data matrix is a representation of the returns from multiple frequency bins over a finite set of azimuthal angles.

\[
D = \begin{bmatrix}
    d_{11} & d_{12} & \cdots & d_{1N} \\
    d_{21} & \ddots & \vdots & \vdots \\
    \vdots & \ddots & \ddots & \vdots \\
    d_{M1} & \cdots & d_{MN}
\end{bmatrix}
\]

\[ D = D_m + jD_\theta \]

[2.1]

Where \([d_{i1} \ d_{i2} \ \cdots \ d_{iM}]\) represents \( M \) frequency bins for a single azimuthal bin. Similarly, a single column of a data matrix represents a response for a fixed frequency bin over the entire range of azimuthal angles. In SAR, the azimuthal data is populated through its multiple sensing locations. In ISAR, the azimuthal data is populated from the target’s rotation.

In the traditional image generation method, the complex data is linearly processed using the Fourier Transform. Additionally, windowing can be applied to remove unwanted image artifacts. The overall flow of the linear image formulation method is illustrated in Figure 2-2.
Using the matrix $D$ as a base, the linear image formulation is given by Eq. 2.2 and Eq. 2.3.

$$D_{cols} = \text{FFT}(D_m \times H_c + jD_\theta, cols \downarrow)$$ \hspace{1cm} \text{[2.2]}$$

$$I = \left| \text{FFT}(D_{cols}^* \times H_r + jD_{\theta}^{cols}, rows \rightarrow) \right|$$ \hspace{1cm} \text{[2.3]}$$

Where $H_c$ is a matrix of size equal to $D$, and each column is a Hanning window; similarly $H_r$ is a matrix where each row is a Hanning window. The final radar image, $I$, is then centered about a central azimuthal angle. Additionally, the radar image can be projected onto the ground plane or any other desired two-dimensional plane. Complex radar systems may also use additional processing to focus and filter the image.

**2.2 Canonical Data**

In order to study some of the properties of SAR and ISAR images, canonical databases were developed. There are three main canonical databases that are used for...
testing – a 2.5-D ray-tracing model, a 2.5-D shadow model, and a point scattering model with occlusion. The 2.5-D models allow for a target height to be specified for each location in a discretized x-y plane; the pseudo-3-dimensional targets are therefore constrained to be solid and designated as 2.5-D. The ray-tracing and shadow models each use rays to traverse the target-radar geometry. The ray-tracing model outputs an estimate of the highlight data according to the reflectivity of a target. The shadow model estimates the shape of the 2-dimensional shadow. In the final model, the target is defined as a set of scattering centers in an x-y plane. Each scattering center has associated geometric and scattering properties. The geometric properties are used to simulate occlusions within the point set, and the scattering properties are used to estimate the reflectivity of the point. Each of the models is described in more detail below.

2.2.1 2.5-D Ray-Tracing Model

The 2.5-D radar model was designed to scan a 1-unit by 1-unit swath in the x-y plane. The swath being scanned can be independently interrogated for any desired resolution in x and y. The radar location, in x and y, is constrained to lie outside of the swath being scanned, but can be located at any positive z. Figure 2-3 shows a target being scanned from a radar located at (-.05, 1.05, 30).
The generation of images for the canonical database uses a simplified model of a side-looking imaging radar. However instead of synthesizing an aperture, a kernel function is used to represent the point scattering blur function. The ray-tracing model images a target in 5 steps:

- A ray connects the radar to every pixel in the image.
- The rays that intersect the object are kept, all other rays are discarded.
- The \((x, y)\) points of the intersections are calculated and projected onto the ground plane.
- 2-dimensional Gaussians, with mean \((x, y)\) and equal variance, are summed for every intersection point.
- The radar image receives additive noise from backscatter according to the desired SNR.

The additive noise from backscatter is calculated in 2 steps.

- Backscatter points are uniformly chosen over the \(M \times N\) image. In actual imaging systems the density of backscatter points varies according to the environmental parameters. To simulate an average density, the model uses 1 point per 40 pixels.
- 2-dimensional Gaussians, centered at each backscatter point, are summed over the entire image.

The normalized image can therefore be defined by Eq. 2.4. The final image can also be corrupted with speckle noise if desired. The initial model accounts for returned
scattering points assuming the object is uniformly and isotropically reflective. The model also accounts for occlusion in the highlight region, but fails to properly generate a radar shadow profile.

\[
I(i, j)_{\text{final}} = \frac{I(i, j)_{\text{auger}} + 10^{-0.05\text{SNR}} \left( \sum_{y'\in y'_{\text{obs}}} \left[ \sum_{x'\in x'_{\text{obs}}} N_{xy} \left( \left( x', y' \right), \left( x, y \right) \right) \right] \right)}{\max \left[ I(i, j)_{\text{auger}} + 10^{-0.05\text{SNR}} \left( \sum_{y'\in y'_{\text{obs}}} \left[ \sum_{x'\in x'_{\text{obs}}} N_{xy} \left( \left( x', y' \right), \left( x, y \right) \right) \right] \right) \right]}^{2.4}
\]

Multiple test images were used in conjunction with the initial ray-tracing model experiments. The first test image is a simple rectangle. The second object represents a more difficult detection problem because of increased geometric complexity. Figure 2-4 shows sample radar images and the two test objects used in the simulations. The figure also illustrates the range of object rotations.

![Test Object 1](image1.png) ![Test Object 2](image2.png)

(a) Targets and sample images  
(b) Target rotation

Figure 2-4: Initial test target database

The figures in (a) illustrate two initial test targets and the corresponding simulated noisy imagery for multiple aspect angles. In the images, the effects of target occlusion are clearly visible. The kernel function can be altered to represent different sized synthetic apertures.
In addition to the simple geometries given above, two tank-like objects were also simulated. The first image has a fixed rectangular base with a gun attached to a circular turret. The second, more complicated model has a domed turret, sloped front face, sloped tread covers, turret non-uniformities, and added rear cargo. Figure 2-5 shows the 2.5-D realizations for both targets and the 2-D height map for the complex target. Figure 2-6 illustrates the simulated images at varying aspects and noise levels.

![Figure 2-5: Simulated tank-like objects](image)

Two tank-like objects were defined and used for testing. The objects are defined in 3-space by a 2-D height map. With this model it is not possible to simulate the circular, raised nature of the tank’s gun.
For consistency, a number of the simulation parameters were fixed. The swath being imaged was a square with unit area centered at (0.5, 0.5). All images had the same resolution and image size, $40 \times 40$. The location of the radar in the XY-plane was further restricted to lay 1.5 units from the center of the imaging swath. The height of the radar was fixed at 8 units. An investigation into the image parameters revealed that at heights lower than 6 units the mean intensity of the target pixels was equal to or less than the mean intensity of the noise pixels. The pixel distributions for the target pixels and noise pixels are seen in Figure 2-7 below.

Figure 2-6: Simulated imagery from tank-like objects
The figure shows sample image generated by the 2.5-D ray-tracing model for the tank-like objects at varying aspect angles and noise levels.
A height of 8 was experimentally chosen because it is large enough to obtain separation between the means, but small enough such that there was still overlap of the density functions. Keeping an overlap between the distributions is important to observe the behavior of the error.

Since the statistics of the image can vary spatially, the image was divided into sub-blocks to allow the statistics of like-pixels to be grouped. The image division was broken into two components – angular and radial. Figure 2-8 below illustrates an example of an image broken into 3 angular and 4 radial regions.

Figure 2-7: Pixel distribution as a function of radar height

The pixel distributions for the rectangular target at low noise are displayed as a function of target and non-target (noise) pixels. The noise pixels populate the low intensity region for all radar heights. For grazing radar looks, there is little separation between the target pixels’ distribution and the noise pixels’ distribution. For steep angles, the separation between the distributions is high. A compromise between the two situations was chosen.
2.2.2 2.5-D Shadow Model

Since the ray-tracing model was unable to capture the shadow profile, a specific model was developed to estimate this feature. Figure 2-9 shows an illustration of the radar, an object, the shadow, and some test rays. The mathematical description of the model is subsequently derived in this section.
To begin, the scan location is defined over a finite grid space in the $x$ and $y$ directions. As in the previous model, the maximum $(x, y)$ point on the grid is $(1, 1)$; the development is equivalent to a separable, normalized realization of an arbitrary grid. Every point on the $M \times N$ grid is defined by its $(x, y)$ location. The $x$-grid is defined as matrix $X$. The grid, $X$, is a composite of a single row vector $\vec{x}$. The row vector is defined from 0 to 1 and consists of $N$ points as defined by Eq. 2.5.

$$X = \begin{bmatrix} \vec{x}_1 \\ \vec{x}_2 \\ \vdots \\ \vec{x}_M \end{bmatrix} = \begin{bmatrix} 0 & \Delta_x & 2\Delta_x & \cdots & 1 \\ 0 & \Delta_x & 2\Delta_x & \cdots & 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \Delta_x & 2\Delta_x & \cdots & 1 \end{bmatrix}, \text{ where } \Delta_x = \frac{1}{N-1} \quad [2.5]$$

Similarly the $y$-grid can be defined by Eq. 2.6.
A single point in either matrix is defined using subscripts \(m\) and \(n\) to denote the row and column number. So the location of an arbitrary point on the ground plane is given by \((X_{mn}, Y_{mn}, 0)\). A ray coming from the radar is defined by a set of \(R\) points connecting the single point on the ground plane to the radar itself. The location of the radar is defined as \((x_R, y_R, z_R)\). The equations in Eq. 2.7, and Eq. 2.8, and Eq. 2.9 assume the radar is located in the first quadrant. Radars in other quadrants are realized by manipulating the appropriate signs of the \(x\) and \(y\) rays.

\[
\begin{bmatrix}
\Delta_y & \Delta_y & \cdots & \Delta_y \\
2\Delta_y & 2\Delta_y & \cdots & 2\Delta_y \\
\vdots & \vdots & \ddots & \vdots \\
1 & 1 & \cdots & 1
\end{bmatrix}
\]

where \(\Delta_y = \frac{1}{M - 1}\) [2.6]

\(Y = \begin{bmatrix} y_1 & y_2 & \cdots & y_N \end{bmatrix}\)

The \(x\) and \(y\) values along the respective rays are converted to the nearest grid location according to Eq. 2.10 and Eq. 2.11.

\[ray^x_{mn} = [X_{mn}, X_{mn} + \Delta^R_x, \cdots, x_R], \text{ where } \Delta^R_x = \frac{x_R - x_{mn}}{R - 1}\] [2.7]

\[ray^y_{mn} = [Y_{mn}, Y_{mn} + \Delta^R_y, \cdots, y_R], \text{ where } \Delta^R_y = \frac{y_R - y_{mn}}{R - 1}\] [2.8]

\[ray^z_{mn} = [0, \Delta^R_z, \cdots, z_R], \text{ where } \Delta^R_z = \frac{z_R}{R - 1}\] [2.9]

where \(\Delta^R_x\) minimizes the distance between \(ray^x_{mn}\) and \(\tilde{ray}^x_{mn}\) and \(\Delta^R_y\) minimizes the distance between \(ray^y_{mn}\) and \(\tilde{ray}^y_{mn}\).
The object which was defined along the same initial grid, \( OB \), is evaluated at all the points in the ray. The new vector \( \vec{h}_{mn} \) corresponds to the object height at each point along the scanning ray, Eq. 2.12.

\[
\vec{h}_{mn} = OB(\vec{p}_\Delta, \vec{q}_\Delta)
\]  

[2.12]

For every point along the ray, the object height can be compared to the ray height, Eq. 2.13. When the \( z \)-coordinate of the ray is greater than the object’s height at that point, an object intersection does not exist. When the \( z \)-coordinate of the ray is less than the object’s height at that point, the scanning ray collides with the object.

\[
\vec{i}_{mn} = \begin{cases} 
1 & \text{if } \text{ray}_{mn}^z \leq h_{mn} \\
0 & \text{if } \text{ray}_{mn}^z > h_{mn}
\end{cases}
\]  

[2.13]

The individual values are calculated point-wise within the vector. The vector is summed. If any point in the intersection vector, \( \vec{i}_{mn} \), is 1 then the given ray will not reach the 0 level \( z \)-plane; the point will be an area of shadow. The darkness image can therefore be defined as Eq. 2.14

\[
d_{mn} = \begin{cases} 
1 & \text{if } \sum \vec{i}_{mn} > 0 \\
0 & \text{if } \sum \vec{i}_{mn} = 0
\end{cases}
\]  

[2.14]

The darkness image, \( D \), can be defined over the entire grid space by evaluating the ray to every grid location, Eq. 2.15.
A shadow is defined by the region where there is darkness due to blockage caused by the target, not the region in which the target itself resides. The occlusions of scattering centers on the target itself are processed according to the initial highlight ray-tracing method, not the shadow model. So finally, the shadow, $S$, is defined by Eq. 2.16.

$$S = D - OB$$  \[2.16\]

Figure 2-10 shows an illustration of a discretized ray intersecting an object in multiple places. This ray results in a shadow at the grid location where the ray intersects the 0 level $z$-plane.

Figure 2-10: An example ray intersecting the target
A sample ray is shown that connects a single pixel to the radar. Since this ray intersects the two tiered target, it is identified as belonging to the shadow region. The radar is defined by its 3-D radar coordinates, and the target is defined by its 2-D height map.
The target in Figure 2-9 is evaluated for multiple aspect angles in Figure 2-11. The shadow is evaluated at 18° increments from 0° to 342°. The radar height is 15 units in z and a distance of 1.7 units in the x-y plane from the center of the object.

2.2.3 Point Scattering Model with Occlusion

The first two canonical datasets used the simplified geometry of a ray-tracing model to estimate the highlight and shadow data within a radar image. Instead of estimating the geometry, the final canonical data model simulates the point spread function obtained from a compilation of $N$ point scatterers. Each point scatterer within the image is defined by its center $(\mu_x, \mu_y)$ and resolution $(\sigma_x, \sigma_y)$. The kernel functions are summed for all point scatterers defined within the image, Eq. 2.17. To
simulate the imagery, a two dimensional sinc function was used as the kernel function, Eq. 2.18

\[ I(i, j) = \sum_{n=1}^{N} \alpha_n K(i, j, u_n^{x,y}, \sigma_n^{x,y}) \]  

\[ K(i, j, u_n^{x,y}, \sigma_n^{x,y}) = \text{sinc}(\sigma_n^{x}(i-u_n^{x})) \times \text{sinc}(\sigma_n^{y}(j-u_n^{y})) \]  

In order to make the point scatterer model more realistic the effects of target occlusion were added. Figure 2-12 illustrates the overall method by which a target point will shadow some nearby points.

---

The figure shows three target point scatterers. The shadow from the left-most point is illustrated. This shadow acts to occlude one of the two remaining points in the image. All associated geometric quantities are illustrated.
In the simulation, the quantities \((H, P,\) and \(d_S)\) are defined by the user while \((x,\) and \(d_P)\) can be calculated from the target geometry. For this implementation the user sets the length of the shadow directly; alternatively, the user could set the depression angle. Using the principle of similar triangles, the height of the shadow at all points can be calculated, Eq. 2.19 and Eq. 2.20.

\[
\frac{H}{d_s} = \frac{x}{d_s - d_p} \tag{2.19}
\]

\[
x = \frac{H \times (d_s - d_p)}{d_s} \tag{2.20}
\]

Next, the shadow height for each point scatterer is compared to the height of all other point scatterers. If \(P_k \geq x_j \ \forall l \neq k\) then point \(k\) will not be occluded, but point \(k\) will be occluded if \(\exists l, l \neq k : P_k < x_j\). Occluded points are not included in the summation of Eq. 2.17. Although a particular scatterer is centered about a single pixel, it has a user-defined width associated with it. The implication is that scattering points can shadow with a width of an arbitrary size.

Figure 2-13 and Figure 2-14 below illustrate scenarios with two point scatterers. In Figure 2-13, both scatterers have equal heights. Therefore given the distance between them, neither point gets shadowed. In Figure 2-14, however, the right point is a fifth the height of left point. When the shadow is left-to-right the taller point shadows the smaller one, but when the shadow is right-to-left the smaller point does not shadow the taller one. In the figures, the red triangles represent the relative location of the radar; the radar is assumed to be infinitely far away and thus all shadows are parallel.
Figures 2-13 and 2-16 illustrate the effects on occlusion when the width and length parameters are varied. When the width is narrow, there is only a small set of aspect angles that will cause one point to shadow another. When the width is wide, there is a greater range of aspects that cause occlusion. The length determines how quickly the height decreases and is directly related to depression angle. For grazing depression angles the effects of shadowing will be stronger; for steep depression angles shadowing has less of an effect.
Finally, a target is given in Figure 2-17 that exhibits multiple point scatterers with varied properties. In the sample images, (b), the edge points are seen to have low heights and are easily occluded. These same points also have the largest amplitude with the poorest resolution. The range and cross range resolutions are assumed to be equal.
The laboratory collected reference ISAR database being examined is of a 1/16\textsuperscript{th} scale variant of the reference model Slicy [27]. Slicy is a commonly used target because it exhibits fundamentally shaped scatterers; Figure 2-18 shows the Slicy target at 45° rotation. The data was collected at 5° and 40° depression, circular azimuth, and at X-Band and Ka-Band frequencies. Table 2-1 outlines the full set of experimentation parameters. The X-Band data is discretized in 64 frequency bins while the Ka-Band data is discretized into 128 frequency bins. The data was provided by Expert Radar Signature Solutions (ERADS) and was approved for public release on October 5, 2004. Their project was sponsored by The United States Army, Intelligence and Security Command and National Ground Intelligence Center (NGIC).
The dataset is 4 data files in total - one for each elevation and frequency band. Each file is in IEEE binary file format. A sample of a header can be found in the Appendix. The data is grouped into polarizations and represented as complex pairs. After the file is opened as IEEE floating point with big-endian byte ordering, the header is read. For each data line the following is performed:

- Read the aspect angle (float32)
- Get the total number of pairs to read (int32)
- Read the IQ data sequentially (float32)
- Format the data into complex pairs
- Formulate structured arrays for each polarization

Figure 2-18: Slicy target
The Slicy target is commonly used for testing and calibration of radar systems. The target exhibits basic scattering geometries.

<table>
<thead>
<tr>
<th>Freq. Band</th>
<th>Depression Angle</th>
<th>Number of Aspect Angles</th>
<th>Range of Aspect Angles</th>
<th>Complex data pairs per aspect (# of freq bins)</th>
<th>Lines in header</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>5°</td>
<td>16731</td>
<td>-5 : 365.0174</td>
<td>256 (64/polarization)</td>
<td>57</td>
</tr>
<tr>
<td>X</td>
<td>40°</td>
<td>4823</td>
<td>-5 : 365.0692</td>
<td>256 (64/polarization)</td>
<td>57</td>
</tr>
<tr>
<td>KA</td>
<td>5°</td>
<td>37002</td>
<td>-5 : 365.01</td>
<td>512 (128/polarization)</td>
<td>56</td>
</tr>
<tr>
<td>KA</td>
<td>40°</td>
<td>37002</td>
<td>-5 : 365.01</td>
<td>512 (128/polarization)</td>
<td>56</td>
</tr>
</tbody>
</table>
It is important to note that the Ka database is quite large so care must be taken when reading the raw data for ISAR processing. Although feasible for the X-band data, exhaustive processing is not possible for the Ka-Band data due to memory constraints. The median RCS values calculated as Eq. 2.21 are plotted as a function of aspect angle in Figure 2-19.

\[ 10 \log_{10} \left( \text{median} \left( |D_{\theta,HH,HV,VH,VV}|^2 \right) \right) \]  

[2.21]

![X-Band RCS plots](image)

(a) 5° depression (b) 40° depression

Figure 2-19: X-Band RCS plots

The plots above show the RCS values for each polarization and depression angle as a function of aspect angle. The RCS value is calculated as the median intensity of the set of complex pairs across the 64 frequency bins.

Single radar images can be generated by selecting sets of data from the desired aspect angles from within the database. The selection of evenly spaced aspect points corresponds to uniform rotational target motion. Figure 2-20 shows an illustration of the complex data for one particular ISAR realization; there are 64 frequency bins in range, and 64 evenly spaced aspects were selected about a central aspect angle.
Once the aspect data is selected, the image formation procedure from Section 2.1 can be applied. After the image is created, there are two corrections that need to be applied. First since the image is split in cross range, the image columns must be barrel rotated by half the image size. The second correction flips the image rows such that the top of the image corresponds to up-range and the bottom of the image corresponds to down-range. Figure 2-21 illustrates the data from Figure 2-20 before and after the correction processes are applied. Figure 2-22 shows the corrected images for all four polarizations.

Figure 2-20: Data for a single ISAR image

The complex data of a particular ISAR realization is illustrated in 64x64 images. The aspect angles were selected from the entire aspect ensemble to simulate uniform rotation in yaw.
The images above used an equal number of samples in aspect and frequency. If the time on target is decreased by 2 then only 32 samples would be available. Similarly, if the time on target was doubled, then 128 aspect samples would be available. Figure 2-23 shows example ISAR images as a function of the number of aspect samples used. By carefully choosing the aspect samples to use, the radar’s pulse repetition frequency, time on target, and target’s rate of rotation can be simulated.

Figure 2-21: Processing of ERADS data

In order to make the formulated image more appealing, two correctional transformations are applied to the ERADS data after image formation. First, the image is shifted in cross-range so that the center of the image corresponds to the central angle. Secondly, the image is flipped so that the top of the image corresponds to up-range, and the bottom to down-range.

Figure 2-22: Multiple polarizations for a single aspect angle

The images show the four polarizations of a particular central aspect angle at X-Band frequencies.

(a) HH                        (b) HV                     (c) VH                      (d) VV

(a) After formation
(b) After correctional rotation
2.4 DARPA’s MSTAR Data

The final database being used is DARPA’s Moving and Stationary Target Acquisition and Recognition (MSTAR) database [28]. The MSTAR dataset is publicly released and was collected from 1995-1997. It contains spotlight SAR images at 1 foot resolution from 5000m range. The images were collected in 1° increments from 0° to 360° in aspect with respect to a static target pose. Because of various sensing restrictions, full sets of circular aperture are not available. Table 2-2 lists the set of MSTAR targets that were available and their respective image characteristics. A copy of the MSTAR header can be found in the Appendix. The header contains sensing and ground truth information from Sandia National Laboratory’s STARLOS radar used to capture the MSTAR data.

Figure 2-23: Varied image sizes for X-Band data

In an ISAR system, the number of angular sets used is determined by the radar’s time-on-target and pulse repetition frequency. The longer a radar obtains returns from a target, the larger the aspect extent which it observes. In SAR systems, the number of angular looks is determined by the number of collections in the flight path. The number of angular sets used affects the cross-range resolution.
Because of computational restrictions, not all of the targets are processed in the subsequent sections. The primary concentration is on the T-72 tank, by far the most common of the MSTAR targets in research literature. The T-72 is a battle tank of Soviet design; Figure 2-24 shows a set of T-72 SAR chips taken at varying aspect angles. Additionally, targets 1, 3, 4, 6, and 8 are used as a representative target subset. Respectively the vehicles are a personnel carrier, reconnaissance vehicle, battle tank, tank, and tank.
cargo truck, and bulldozer. These targets represent a good mix of the types of vehicles that might be found on the battlefield.

Figure 2-24: T-72 SAR chips for varying aspects

The images represent a set of aspect varying SAR chips at 15° depression for the T-72.
Chapter 3

Data Pre-Processing

3.1 Problem Overview

Before any of the advanced fusion algorithms or classification techniques can be applied, there are a few necessary pre-processing steps that must be performed. One of the assumptions of the fusion algorithms is that the images are pre-registered. Registration of multiple radar images results in a pixel-to-pixel correlation between the images. The MSTAR database has pre-registered SAR chips; registration was taken into account when formatting the initial data. Unfortunately, data is often not this well-conditioned and registration is therefore necessary. Section 3.2 outlines the problem of registration and offers two methods of registration – centroid and correlation registration.

The second pre-processing step that is common to many algorithms is segmentation. It is assumed that an image being processed consists of a single stationary target. Therefore, a SAR image can be divided into three main regions – highlight, shadow, and background clutter. The highlight region takes into account all of the radar returns that are received due to the scattering properties of the target. The highlight region is the most easily detected of the three regions and can be obtained with a fixed threshold if necessary. The background clutter and shadow region however are not as easily extracted. Section 3.3 outlines three methods of segmentation. The first is a series of morphological operations designed to extract the shadow region. The other two
methods use Gaussian mixture models to represent the intensity and spatial features of individual pixels. The segmentation algorithms are tested using the MSTAR database.

### 3.2 Registration

In radar images the x and y axes correspond to the range and cross-range resolution cells. These cells are a function of the aspect angle. To get a pixel-to-pixel correlation between images they need to be rotated. It is assumed that the radar aspect angle is known in all of the registration algorithms. Using the known aspect all of the radar images can be properly rotated into the spatial coordinates of the target. After rotation, the range and cross-range axes do not align with the primary image axes, but the image pixels are correlated in the target’s spatial coordinate system. This new alignment is critical for fusion algorithms. This alignment procedure holds for image chips of isolated targets.

In practice, however, a radar chip may not necessarily only be of a single target. Larger swath images such as the one displayed in Figure 3-1 display multiple targets, buildings, and geographical features. Other, more robust, registration techniques can be used in these instances since there is an increased number of features that can be used for registration [29] [30]. Although images with single chips have a lower feature space there are still some techniques that can be applied to correct for translation shifts in images. The following section identifies two methods that correct for translation shifts of SAR/ISAR chips with known aspect.
3.2.1 Centroid Registration

The first registration method being investigated is alignment using the center of mass within the radar images, the centroid. First, an input image is thresholded into a binary image. Next, the centroid is calculated. The image is then placed into a zeroed image such that the center of mass of the object corresponds to the center pixel of the image. Subsequently, the image is rotated according to the known central aspect angle. Finally, the image is cropped back down to its original size. The overview of this

Figure 3-1: Typical SAR image [31]

A SAR image of the Redstone target array is displayed. In the algorithms considered here, it is assumed that each individual target would be detected and a single SAR chip formed using only that particular swath of the image. For registration purposes, this image clearly has more distinctive features than a single target SAR chip.
registration process is given in Figure 3-2. This technique assumes that the center of mass of the target corresponds to its rotational center.

Figure 3-2: Alignment using target centroid

The alignment process illustrated above is useful for single target chips with a known aspect angle. The aspect angle is defined as a function of an absolute geographic coordinate system, not target pose. The technique centers the images based on the center of mass of the visible scattering centers.

Registration of the X-Band ERADS database was investigated to test the correlation method. The test data was generated at 1° increments in aspect from 0° to 359°. The images were of size 64 x 64. Accordingly, the center of the image is (32.5, 32.5). The centroid misadjustment is defined as the displacement of the object’s center of mass relative to the image center. Figure 3-3 shows a plot of the x and y displacement for the set of images. The dots represent the true displacement, and the lines show the rounded result used in the alignment procedure; the alignment can only account for whole-pixel displacements.
Since two different thresholds lead to two different sets of registered images, both cannot be correct. Determining the optimality of a particular threshold is nontrivial, so an acceptable threshold value was obtained experimentally. The threshold was varied as a function of the polarization. A value of -20 dB was seen to produce the most desirable results for the HH polarized X-Band images of size 64x64.

Figure 3-3: Misadjustment of the centroid for two threshold levels

The figure shows the misadjustment of the centroid for the x and y axes. There are a number of interesting factors within the centroid analysis. First, the centroid varies greatly for only small changes in aspect. Next, the threshold is also seen to effect the centroid location. Finally, the centroid is misregistered in cross-range even though image formation centers about this axis.

Since two different thresholds lead to two different sets of registered images, both cannot be correct. Determining the optimality of a particular threshold is nontrivial, so an acceptable threshold value was obtained experimentally. The threshold was varied as a function of the polarization. A value of -20 dB was seen to produce the most desirable results for the HH polarized X-Band images of size 64x64.

Figure 3-4 (a) shows the location of the centroids after they are aligned into the larger image. Notice that all points are bounded by ±.5 pixels from the desired value. The standard deviations are also given to illustrate the prevalence of sub-pixel errors. The centroid analysis was performed again after rotation and cropping, Figure 3-4 (b). Since rotation involves interpolation, the centroids are no longer bounded by ±.5 pixels from the desired value.

The errors depicted illustrate sub-pixel and interpolation errors and are a fundamental limitation in the implementation described in Figure 3-2. With additional
processing, the amount of sub-pixel registration error can be reduced; a nearest-neighbor
approximation was used to up-sample by a factor of 4 before alignment. Up-sampling
clearly reduced the effects of the sub-pixel centroid errors, (c). Figure 3-5 shows the
mean image after registration and the mean image after registration with up-sampling.

Figure 3-4: Centroid position during registration

Due to the pixel-level implementations, there are variations in the centroid location for the registered
datasets. The standard deviations for the final registration are $\sigma_x = .3266$ and $\sigma_y = .2931$. The variance can
be reduced by introducing up-sampling into the registration process. The results deviations are reduced to
$\sigma_x = .1719$ and $\sigma_y = .1432$.

Figure 3-5: Resulting mean images of X-Band data

The images illustrate the mean of a dataset of registered SLICY images. Up-sampling in the registration
process helps to eliminate some of the sub-pixel errors, and therefore results in a slightly improved set of
registered image.
From Figure 3-5, it is difficult to tell the magnitude of the improvements. Visually, the scattering centers are seen to be slightly more localized in dataset processed with up-sampling. With images of poor resolution it is difficult to determine the exact significance. The variances of the centroid errors, however, are in fact reduced considerably.

In conclusion, the alignment results seem to be acceptable although some noticeable errors in individual images persisted. The three main sources of error that were encountered are centroid errors, threshold errors, and sub-pixel errors. The description of each and the steps used to counteract these errors are listed below.

Centroid errors occur due to the fact that the centroid of the image does not correspond to target’s center of rotation. This is illustrated in the location of the cross-range centroid. After processing, the images should, by definition, already be registered in cross-range, but the centroid indicates ± pixel errors. This error is intrinsic to the method and cannot be counteracted without complex model geometries to account for the percentage of the target occluded in a given image.

Next, the centroid was also illustrated to be dependent on the threshold used. Using a single fixed threshold for all images can lead to jittering. The centroid calculated for consecutive images in the database can vary drastically; the centroid should vary smoothly. Smoothing functions were implemented to combat some amount of jittering. Selection of a sub-optimal threshold can also lead to misalignment. An experimentally determined optimal threshold also seemed to help reduce errors. Finally, sub-pixel errors result from pixel limitations in the process implementation. If the image is interpolated
into a larger space, the sub-pixel errors can be drastically reduced at the cost of computational complexity.

Centroid registration is essentially a manifestation of a 1-point template registration technique. A single seed template image is used for the alignment all of the individual images. Centroid alignment can be successfully used for gross alignment, but because of the persistence of errors the final registration is still flawed. Therefore a secondary registration technique should be applied to more optimally register the images.

3.2.2 Jitter Analysis

Since the centroid registration technique had some fundamental limitations a normalized cross-correlation technique was developed and tested. The cross-correlation technique calculates the x and y translations where the maximum correlation occurs. The maximum correlation corresponds to the best translational match between images. This technique relies on the fact that images will only change slightly from one aspect to another. Large variations of the image features or large gaps in aspect can cause misadjustment since the images themselves will not be correlated.

In order to perform the correlation registration process a known template with good registration is needed. Since there is no single image that all images can be mapped to, the template must change as a function of the aspect angle within the database. One such method for handling this approach is given in Figure 3-6. In this approach there is a single fixed initial template. Each image is then registered to the image that immediately precedes it within the aspect varying database. This process is iteratively performed for
all images within the database. Theoretically, all of the images will properly adjust so that the features are perfectly correlated within the pixels.

Figure 3-6: Method of registration for the entire dataset from a single template
The figure shows a method by which an entire dataset of aspect varying looks can be co-registered (dejittered). Each image is mapped to match the previous aspect image. The initial template can be any image within the template. The closet angle to 0° in pose is used in the experiments.

A jitter analysis can be performed to co-register sets of subsequent images within a data set. For example, assume a set of SAR images are rotationally aligned, but the translation component contains an error of ± a few pixels. If the set of SAR images are viewed sequentially as a movie, the target will jump around and appear to jitter on the screen. The mis-registration in the images will cause fusion methods to fail. The correlation technique should correct this jitter.

To test the correlation method of translational registration, multiple sets of canonical images were created. Figure 3-7 illustrates the mean, median, and maximum images for the target 1 canonical dataset under low-noise. Misalignment was added to the data in the form of uniformly distrusted translations of ±3 pixels in both directions. The target features within the mean image are seen to blur when the dataset exhibits a jitter. After the proper alignment the target features are clearly visible. Figure 3-8 shows...
the correlation of sequential images and the corresponding offset error. In the 360 images there was only a single pixel error. The drawback, however, is that this error is propagated to all subsequent images in the dataset.

Figure 3-7: Registration results for target 1 under low noise

A simplified tank-like target was imaged using the ray-tracing method at 1° increments. Each image was subjected to translational noise of ±3 pixels. The result of this jitter is rounding of features in the mean image and blurring in the maximum image. After alignment, however, the images are properly aligned.
The same target was again tested in the presence of high noise. The jitter images and corresponding registered images are given in Figure 3-9. Again, rounding is seen to occur in the mean image. The noise in the background region is seen to be canceled. The maximum image shows that the target features are lost under a curtain of noise. Unfortunately, the registration technique fails. The de-jittered set of images is not in fact registered and contains more translational errors than the initial dataset. Figure 3-10 shows that the correlation between subsequent images in the noisy dataset is significantly lower than the low-noise dataset. Moreover, the error introduced is not stable. Since the translational errors propagate, the image is seen to start to slide across the surface of the image. A given subset of two images may appear to be registered, but due to the propagation of errors, the target is seen to float across the entire image space as the angular index is varied.

Figure 3-8: Registration performance for target 1 under low noise

The correlation between subsequent images in the dataset is seen to be high. For a total of 360 images, the correlation matching produced an error of only a single pixel. This error, however, was propagated to all subsequent images in the database.
Figure 3-9: Registration results for target 1 under high noise
A simplified tank-like target was imaged using the ray-tracing method at 1° increments. Each image was subjected to translational noise of ±3 pixels and high additive noise. The result of this jitter is rounding of features in the mean image and severe blurring in the maximum image. The high additive noise is not correlated between images and pixel errors are propagated; the registration method fails.

Figure 3-10: Registration performance for target 1 under low noise
The correlation between subsequent images in the dataset is seen low due to the noise. The errors propagate from image-to-image and cause the registration method to fail.
The same test was performed for target 2, a more detailed tank-like object. Since the target itself had more features, the registration method was tolerant of higher levels of noise. A comparison of the absolute errors of target 1 (low noise), target 1(high noise), and target 2 (low noise), reveals the overall performance of the algorithm - Figure 3-11.

In both low noise cases, there are almost no errors. The high noise case, however, shows that there is an error in every image. Not only is there an error in the correlation of sets of subsequent images, but this error is propagated from previous misadjustments.

The final canonical data analysis focused on determining the maximum aspect between looks that could be used to obtain a satisfactory correlation. The analysis was performed using the data from target 2. Figure 3-12 shows the error analysis. The correlation is seen to decrease steadily as the angle between subsequent images is increased; accordingly, the error vector increases. After 10° degrees there will be a translation error of 1 pixel in 1 out of every 10 images. If this level of error is propagated through a dataset of 360 images, the final image could be offset by over half the input.
The constraint of the algorithms is therefore that the correlation be performed with subsequent images that are as close in aspect as possible.

An analysis of the MSTAR SLICY data reveals that registration errors are occurring for this particular target set. This jitter destroys the integrity of the pixel-wise statistics and degrades any subsequent fusion. An example of the jitter in subsequent images in the dataset is displayed in Figure 3-13. The initial template, (a), is an image with a certain aspect dataset. The subsequent image in aspect, (b), has a small translational error compared to the initial template. From the error image (d) it is clear that the image jitters ± a few pixels. After translational adjustment (c) and (e), the image is better registered to the initial template.

Figure 3-12: Error as a function of configuration
The correlation technique relies on the fact that subsequent images in the dataset can be correlated to one another. As the aspect angle between these images is increased, the level of correlation between the images drops. For images that are not highly correlated, the prevalence of errors degrades the overall effectiveness of the technique.
Figure 3-14 shows the cross-correlation images for the aspect images illustrated above. The correlation image indicates that the test image should be shifted to better match the template. After the appropriate translation, the correlation is again calculated. From (b), the two images are more highly correlated after the registration and the jitter is reduced. The optimal correlation is not 1.0 because the image features change for even slight changes in aspect.
Unfortunately, this technique is seen to fall short of the desired de-jittering effects in the long term. The local jitter is gone. From image to image the scattering centers do not jump around the image. The features however slowly float around within the image. This is a common problem in many registration problems – the propagation of errors. For ISAR/SAR images this problem is even more serious since there are few, if any, features that persist in multiple images for large changes in aspect.

One method for correcting this propagation of errors is to use multiple initial templates, Figure 3-15. Also, the images are compared to the templates in multiple directions within the dataset. Initially, images were compared as a function of increasing aspect; the bidirectional method compares images as a function of increasing and decreasing aspect to minimize the distance from the nearest template. In the illustrated implementation, the result is a 4-fold increase in the number of initial templates, but an 8-fold decrease in the number of images that are sequentially registered to one another.

Figure 3-14: Normalized cross-correlation for translation corrections
The initial cross-correlation plot shows that the images are misadjusted by 3 pixels in x and 1 pixel in y. After the translational correction, the peak correlation is seen to be centered.
3.3 Segmentation

As mentioned in the introduction, the segmentation of a SAR chip into its respective regions is an important pre-processing step for many of the advanced algorithms. One of the factors that makes the segmentation of SAR images difficult is the variation in intensity levels of the shadow and background from one image to another. The relative pixel intensities vary as a function of the aspect angle of the radar, thus it is
impossible to segment via tradition methods such as a single fixed threshold. The SAR images in Figure 3-16 illustrate these variable statistics.

![SAR images of T-72 illustrating variable region statistics](image)

Figure 3-16: SAR images of T-72 illustrating variable region statistics

The nature of SAR images is that they change drastically with respect to angle. The background and shadow regions illustrated above vary significantly for the same target and same physical background.

In optical applications where images exhibit variable intensity, normalization schemes have been used prior to applying segmentation schemes. Normalization strategies are not appropriate solutions since the region properties themselves are changing and radar images contain speckle noise. A common alternative technique that has shown some promise in SAR imagery is the calculation of optimal thresholds on an image-to-image basis [32] [33].

Clustering has also been shown to be an effective method of segmenting SAR images. [33] combines thresholding and clustering techniques; first, the threshold for a SAR image is calculated based on a 5% / 95% median value to obtain a target estimate. Clustering is then used to estimate the target highlight cluster. Finally, some morphological operations are used to determine and eliminate spurious pixels. Section 3.3.1 outlines a technique for extracted the shadows based on similar morphological operations. [34] proposes an automated approach that clusters the data based on texture
features, intensity, and edge features at multiple scales. [35] were able to show that a Bayesian approach could also be used to obtain satisfactory results on the MSTAR database. In [35] however, training datasets were needed to create a suitable knowledge-base from which the appropriate probabilistic quantities could be obtained.

The expectation-maximization algorithm is an ideal blend of clustering and Bayesian probability [36] [37]. The EM algorithm can be run over all pixels for each image. The associated class labels can be used for image segmentation while the class conditional probabilities and likelihoods can be used for data fusion or classification.

Sections 3.3.2 and 3.3.3 focus on the use of the class labels and class conditional probabilities for the purpose of image segmentation. The EM algorithm and subsequent morphologic processing used for segmentation is described in Section 3.3.2. This automated segmentation approach will be compared to a semi-automated thresholding technique. The semi-automated technique involves a user picking an optimal threshold for the highlight and shadow regions in every image. The thresholded image is then also subjected to morphological post-processing to further refine the segmentation. The comparison of techniques is based on an optimal manual segmentation that involved hand tracing the highlight and shadow regions. The goal of the comparison is to show that by only using intensity and location inputs, the fully automated algorithm performs the same, if not better, than the semi-automated approach. Finally, Section 3.3.3 improves upon the automated EM model by including smoothing functions.
3.3.1 Morphology

Since the highlight values do not change drastically from one image to another a fixed threshold can be sufficiently used to obtain an estimate of this region. The shadow regions however are not as easily obtained. The following section outlines an 8 step approach to obtaining an estimate of the shadow region. Gaussian smoothing is used to help combat the speckle noise. Histogram equalization is used to account for the variable region statistics from image to image. The algorithm is given by the following 8 steps:

- **Gaussian Smoothing**: The original image is smoothed in order reduce noise artifacts. An experimentally determined optimal variance was found to be $\sigma=3$ for the targets under consideration within the MSTAR database. Values between 1.5 and 5 can be used depending on the desired ‘tightness’ of the final fit. The smoothing operation is given by Eq. 3.1

$$I_{\text{smooth}}(i, j) = I_{\text{in}}(i, j) * G(0, \sigma) \quad [3.1]$$

- **Image inverse**: The image is first normalized between [0 1]. Next, an inverse map is applied to the image, Eq. 3.2, such that the shadow region is high valued.

$$I_{\text{inv}}(i, j) = 1 - I_{\text{smooth}}(i, j) \quad [3.2]$$

- **Histogram equalization**: Histogram equalization transforms the input pixels $r_{\text{inv}}$ with $T(.)$ such that the output histogram has pixels values spread evenly on the interval [0 1]. The mapping function is given by Eq. 3.3

$$s_{\text{eq}} = T(r_{\text{inv}}) = \sum_{j=0}^{k} \frac{n_j}{n} \quad k = 0,1,2,\ldots, N-1 \quad [3.3]$$
- **Thresholding**: The next step converts the intensity image into a binary image according to Eq. 3.4.

\[
I_{\text{thresh}}(i, j) = \begin{cases} 
0 & I_{eq}(i, j) \leq \text{threshold} \\
1 & I_{eq}(i, j) > \text{threshold} 
\end{cases} \tag{3.4}
\]

- **Edge Removal**: Removes the edge area caused by the initial rotation of the SAR image.

- **Removal of small pixel groups**: Object groups containing less than the desired number of pixels are discarded. From this dataset, the optimal size was determined to be 500 pixels.

- **Fill holes**: Pixel values of 0 inside the area of all object groupings are set to 1.

- **Selection**: The shadow region is assumed to be a region(s) centrally localized within the image. Any object grouping that has 8-connectivity with the selection box is retained; the remaining objects are discarded.

The overall procedure is illustrated on a test image in Figure 3-17. After a visual analysis of the results, the algorithm has performance comparable to some of the other published techniques. The problem, however, is that the technique is not robust enough to handle a variety of cases for a variety of targets. Additionally, the parameters need to ‘dialed in’ for the desired results; the method is very database dependent. The technique would be improved by a greater level of automation.
3.3.2 Mixture Modeling

In order to improve the robustness and automation of the segmentation procedure a mixture modeling approach was applied. Expectation-maximization (EM) can be used to estimate the model parameters and thus segment the image. The overall automated EM segmentation algorithm can be broken into 4 basic steps – EM, selection,
thresholding, and morphology. The first step in the algorithm is to perform the expectation maximization. The EM steps are designed under the assumption of a mixture of $M$ multivariate Gaussians each with $D$ dimensions. The initialization, parameter selection, and other details associated with this part of the algorithm are given in Section 3.3.2.1. Once the class labels are obtained, the second stage of the algorithm takes place, selection. The selection process requires identifying the appropriate labels for the highlight and shadow regions. Once the labels are obtained, the corresponding class conditional probabilities are taken from the output of the EM algorithm. These probabilities are thresholded to obtain segmentation estimates. The final step is a morphological operation that cleans the given estimates. Once the final segmentation is obtained, it can be displayed independently or overlaid on the original SAR image. The overall process is illustrated in Figure 3-18.

Figure 3-18: EM segmentation algorithm

The EM segmentation algorithm is outlined above. The technique used the class conditional probabilities output from a multi-variant Gaussian mixture model to estimate the probabilities of a pixel belonging to the shadow and highlight regions. The probabilities are thresholded and cleaned to obtain the final estimate.
3.3.2.1 Expectation Maximization

The first step in segmenting a given SAR image is to calculate the maximum likelihood parameters of a mixture of Gaussians in a $D$-dimensional feature space. This is a well developed application of EM for general Gaussian mixture models. In the E-Step, the algorithm is used to calculate the expected class of all pixels for each class. The M-Step computes the updated maximum likelihood parameter estimates given the class memberships [38]. After the algorithm is initialized, this process is iterated until the necessary convergence criterion is met. Upon completion, the class conditional probabilities, a posterior estimates, provide the necessary information to determine which of the $M$ Gaussian distributions each pixel belongs to. The goal is to have the shadow, highlight, and clutter regions all belonging to different classes. Each region need not be associated with only one class, but one class should not be associated with more then one region.

The probability density of a Gaussian mixture model of $M$ elements is given by Eq. 3.5.

$$p(x \mid \Theta) = \sum_{i=1}^{M} \pi_i p(x \mid \theta_i) \quad [3.5]$$

Where $\Theta$ is the collection of parameters $\{\pi_1, \ldots, \pi_M, \theta_1, \ldots, \theta_M\}$. The parameter $\pi_k$ is the prior probabilities that $x$ is generated by the $k^{th}$ distribution of the mixture, $\omega_k$; these are also known as the mixing weights and are subject to the condition in Eq. 3.6.

$$\sum_{i=1}^{M} \pi_i = 1 \quad [3.6]$$
From Eq. 3.5, $p(x | \theta_i)$ is a Gaussian density parameterized by $\theta_i$ in terms of the mean and covariance matrix, $\mu_i$ and $\Sigma_i$, respectively, and is associated with class $\omega_i$. A $D$-dimensional Gaussian density with full covariance matrix is given by Eq. 3.7.

$$p_{\omega_i}(x | \theta_i) = \frac{1}{(2\pi)^{d/2} |\Sigma_i|^{1/2}} e^{-\frac{1}{2}(x-\mu_i)^T \Sigma_i^{-1} (x-\mu_i)}$$  \[3.7\]

The feature selection determines the dimensionality $D$. For the purpose of segmenting the images used here, 1 and 3 dimensional input vectors are tested. First, the performance is tested when the mixture is a function of the 1-D intensity space. To improve on the performance and to take into account the spatial variability, a 3 dimensional vector is also used. The dimensions are intensity, and the pixel’s $(x, y)$ coordinate. All parameters are normalized to lie between 0 and 1.

In Eq. 3.7 a full covariance matrix is used to capture the correlation between multiple spatial features and the correlation between spatial and intensity features. For this implementation, segmentation performance is more critical than run-time. If run-time is critical, then a diagonal covariance matrix could be used.

The algorithm is initialized based solely on the intensity values for both the 1D and 3D cases. The intensity values between 0 and 1 are divided into $M$ evenly spaced bins. After the image is thresholded, the pixels falling within the $i^{th}$ bin are assigned to class $\omega_i$. The appropriate model parameters are then estimated based on these class memberships. Once the parameter estimates are obtained, the algorithm can then be iterated. The E-step, Eq. 3.8 estimates the class conditional probabilities for each class over all pixels.
The M-step, updates the maximum likelihood parameter estimates given the new class memberships. The new priors, means, and covariance matrices for each class are given by Eq. 3.9, Eq. 3.10, and Eq. 3.11 respectively.

\[
p(\omega_j | x_j, \Theta) = \frac{\pi_i p_{\omega_i} (x_j | \theta_j)}{\sum_{k=1}^{M} \pi_k p_{\omega_k} (x_j | \theta_k)} \]  

[3.8]

Traditional algorithms terminate when the likelihood from one iteration to the next increases by less than some percent \( p \). However, since here, the class conditional probabilities are used in subsequent processing, the algorithm is terminated when this quantity changes by less than a designated percentage. Alternatively, the algorithm is stopped when the number of iterations passes a certain fixed threshold.

3.3.2.2 Post-EM-processing

One of the outputs of the EM algorithm is the class labels, an estimate of which of the \( M \) Gaussians in the mixture a given pixel most probably belongs to. This class information alone is not sufficient to segment the image when there are more than 3
classes. If exactly three classes are used (1 for shadow, 1 for highlight, and 1 for clutter) then the class labels can be thought of as a segmentation estimate directly. For this implementation it is assumed that a single class can be obtained for the target highlight and shadow regions. The clutter is not restricted to a single class mixture.

In order to select the appropriate classes it is assumed that the target will be centrally located within the image; it is not necessary that the target be exactly centered or in a particular orientation. The algorithm for finding the class belonging to the target highlight is:

For each class:
- Discard all pixels not within a 30% radius from the center of the image
- Count the number of labels that correspond to a pixel with a value above a threshold of .8
- Pick the class with the highest count

The algorithm for finding the class belonging to the target shadow region is:

For each class:
- Discard all pixels not within a 20% radius from the center of the image
- Count the number of labels that correspond to a pixel with a value below a threshold of .1
- Pick the class with the highest count

This method is essentially equivalent to obtaining a fixed threshold segmentation and picking the set of class labels that best match the estimate.

Once the class labels are known, the highlight and shadow regions can be described by the appropriate class conditional probabilities. A high class conditional probability indicates that the probability the given pixel belongs to the highlight or shadow class is very good. Class labels are based on the fact that a given pixel has a greater probability of belonging to class $j$ than any other class. This does not require the probability of belonging to $j$ to be sufficiently high, just higher than any other class. Therefore, a threshold is used to obtain only those pixels with high class conditional probabilities for the highlight and shadow regions.
Next, morphological operations are applied. Their purpose is to perform some simple connectivity filtering to smooth the image and remove spurious pixels. The highlight filtering is given by 5 steps: a closing operation, an h-break, removal of small objects, filling holes, and smoothing with a majority filter. The shadow filtering is given by 6 steps: an h-break, removal of very small objects, smoothing with a majority filter, removal of small objects, selection of objects with close spatial relation to highlight region, and filling of holes.

Again, these refinement steps are designed to select the main highlight and shadow regions, remove spurious pixels, and smooth the edges. The filters do not substantially change the overall estimate.

### 3.3.2.3 Overview of Results

The following section illustrates an overview of the segmentation results. Figure 3-19 shows an example SAR image of the T-72 tank and the corresponding class labels. The results are from a 3D mixture model for a mixture of 12 Gaussians; both spatial and intensity features are used.
After the EM algorithm is iterated until the necessary convergence criterion is met, the class conditional probabilities are processed. Figure 3-20 (a) and Figure 3-21 (a) confirm that refinement procedure correctly identified the 2 classes (out of 12) that belong to the shadow and highlight regions respectively. The highlight region is more distinctive from the shadow regions; this distinction can clearly be seen in the high class conditional probabilities. The shadow class, on the other hand, is categorized by both high and low spatially distributed probabilities. The class conditional probabilities form a 3 dimensional ellipsoid centered on the shadow. Once the class conditional probabilities are thresholded, only the valid shadow region and some spurious pixels are left, Figure 3-20 (b) and Figure 3-21 (b). Some simple morphological operations clean the spurious pixels and the final segments are obtained.

Figure 3-19: Sample result of class labeling
In this example an input T-72 SAR image at 15° depression is labeled according to a 12 class mixture model. The highlight and shadow regions belong to 1 class each. The remaining classes categorized the clutter.
Once the respective regions have been identified, they can be labeled accordingly. These labels can then be superimposed back into the initial SAR image as in Figure 3-22.
The area overlap factor, Eq. 3.12, is used for evaluating the performance of the segmentation procedure versus a manually segmented image. \( T(i, j) \) represents the true segmentation and \( \tilde{T}(i, j) \) represents the estimate [39].

\[
\Omega = \frac{\sum_{i=1}^{N} \sum_{j=1}^{M} T(i, j) \& \tilde{T}(i, j)}{\sum_{i=1}^{N} \sum_{j=1}^{M} T(i, j) + \sum_{i=1}^{N} \sum_{j=1}^{M} \tilde{T}(i, j) - \sum_{i=1}^{N} \sum_{j=1}^{M} T(i, j) \& \tilde{T}(i, j)} \tag{3.12}
\]

This overlap factor can be applied to both the semi-automated thresholding method of segmentation as well as the automated EM segmentation. The manually segmented images will be used as the ground truth to which both methods will be compared.

Clearly, it is desirable to have the overlap factor as high as possible. The floor for acceptable matches varies from application to application. In this case, subsequent processing of the images will further spur and smooth the edges so therefore the cutoff is set to accept the shapes that have the same overall appearance, location, and features, not perfect area correspondences. For the application of segmentation used here, it is desired
that the overlap factor is greater than .7, but over .6 is still acceptable. Also, since the manually segmented images themselves are partially subjective there are some fundamental limitations to the ratio itself.

Figure 3-23 shows 2 additional example segmentations. The EM segmentation on (d) has performance \( \{ \Omega_S, \Omega_H \}, \{ .83, .83 \} \) and the threshold method yielded an image with performance \( \{ .78, .88 \} \). The plot of the error for the EM segmentation is given in Figure 3-24.

---

Figure 3-23: Sample segmentations

The following shows the sample segmentations for two SAR chips. The ideal method is obtained by hand and included an influence for the background clutter. In both examples the EM output seems to be more desirable than the results obtained using the variable threshold.
Both the shadow and highlight segments are satisfactorily obtained. The average boundary error appears to be approximately 1 pixel. The first segmentation in Figure 3-24 (a) illustrates an example of a result that is acceptable, but whose performance is less than desired. The EM segmentation had performance \{.64, .76\}, and the threshold method was at \{.74, .77\}. From the error image, it is clearly identifiable which areas on the shadow the segmentation missed.

The fact that the gun turret was not included in the segmentation is not surprising and does not indicate failure. In addition to the gun turret, the segmentation method did not clearly identify the bottom boundary of the shadow. Because of these areas, the overlap factor was only .64. This value is still considered acceptable because the algorithm was able to capture the overall shape of the shadow with an average boundary error less than approximately 5 pixels.
Using the overlap area error metric, the performance of a reduced set of T-72 images from the MSTAR database can be compared. Figure 3-25 shows the performance on 25 test images for the thresholding and EM methods. A reference line for the minimum acceptable value (red) and the desired value (magenta) is also shown. In both cases the EM algorithm does not produce any segmentation under the acceptable limits. For the highlight region all but 2 values are above the desired .7 ratio. Not surprisingly, the performance on the shadow area is not as high as the highlight. Still, however, the overall performance is quite good. Table 3-1 gives the mean values and variances for the four different cases. Clearly, the automated EM algorithm performs as well if not slightly better the semi-automated thresholding method.

Figure 3-25: Performance metric

The EM method is seen to be better then the semi-automated thresholding method for a majority of the test images. Additionally, the segmentation is always above the acceptable minimum performance level for all cases using the EM algorithm.
The convergence of the algorithm required between 100 and 400 iterations depending on the image. In comparison to algorithms for optical images this approach seems costly; however the time required to segment a SAR image is quite good compared to other methods commonly used. The simulation code was written in MatLab and was written for ease of simulation and not optimized for speed. Running on a P4 2.8 GHz machine with 1 GB RAM, a 139x138 image that required 381 iterations took 215 seconds. In comparison, [34] reported 420 seconds for segmentation using fuzzy C means clustering of a 90x90 image. The EM technique is at least one full order of magnitude faster.

The remainder of the section is dedicated to interim test results and the derivation of optimal operating parameters. Two of the most crucial parts of designing the EM algorithm were selecting the input features, and choosing the model order. As a first approximation, the input features were selected to be the intensity values. The histogram for a manually segmented target is given in Figure 3-26.

<table>
<thead>
<tr>
<th>Region</th>
<th>EM Mean</th>
<th>EM Variance</th>
<th>Threshold Mean</th>
<th>Threshold Variance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shadow</td>
<td>0.7494</td>
<td>0.0063</td>
<td>0.7292</td>
<td>0.0065</td>
</tr>
<tr>
<td>Highlight</td>
<td>0.7989</td>
<td>0.0052</td>
<td>0.7948</td>
<td>0.0101</td>
</tr>
</tbody>
</table>
The model order was selected to be 3; one class each for the shadow, highlight, and clutter. Figure 3-27 shows that only 3 classes are not sufficient for accurately segmenting the given image. When the image is broken into 3 components the shadows are identified as part of ground clutter and ground shadows. The target shadow is not isolated. Also the peak in the histogram near 1 is not well estimated.

Figure 3-26: Histogram for a manually segmented target

The figure shows a histogram of the intensity values as obtained via a manual segmentation. The region was divided into three regions. It is clear from the intensity histogram that the highlight information is more easily segmented from the clutter than the shadow.

The model order was selected to be 3; one class each for the shadow, highlight, and clutter. Figure 3-27 shows that only 3 classes are not sufficient for accurately segmenting the given image. When the image is broken into 3 components the shadows are identified as part of ground clutter and ground shadows. The target shadow is not isolated. Also the peak in the histogram near 1 is not well estimated.

Figure 3-27: Results for target segmentation

Using 3 classes and only intensity based features the shadow cannot be well segmented.
A closer examination to the histogram of the manually segmented target reveals that the clutter distribution is itself composed of more than a single Gaussian. When the model order is increased to 4, the estimation obtained improves, Figure 3-28.

![Class estimates](image1.png) ![Image histogram and estimates](image2.png)

Figure 3-28: Results for target segmentation

When the number of classes is increased to 4, the target shadow and the ground clutter become more distinctive, but the target shadows and ground shadows do not. The highlight information is well estimated.

The estimate of the data is improved when the number of classes increases from 3 to 4. The highlight peak close to 1.0 is more accurately modeled; the target highlight is distinguished from high valued clutter. However, there is still no distinction between the shadows due to clutter and the shadows due to the target.

Figure 3-29 shows that the overall performance of the 1D segmentation method is quite poor with respect to the segmentation of the shadow region. Approximately a third of the segmentations are below the minimum overlap. Increasing the model order will not serve to increase performance as the intensity values alone do not hold the necessary discriminatory information.
Since, the intensity values alone cannot distinguish between clutter shadows and target shadows, a spatial component was introduced into the feature vector. The (x, y) coordinate of the point was added to the input feature vector to make the Gaussians 3 dimensional. A full covariance matrix was used. Figure 3-30 illustrates the class structure and final segmentation for model orders of 5, 12, and 20.

Figure 3-29: Performance of 1D estimations

The EM algorithm using only intensity based features fails to accurately segment the shadow in approximately a third of all cases. The performance is unacceptable for the shadow. The highlight performs well in many of the test cases, but still fails for a small percentage.
When the model order is only 5, the clusters only account for the variations in the intensity and in the x-direction. As a result the centralized clutter is all labeled as shadow. This is clearly undesirable and represents a case of under fitting. For the case of 12 clusters, the model closely matches the expected state. Finally, for the case of 20 clusters over fitting occurs in the shadow. The target shadow is broken into multiple components and can therefore not be accurately recovered.
An analysis was then performed over a range of model orders to see if the above trends were applicable over the entire dataset. Also, the simulation serves to determine the dependence of the system on the parameter used to threshold the a posterior probability. Figure 3-31 below illustrates the number of images (out of 25) that exhibited an error metric value above the desired .7. Figure 3-32 illustrates the mean overlap factor for both the shadow and highlight regions. The model order was varied from 2 to 20 and the threshold parameter was set to \{.7, .8, .9\}.

![Highlight analysis](image1)
![Shadow analysis](image2)

Figure 3-31: Number of test instances that had an overlap factor over the desired level
In the case of the highlight, the performance levels off around 10 classes. For the case of the shadow, the performance seems to peak at approximately 12 classes.
An analysis of the behavior of the high light region shows diminishing return and asymptotic performance. After approximately 10 clusters, the performance does not change. Also, the performance seems to be independent of the threshold used. This robustness is important; not only is the technique fully automated over an entire dataset, but the technique is applicable over multiple datasets without requiring the parameters to be adjusted.

For the shadow class, increasing the model order creates overfitting. After a certain number of clusters, the algorithm begins to identify the target shadow as more than 1 cluster. Once this happens, the recovery of an appropriate segmentation is no longer possible. In this case, the algorithm is more sensitive to the choice of the threshold parameter. When the threshold is set very high, only pixels that have a strong class conditional probability are used in the estimate; this causes the sharp decrease in performance. As the threshold is lowered to .7, the drop-off in performance is still
evident, but less severe. This indicates that at this threshold, the algorithm performance is fairly constant in a range of model orders.

A good trade off between performance and algorithm complexity seems to be 12 clusters and a threshold of .75 for the shadow and .8 for the highlight region. From the graphs, the mean error metric for the shadow region is relatively flat from 10 to 15; this indicates that for the chosen threshold parameter, overfitting is not drastic. Since overfitting is not observed over a range of model orders, both simpler and more complex sets of figures should be well segmented using these fixed parameters.

It was shown that a model order of four was good for the 1D feature vector, but 12 was ideal for the 3D vector. Model order selection is always an issue when dealing with mixture models. In this case, extensive testing was performed to investigate the overall trends of the data with respect to model order selection. As the number of mixtures increased passed 12, little, if any, gain was observed. As the model orders increased well beyond 12, diminishing returns and over fitting were observed. Testing also showed that this technique is robust in that it is applicable over multiple images and multiple targets without changing any of the algorithm’s parameters. This contrasts sharply to thresholding and morphological techniques that require multiple thresholds for every SAR image chip.

The overall performance of the EM segmentation algorithm was quite good. The average overlap factor was as good, if not better, then the semi-automated thresholding technique for most of the test images. The overlap factor indicated that the EM segmentation was capable of obtaining a segmentation that was consistent with the shape and location of an ideally segmented image. Part of the drop in performance is because
of the algorithm’s lack of the ability to accurately segment the full shadow of the turret. Using only spatial and intensity based features, this result is not surprising. A natural extension of the EM algorithm presented here that would increase performance is the inclusion of smoothing, texture based, or multi-scale features.

### 3.3.3 Final Implementation

The EM segmentation algorithm presented above utilized a 12 class mixture model. The mixture model was defined in terms of the pixels’ intensities and x, y coordinates. Although the results were promising, this model was problematic for a number of reasons. First, the run-time for this algorithm on the entire database was quite costly. Next a mixture model was being used that included a large number of components. Since the highlight, the shadow, and the background are the only areas of consideration it would be ideal if the mixture model only contained 3 components. Those components do not necessarily have to be Gaussian, but there should be only three. Finally, this technique left edge effects as a function of background clutter. These effects are highly undesirable. Figure 3-33 shows a shadow profile from a 12 class segmented image and highlights the perturbations in its edge.
[35] showed that SAR images could be segmented into 3 classes using intensity based features if the appropriate probability density functions were known. One major drawback of their technique is the need for a training dataset and known segmentations to obtain the appropriate density functions. However, an important development in this work is the application of a smoothing technique developed in [40].

Incorporating the smoothing functions as described in [40] with the automated EM segmentation technique described in Section 3.3.2, a new technique was developed, Figure 3-34. Using this method, a 3-class segmentation can be obtained. Additionally, because of the incorporation of smoothing terms, the extracted shadow profile is smooth and free from edge effects.
Since noise in SAR images is typically nonadditive, direct smoothing techniques give poor results. A common smoothing approach is to apply a Gaussian kernel with fixed or variable variance over an image. Unfortunately, in addition to removing the noise components the Gaussian kernel blurs the desired object boundaries as well. In a technique originally developed for MRI images, [40] showed that nonlinear anisotropic diffusion produced a reliable result. In this technique, the input image is initially smoothed. Estimates of the posterior probabilities are then obtained using learned distributions, Markov fields, or in this case mixture models. Once the posterior probability is obtained, it is also smoothed using anisotropic diffusion.

Figure 3-34: Overview of segmentation method

The flowchart for the final segmentation method is given above. The initial input images are smoothed and then subjected to a mixture model. The output probabilities of the EM algorithm are again subsequently smoother. The final class estimates are obtained using a MAP classifier.
The diffusion filter is defined by a smoothing function. The smoothing function is discretized since it is applied to the image pixels. [35] has shown that the partial differential equation to define affine geometric heat flow produces the desirable smoothing effects. This PDE is given by Eq. 3.13.

\[
\frac{\partial I}{\partial t} = \left( I_{yy}^2 I_{xx} - 2 I_{y} I_{x} I_{xy} + I_{x}^2 I_{yy} \right) \tag{3.13}
\]

Where \( I_x \) is the first derivative of image \( I(i, j) \) in the \( x \) direction. The second factor is how quickly to apply the diffusion to the image. The ‘cooling’ term, \( c \), describes this phenomena. The update rule is given as Eq. 3.14.

\[
I'(i, j) = I(i, j) + c \times \text{Diffuse}_{\text{step}} \left( I(i, j), \text{map} \right) \tag{3.14}
\]

Figure 3-35 shows an initial SAR image of the T-72 and the corresponding diffusion map defined by the Eq. 3.14. Figure 3-36 shows an example of diffusion over multiple iterations.

---

![Figure 3-35: Diffusion process](image)

(a) Input image  (b) Diffusion map

The figures show a sample image and the corresponding diffusion map using the affine geometric heat flow approach for smoothing.
It is important to keep the cooling term low. Large diffusion steps can lead to undesirable segmentations. Figure 3-37 shows convergence after 100 iterations with the cooling parameter set to .4. The convergence is too fast and a suboptimal result is obtained.

Figure 3-36: Smoothing over multiple iterations

When set to low values the cooling parameter, c, allows the algorithm to converge to an optimum level. In this illustration it is set to .1.

It is important to keep the cooling term low. Large diffusion steps can lead to undesirable segmentations. Figure 3-37 shows convergence after 100 iterations with the cooling parameter set to .4. The convergence is too fast and a suboptimal result is obtained.

Figure 3-37: Rapid diffusion

When the cooling parameter is set too high, the convergence is quick, but the result is less desirable. In this example the diffusion was .4 and the image blurred with undesirable effects. Multiple smoothing steps with smaller change per step are necessary.
When observed closely, as in Figure 3-38, the values connecting the turret to the main body of the tank are seen to be more uniform with diffusion than for Gaussian smoothing. This is expected since diffusion smoothes within the boundaries defined by the natural textures of the image. A Gaussian window simply averages weighted pixel values.

![Image](image.png)

(a) Gaussian smoothing  
(b) Diffused smoothing

Figure 3-38: Effects of smoothing in the turret region of the T-72

The diffused smoothing technique more accurately smoothes the turret region within the shadow than Gaussian smoothing. The values are more uniform across the entire region for diffusion.

Figure 3-39 shows the shadow segmentation results when only 3 classes are used. Each class corresponds to the highlight, shadow, or background. The shadow outline is overlaid onto the initial T-72 SAR image. Various combinations of diffusion are used on the input image and output probabilities from the EM algorithm. Notice that diffusion of the input is required when only 3 classes are used in the mixture model. Also note that the diffusion of the posterior probabilities alleviates the effect of the background clutter. In (d) only the input image is diffused; the shadow profile is full of edge perturbations as
a function of the clutter environment. In (b) the shadow profile is smooth and free from the effects of the cluttered background.

Figure 3-39: Shadow segmentation results for various combinations of diffused input and output probabilities. (diffused input / diffused output)

The set of figures shows that the diffusion of the input image is essential when only 3 classes are used in the mixture model. When the input is not diffused the output is localized to a small low-intensity region with the shadow itself. The diffusion on the output probability elevates the effects of background clutter on the edge of the shadow profile.

The computation times for the four segmentations above were 40s, 15s, 38s, and 12s respectively. The segmentations where the input image is diffused saves significantly on computational time. This occurs because the addition of the diffusion process reduces the time to convergence of the EM algorithm; after smoothing the shadow region represents a more natural cluster. The diffusion of the posterior probabilities requires an additional 25% computational time, but alleviates the clutter edge effects. Figure 3-40 illustrates the effects of this smoothing
In the final diffusion implementation, the input image was smoothed at a cooling rate of .1 for 200 iterations. This corresponds to a settling time of 20 seconds. The number of iterations was determined by approximating the time to steady state. This value was chosen experimentally such that subsequent diffusions of the image resulted in only minimal changes to the image. The posterior was smoothed for 20 iterations at a rate of .5; this is analogous to a settling time of 10 seconds.

The EM algorithm was run using a 3 dimensional Gaussian mixture model with only three classes. Again, the three classes corresponded to shadow, clutter, and highlight. The three dimensions were intensity and (x, y) pixel location. The algorithm was iterated until subsequent iterations resulted in a change of less then .001 in the observation likelihood. For the case where that level of change was not reached, the maximum number of iterations was set to 400. Table 3-2 below shows the initialization parameters used in the EM algorithm. The covariance matrices were initialed to be diagonal.

Figure 3-40: Effects of smoothing on the posterior probability
The edge of the initial estimate of the posterior probability is severely affected by the background clutter environment of the target. Smoothing allows for these effects to be eliminated.
The final component of the algorithm was that the shadow component was constrained such that the spatial variance could not exceed a certain threshold. Without constraints, the algorithm would allow the shadow to grow out of control in cases of ill-sensed images. Figure 3-41 shows the effects of constraining the shadow variance on an ill-sensed image.

<table>
<thead>
<tr>
<th>Class</th>
<th>$\pi_i$</th>
<th>$\mu_{\text{intensity}}$</th>
<th>$\mu_x$</th>
<th>$\mu_y$</th>
<th>$\sigma_{\text{intensity}}$</th>
<th>$\sigma_x$</th>
<th>$\sigma_y$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Highlight</td>
<td>0.05</td>
<td>0.65</td>
<td>0.00</td>
<td>0.01</td>
<td>0.14</td>
<td>0.03</td>
<td>0.03</td>
</tr>
<tr>
<td>Shadow</td>
<td>0.05</td>
<td>0.15</td>
<td>0.00</td>
<td>-0.25</td>
<td>0.14</td>
<td>0.03</td>
<td>0.03</td>
</tr>
<tr>
<td>Background</td>
<td>0.90</td>
<td>0.30</td>
<td>0.00</td>
<td>0.00</td>
<td>0.14</td>
<td>0.32</td>
<td>0.32</td>
</tr>
</tbody>
</table>

Figure 3-41: Effect of spatial variance constraints on shadow class

The variance of the spatial variation of the shadow class was limited. In some ill-sensed images the shadow can grow uncontrollably and incorporate an entire clutter region within an image. When the variance is limited, the shadow stays localized. There are still some artifacts from the ill-sensed image, but they are greatly reduced.

From the figure above, it is clear that constraining helps to eliminate some unwanted effects, but still does not result in a completely reliable segmentation. The
shadow in this, and other similar images, is not discernable. However, the number of unreliable segmentations is significantly reduced using the constrained, diffused, EM approach. Some of the images in the database are simply ill-conditioned and therefore will not result in a good shadow profile regardless of the algorithm used.

Table 3-3 below shows a breakdown of performance for various targets in the MSTAR database with respect to shadow segmentation. The number of chips that were rejected based on poor segmentations is reported as a function of the 5 different target types and 2 depression angles. Undesirable images were those that did not obey the same general pattern as their neighbors and appeared to have a ‘wandering shadow.’ A wandering shadow is one that is not locked onto the target shadow and has branching components due to clutter or other ill sensed features. The trimming statistics were obtained by manually assessing the database. In total, 5% of the 15° data and 4.1% of the 17° data was considered to have a non-usable shadow. Figure 3-42 shows an example of proper segmentation for each of the 5 targets under consideration.

<table>
<thead>
<tr>
<th>Target</th>
<th>Depression Angle</th>
<th>15°</th>
<th>17°</th>
<th>15°</th>
<th>17°</th>
<th>15°</th>
<th>17°</th>
<th>15°</th>
<th>17°</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Total number of SAR chips</td>
<td>195</td>
<td>256</td>
<td>274</td>
<td>298</td>
<td>274</td>
<td>299</td>
<td>274</td>
<td>299</td>
</tr>
<tr>
<td>2</td>
<td>Non-valid segmentations</td>
<td>13</td>
<td>1</td>
<td>19</td>
<td>29</td>
<td>3</td>
<td>5</td>
<td>4</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>Percentage discarded</td>
<td>6.7%</td>
<td>0.4%</td>
<td>6.9%</td>
<td>9.7%</td>
<td>1.1%</td>
<td>1.7%</td>
<td>1.5%</td>
<td>3.3%</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 3-3: Trim statistics for shadow segmentation
3.4 Conclusions

Pre-processing of the radar images is necessary to ensure that the subsequently developed algorithms have well conditioned inputs. In real-world radar systems a SAR image will contain information from a large scene. The scene will be analyzed and potential targets within that scene will be detected and isolated. These single target chips are the data being considered. In order to ensure that there is a pixel-to-pixel correlation between chips from multiple looks, registration is needed. Since the larger scenes contain more information, the registration of these images is the most practical solution. However, if these scenes are not available additional processing of the individual chips is required.

The registration methods described in this chapter focus on cases that have only single-target chips available for processing. Moreover, it is assumed that the images can be rotational corrected with the known aspect angle of the individual radars. Registration is therefore a problem of calculating an optimal translation vector. The first method uses the centroid of the scattering centers to estimate the center of rotation of the target. This
method works well for large translations, but results in errors on the magnitude of a few pixels. The assumption that the centroid of the scattering centers matches to the target’s rotational center does not hold for all targets.

The second method of registration that was implemented utilized the correlation between images of similar aspect. This technique works well for small numbers of images. For larger databases, multiple initial templates are needed to combat propagation of errors in alignment. Once the images are registered the subsequent algorithms can perform pixel-to-pixel operations for multi-look processing.

The final pre-processing step that was considered was segmentation. Segmentation is needed for single SAR chips to identify the highlight region (the reflectivity of the targets’ scattering centers), the target shadow (a back-projection of the targets’ profiles caused by occlusions), and the background clutter. Since the intensities of the highlight region does not vary drastically from image-to-image or target-to-target a fixed threshold can be used as reliable estimate for this region. The shadow and background clutter have variable statistics and therefore require more advanced processing.

Initially, a normalized thresholding technique with morphological operations was used to extract the target shadows. Although this technique can achieve desirable results it requires many parameters to be manipulated for various sets of images. To achieve more automation, a Gaussian mixture model was applied. Each pixel was represented as a 3 dimensional vector in an $N$ class space. The pixel dimensions corresponded to its intensity and spatial location. When the raw image was processed, 12 classes were needed to adequately model the image. The profile of the resulting segmentation
exhibited perturbations in the edge as a function of the background clutter. This is highly undesirable since the shadow should only be a function of the target’s features not the ground’s.

An anisotropic smoothing technique was applied to help smooth the image. After smoothing, the input data could be clustered using a 3 class mixture model – one class each for highlight, shadow, and background. Additionally, the smoothing of probability estimates within the algorithm eliminated the effects of the clutter. The final implementation produced reliable segmentations across a range of targets and multiple depression angles.
Chapter 4

Highlight Analysis

4.1 Problem Overview

As described in Chapter 3 there are two areas of interest within a radar image that are useful for extracting information about a target vehicle – the highlight region and the shadow region. Traditional algorithms focus on the scattering centers within the highlight region for classification \[1\] \[16\] \[17\]. Although classification is important, it is not the only type of information extracted from the images. Change detection algorithms are commonly used in surveillance to identify when vehicles have left or entered the battle space. Tracking algorithms can be used to monitor suspicious vehicles \[41\]. Characterization algorithms are used to identify various structures that are attached to a specific object \[2\]. The common thread between all of these algorithms is that they operate on the scattering centers seen in the radar image.

In a single look case, the algorithms above would take the single input image and process it accordingly. The multi-look case, however, presents a challenge to many of those same algorithms since each of the individual images contains different information. The goal, as illustrated in Figure 1-1, is to take multiple images and combine them such that the new single image contains the information from the multiple looks.

This chapter is dedicated to the concept of image fusion. The main objectives are to assess multi-look image fusion capabilities and further them by incorporating some
application specific information. A completely new approach to fusion is not developed, but rather a novel extension of existing capabilities that aid in the problem of NCTR for SAR and ISAR imagery [42]. Section 4.2 outlines multiple methods for evaluation of image content. Since there is no specific application tied to fusion (classification, change detection, etc...), this evaluation is important in determining the fusion capabilities. The methods examined for fusion are presented in Section 4.3. Section 4.4 details the results of various fusion methodologies and their applications in practical situations.

4.2 Evaluation Methods

In order to assess the fusion algorithms and corresponding outputs, it is important to quantify the quality an image. The analysis techniques are evaluated using a variety of traditional error metrics including, but not limited to, mean squared error, SNR, correlation, and algorithm complexity. Two unique error metrics are also formulated – a modified receiver operating characteristics curve (ROC), and a power ratio. These tools were created to help evaluate image quality and thereby identify the strengths and weaknesses of the fusion algorithms. The following two sections outline these evaluation methodologies.
4.2.1 Receiver Operating Characteristics

To begin, a simple error image metric was investigated. An absolute error image is simply defined as the absolute value of the difference between the true image and the estimated image, Eq. 4.1.

\[ e(i, j) = \left| I(i, j) - \tilde{I}(i, j) \right| \quad [4.1] \]

The error image conveys information about the location of the errors within the estimation, but does not provide for a direct quantitative comparison for many different images. Therefore, the total squared error (TSE) of an image with size \( M \times N \) was calculated. The TSE is defined in Eq. 4.2.

\[ e_{TSE} = \sum_{i=1}^{M} \sum_{j=1}^{N} \left[ I(i, j) - \tilde{I}(i, j) \right]^2 \quad [4.2] \]

Although the TSE contains information about quality compared to a singular image, it still does not convey the overall performance for a given fusion / detection scheme. Therefore an ROC curve was utilized in order to quantify, in terms of probability, the performance of a system under a variety of operating conditions.

A receiver operating characteristic curve (ROC) is a classic plot used in radar processing to convey the performance of a system. In a typical ROC analysis, the probability of detection of a target is reported as a function of probability false alarm [20]. This concept can be applied directly to image pixels to determine the quality of information within the image. Under a binary hypothesis, the ideal object can be defined by \( \delta_{\text{object}}(i, j) \). The pixel value is ‘1’ where the target exists and ‘0’ where no target...
features are detected. An estimate of the binary object image is made from the intensity-valued test image $I(i, j)$ according to the threshold $\gamma$, Eq. 4.3.

$$I'(i, j) = \begin{cases} 1 & I(i, j) > \gamma \\ 0 & I(i, j) < \gamma \end{cases}$$ \hfill [4.3]

Next, the pixel based statistics can be calculated. The probability of detection for the test image can then be calculated as Eq. 4.4, and the probability of false alarm can be calculated as Eq. 4.5.

$$p_D^I(\gamma) = \frac{\sum_{i=1}^{M} \sum_{j=1}^{N} \delta_{object}(i, j) \times I'(i, j)}{\sum_{i=1}^{M} \sum_{j=1}^{N} \delta_{object}(i, j)}$$ \hfill [4.4]

$$p_{FA}^I(\gamma) = \frac{\sum_{i=1}^{M} \sum_{j=1}^{N} (1 - \delta_{object}(i, j)) \times I'(i, j)}{MN - \sum_{i=1}^{M} \sum_{j=1}^{N} \delta_{object}(i, j)}$$ \hfill [4.5]

The above quantities are essentially ratios of areas within the images. This can be visualized using the example in Figure 4-1. The probability of detection is calculated as the area correctly detected in the estimated image, white region in 4-1(d), divided by the total target area, white area 4-1(a). Similarly, the probability of false alarm is the area incorrectly detected in the estimated image, white region in 4-1(c), divided by the total non-target area, black region in 4-1(a).
As the threshold $\gamma$ is varied, $I'(i,j)$ will also vary. Figure 4-2 illustrates the method by which a sample image is thresholded to create multiple binary hypotheses. Eq. 4.4 and Eq. 4.5 are then used to generate multiple points on the ROC curve for the given test image’s multiple operating points. Figure 4-3 illustrates two example ROC curves. Since the probability of detection should be maximized for a fixed probability of false alarm, an improvement factor can also be defined between 2 ROC curves.

As the threshold $\gamma$ is varied, $I'(i,j)$ will also vary. Figure 4-2 illustrates the method by which a sample image is thresholded to create multiple binary hypotheses. Eq. 4.4 and Eq. 4.5 are then used to generate multiple points on the ROC curve for the given test image’s multiple operating points. Figure 4-3 illustrates two example ROC curves. Since the probability of detection should be maximized for a fixed probability of false alarm, an improvement factor can also be defined between 2 ROC curves.

---

Figure 4-1: Areas of interest for image evaluation

The binary target image, (a), is defined absolutely and used for evaluating other image estimates. The area within an estimated image, (b), is defined as either a false alarm, (c), or correctly detected, (d). A false alarm is where the estimate outputs a ‘1’ where the true pixel state is a ‘0’. A correctly detected pixel is when the estimate is a ‘1’ and the true state is also a ‘1’.

---

Figure 4-2: Concept of multiple operating points for an intensity-valued image

Applying a sliding threshold to an intensity-valued image allows for multiple binary estimate images. Each image is evaluated to create multiple operating points on the ROC.
For comparison purposes it is important that the probability of false alarm be fixed. This probability, however, is a function of the thresholded image. Therefore, two ROCs will not necessarily have common fixed points for comparison. Interpolation and resampling of the ROC is performed. The resampling is performed at regular intervals such that multiple ROC curves can be easily compared.

**4.2.2 Target Power Scale Space**

As described in the previous section, a binary thresholding technique was used to evaluate images over multiple operating points and thus generate an ROC for a given image. This technique relies heavily on the ability to compare a thresholded image to a known binary target. That type of technique is mostly applicable to SAR images. When the imagery depicts isolated point scatterers, as is often the case with ISAR images, the ROC method is no longer an accurate measure of image information.

![Figure 4-3: Concept of ROC curve](image)

The ROC curve conveys the overall quality of a test image. Images with high area under the curve are statistically more similar to the ideal image, and are therefore ‘better’. Two curves can be compared by assessing the probability of detection under a fixed probability of false alarm.
Instead, a target power scale space (TPSS) representation is presented here. For an image with known scattering centers this method provides the average noise power in non-target pixels and the average signal power for target pixels. Multiple operating points are generated by varying the desired resolution of the ‘ideal’ point scatterer. The ratio of pixel values, as subsequently derived, is equivalent to an SNR for a scattering center of a particular resolution. The following derivation details the TPSS methodology.

In an image, a single pixel point scatterer is defined by Eq. 4.6.

\[ I_{\text{target}}(i, j) = \delta(i - m, j - n) \]
\[ m \equiv \text{Target center wrt x-axis} \]  \[ n \equiv \text{Target center wrt y-axis} \]  \[ 4.6 \]

In order to vary the pixel-resolution of the scattering center, a kernel function is centered at \((m, n)\). The kernel function \( K(i, j, \Delta_x, \Delta_y) \) has two delta parameters that vary the resolution in the \(x\) and \(y\) axis directions respectively. Various kernel functions can be applied to gain different interpretations of the TPSS model: a Gaussian kernel represents a scale-space blurring model; a circular kernel represents a Euclidean distance metric; and a rectangular kernel represents a pixel-space interpretation of impulse scaling. For the remaining derivations a rectangular-pulse kernel was used. A rectangular-pulse kernel is defined by Eq. 4.7.

\[ \Pi \left( \frac{i}{I}, \frac{j}{J} \right) = \begin{cases} 1 & \left( |i - m| < \frac{I}{2} \right) \& \left( |j - n| < \frac{J}{2} \right) \\ 0 & \text{else} \end{cases} \]  \[ 4.7 \]
The figure below, Figure 4-4, illustrates a 40x40 image with three arbitrary point scatterers. The scale is varied from 1x1 to 16x16. The white regions represent the target point scatterer regions.

![Figure 4-4: Rectangular scale-spaces kernels for 3 arbitrary point scatterers](image)

Three arbitrary point scatterers are represented in the TPSS model using a rectangular-pulse kernel function. The kernel images provide a basis for interpreting the image power at the various locations within the image.

From the regions defined in the kernel space, two approximations of the image are subsequently calculated - target power and noise power. For this implementation, it is assumed that in an image the image pixel value gives a direct measurement of the power of that pixel; the power is normalized within the interval [0 1]. Using rectangular-pulse kernels, the approximate power in an image generated by the point scatterers can be calculated as Eq. 4.8.

\[
P_{\text{target}} = \sum_{i=1}^{M} \sum_{j=1}^{N} \prod \left( \frac{i}{I}, \frac{j}{J} \right) \times I(i, j) \tag{4.8}
\]

And the approximate noise power is defined by Eq. 4.9:

\[
P_{\text{noise}} = \sum_{i=1}^{M} \sum_{j=1}^{N} \left( 1 - \prod \left( \frac{i}{I}, \frac{j}{J} \right) \right) \times I(i, j) \tag{4.9}
\]

The average power values are calculated for various scales of \( I \) and \( J \). The quantities can be normalized by the number of pixels in the given region to obtain an
average pixel value / power level for both the target and non-target regions of a given scale, Eq. 4.10 and Eq. 4.11.

\[
< P_{\text{target}}(\text{scale}) > = \frac{P_{\text{target}}}{\sum_{i=1}^{M} \sum_{j=1}^{N} \prod \left( \frac{i}{I}, \frac{j}{J} \right)} \tag{4.10}
\]

\[
< P_{\text{noise}}(\text{scale}) > = \frac{P_{\text{noise}}}{\sum_{i=1}^{M} \sum_{j=1}^{N} \left( 1 - \prod \left( \frac{i}{I}, \frac{j}{J} \right) \right)} \tag{4.11}
\]

A plot of \(< P_{\text{target}}(\text{scale}) >\) vs. \(< P_{\text{noise}}(\text{scale}) >\) for multiple scale values illustrates the relationship of image quality for a particular image over a variety of target conditions. If a small subset of target conditions is evaluated (varied kernel sizes), the maximum ratio of target power to noise power is analogous to the optimal SNR within the considered target operating region.

### 4.3 Implementations

#### 4.3.1 Baseline Tests

In order to evaluate the effectiveness of image fusion, the algorithms need to be compared and tested for a baseline case. A spatial average is considered as the default for baseline testing. There are a number of reasons that this is considered to be base. First, it is the most basic of combinatorial structures [10]. Secondly, for a multi-class classification problem, such as the one described by the binary target hypothesis, a spatial average represents an unbiased linear classification boundary. Assume that for a given
pixel its value, \( x \), is subjected to a threshold; the pixel either belongs to the target class or non-target class, ‘1’ or ‘0’ respectively. This is given by Eq. 4.12.

\[
x \lesssim \gamma_n
\]

[4.12]

Next, a 2-D case can be considered. If each data point is projected onto the line \( x_1 = x_2 \) a linear classifier can be defined. The projection and thresholding of the data is therefore given by Eq. 4.13.

\[
\sqrt{2 \left( \frac{x_1 + x_2}{2} \right)^2} \lesssim \gamma_n
\]

[4.13]

Since the actual value of the threshold is incidental, Eq. 4.13 can be rewritten as Eq. 4.14. This represents a spatial average of the two input SAR/ISAR images; this is used as the default case for baseline testing.

\[
x_1 + x_2 \lesssim \gamma_n
\]

[4.14]

### 4.3.2 Fusion Rules

A variety of techniques are proposed for evaluation in order to fuse the SAR/ISAR images. Four such techniques under investigation are averaging, probabilistic evaluation, component analysis, and wavelet decomposition. All of these fusion techniques attempt to generate a single intensity image that best describes the data in the initial input images.
The simplest designation of an image is again a pixel-wise binary description. In this case, the ideal target definition is simply ‘1’ where the target has significant scattering centers and ‘0’ otherwise. The 2-dimensional radar image is defined over a geographical region of interest in the XY-plane. The pixel values for a normalized radar image over this swath area can be defined in the range \([0, 1]\) for an image of size \(M \times N\). The ideal target image, also given by size \(M \times N\), is assumed to have a one-to-one spatial mapping with the radar image. The target image can therefore be defined over \(\{0, 1\}\), where a ‘1’ denotes the presence of the target and a ‘0’ denotes no target. These two hypotheses can be represented as \(H_1\) and \(H_0\), respectively. The goal of the fusion problem is to define an estimate of the target over the \(M \times N\) image swath.

Using the binary definition, a consideration common in many of the fusion methods is the probably of observing particular pixel values in the test image given each of the two hypotheses. The likelihood that a particular pixel \((i, j)\) has value \(k\) is given it is in the target region is given by Eq. 4.15.

\[
p(I(i, j) = k \mid H_1) \tag{4.15}
\]

Similarly, the probability a particular pixel \((i, j)\) has value \(k\) is given that it is not in the target region is given by Eq. 4.16.

\[
p(I(i, j) = k \mid H_0) \tag{4.16}
\]
4.3.2.1 Averaging Rules

As described in Eq. 4.14, the baseline test used for fusion is a spatial average. Initially, the linear decision boundary is given and corresponds to a line as described in Eq. 4.17,

\[
x_1 + x_2 < T \\
x_2 = (-1)x_1 + T
\]

[4.17]

where varying \( T \) changes the location of the boundary. The equations above can be generalized such that the slope of the line is \( m \). Therefore a more general decision boundary is given in Eq. 4.19.

\[
x_2 = (-m)x_1 + T
\]

[4.18]

\[
x_2 + mx_1 < T
\]

[4.19]

Where \( 0 \leq m \leq \infty \). For the given signal space, the threshold can vary over a range of operating characteristics: \( T_{\text{min}} \leq T \leq T_{\text{max}} \), where the extremes are given in Eq. 4.20 and Eq. 4.21. Each operating point (each different threshold) represents a single point on the ROC curve.

\[
T_{\text{min}} = (0) + m(0) = 0
\]

[4.20]

\[
T_{\text{max}} = (1) + m(1) = 1 + m
\]

[4.21]

Figure 4-5 describes the physical interpretation of the decision boundary. When \( m=1 \), the input dimensions are equally weighted; the decision boundary corresponds to that of an average. When the slope is changed, one of the two input images are weighted
more heavily. For a slope of 0, the decision boundary is a horizontal line. A horizontal line means that only the information in the $x_2$ direction (image 2) contributes to the decision. Conversely, a slope of $\infty$ corresponds to a vertical line, and therefore only the data in image 1 is being used in the decision process.

4.3.2.2 Probabilistic Rules

Next, a non-linear decision boundary is applied using the probability of pixels’ intensities. This decision boundary is derived from Bayes conditional risk [43]. Risk allows some errors to be penalized more then others, Eq. 4.22.

$$R(\alpha_i \mid x) = \sum_{j=1}^{c} \lambda(\alpha_i \mid \omega_j) P(\omega_j \mid x)$$

where the posterior probability given by Bayes rule is given in Eq. 4.23.
Applying these quantities to the given 2-category problem the risks are given in Eq. 4.24 and Eq. 4.25.

\[
P(\omega_j | x) = \frac{P(x | \omega_j)P(\omega_j)}{p(x)} \tag{4.23}
\]

The fundamental rule is to decide \( H_0 \) if \( R(H_0 | x) < R(H_1 | x) \). Typically, correct decisions are not penalized. Similarly, the posterior probabilities can be expressed using Bayes rule. The evidence can be canceled from both sides. Similarly, the risk and a priori probability can be lumped into a single threshold leading to the following equivalent rule, Eq. 4.26. This decision rules is also comparable to Eq. 4.27.

Decide \( H_0 \) if \( T \times p(x | H_0) > p(x | H_1) \)
Otherwise decide \( H_1 \) \[4.26\]

\[
T \times p(x | H_0) > (1 - T) \times p(x | H_1) \tag{4.27}
\]

A likelihood ratio test can also be defined as the decision rule for each pixel within the \( M \times N \) image [44]. Using the likelihood functions defined from available data, the log-likelihood ratio test can be defined by Eq. 4.28.

\[
\ln \Lambda(I(i, j)) = \ln \left( \frac{p(I(i, j) | H_1)}{p(I(i, j) | H_0)} \right)_{H_i}^{H_i} \ln(\eta) = \gamma \tag{4.28}
\]
Eq. 4.28 depicts the case for a single observation, a single radar system. When the detection system is expanded to two radar images, the images are assumed to be independent and identically distributed. As such, the likelihood ratio becomes the pixel-wise product of the individual likelihoods defined for each image. This allows for the performance of multi-look systems to be evaluated.

### 4.3.2.3 Component Analysis

Two common versions of component analysis techniques are examined: principal component analysis (PCA) and independent component analysis (ICA). The main goal of PCA is to determine a set of basis vectors that completely describe the information in a data matrix, $D$. PCA seeks a projection such that the data is best represented in the least squares sense, Eq. 4.29. PCA keeps the dimensionalities with the largest variance.

$$J_0(x_0) = \sum_{k=1}^{n} ||x_0 - x_k||^2$$  [4.29]

PCA is also known as the Karhunen-Loève transform (KLT) or the Hotelling transform. The procedure for obtaining the PCA is outlined below [43]:

- Organize the data into column vectors, to obtain a $[d \times n]$ matrix, $D$.
- Find the empirical mean along each dimension, the result is a $[d \times 1]$ empirical mean vector, $M$.
- Subtract the empirical mean vector $M$ from each column of the data matrix $D$. Store mean-subtracted data $[d \times n]$ matrix in $S$.
- Find the empirical covariance $[d \times n]$ matrix $C$ of $S$. $C = S' S$.
- Compute and sort by decreasing eigenvalue, the eigenvectors $V$ of $C$.
- Save the mean vector $M$. Save the first $k$ columns of $V$ as $P$. $P$ will have dimension $[d \times k]$. $1 \leq k \leq d$
The main goal of ICA is to represent a multi-dimensional random vector as a linear combination of independent non-Gaussian random variables [45]. This model can represent the observed random vector, \( x \), as a function of the basis vectors, Eq. 4.30.

\[
x = \sum_{k=1}^{n} a_k s_k
\]  

[4.30]

The basis vectors form the mixing matrix \( A \). Thus the signal can be represented as Eq. 4.31.

\[
x = As
\]  

[4.31]

Where \( s \) is an n-dimensional random vector that contains the independent source signals. One of the common approaches used to calculate these quantities is FastICA. FastICA is an iterative fixed-point scheme for calculating ICA developed by A. Hyvärinen [46].

The concept of “non-Gaussian is independent” is the driving force behind the ICA algorithm. According to the central limit theorem, the sum of independent random variables tends towards a Gaussian distribution. The sum of two independent random variables has a distribution that is more Gaussian than the original two distributions. Thus when the independent components themselves are desired the non-Gaussianity needs to be maximized. When considering a linear combination as given above, the vector \( \mathbf{w} \) that maximizes the non-Gaussianity of \( \mathbf{w}^T \mathbf{x} \) necessarily corresponds to one of the independent components.

Before performing ICA, the initial data needs to be centered and whitened. PCA can be used as a preprocessing step to obtain these necessary conditions. The FastICA algorithm as given by Hyvärinen maximizes the non-Gaussianity of \( \mathbf{w}^T \mathbf{x} \) [45]:
(i) Choose an initial weight vector \( w \)
(ii) Let \( w^+ = E \{ x g(w^T x) \} - E \{'g'(w^T x)\}w \)
(iii) Let \( w = w^+/||w^+|| \)
(iv) If not converged, go back to (ii)

The minimization of Gaussianity is used in FastICA but is equivalent to the minimization of mutual entropy; the optimization in this implementation is based on a particular measure of non-Gaussianity. Two popular measures are kurtosis and negentropy. The estimations of both have positive and negative tradeoffs. Based on the maximum-entropy principle of estimating negentropy, the term in (ii) emerges as a method of optimization. This term is dependent on a nonquadratic nonlinearity function, \( g(\cdot) \). The choice of this function is important for approximation accuracy and robustness. Common, possible functions are Eq. 4.32, Eq. 4.34, and Eq. 4.33.

\[
g(x) = x^3 \quad \text{[4.32]}
\]

\[
g(x) = \tanh(a_i x) \quad \text{[4.33]}
\]

\[
g(x) = x e^{(-a x^2/2)} \quad \text{[4.34]}
\]

The algorithm, FastICA, is not the only method of calculating the independent components, but has proven computational efficiency and is widely used. JADE and InfoMax are two alternative methods. Studies have shown they all converge to the same approximate solutions since their derivations show the same global optimums.
4.3.2.4 Wavelets

The final fusion method that is investigated is 2-D wavelet fusion. The basic algorithm for wavelet fusion consists of three steps – image decomposition, fusion mapping, and image reconstruction. The main idea behind decomposition is that the image can be deconstructed into multiple levels; each level of the decomposition represents different frequency content within the image. The mapping allows a mechanism by which the wavelet coefficients of multiple images can be fused together. The reconstruction process allows an image to be formed from the fused coefficients. This process is widespread in current literature and has been used successfully for a variety of applications [47] [48] [49] [50].

At each level during decomposition, the image is filtered into an approximation image and detail images. This process is illustrated in Figure 4-6. The approximation image represents a low pass filtered image. The high frequency content is located in the three detail images. For multi-level decomposition, the approximation image is subsequently decomposed again.
In the figure, ‘Lo_D’ is the decomposition low pass filter and ‘Hi_D’ is the high pass decomposition filter. The filters coefficients are determined by the wavelet being used. Figure 4-7 is an illustration of the Daubechies 3 wavelet and the associated filters.

Figure 4-6: 2-D wavelet decomposition [51]

The figure is an illustration of the algorithm used for wavelet decomposition. The filter coefficients are a function of the wavelet used. For multi-level decomposition the approximation image is iteratively subjected to the decomposition.
For image fusion, the decomposition is performed for each of the images. The multiple sets of coefficients are then mapped using a fusion rule. Typically, two fusion regions are defined. The first rule is used for the approximation image and the second rule is applied to the detail images. Some typical fusion rules include, min, max, mean, and weighted averaging. The inverse discrete wavelet transform (IDWT) is then used to formulate the fusion image from the fused coefficients. Figure 4-8 illustrates the mapping process and Figure 4-9 illustrates the steps in calculating the IDWT. ‘Lo_R’ is the reconstruction high pass filter and ‘Hi_R’ is the high pass reconstruction high pass filter.
The figure illustrates the deconstruction, mapping, and reconstruction procedure for wavelet image fusion. Alterations to the mapping functions allows for various effects with the wavelet fusion technique.

**Figure 4-8: Mapping procedure for wavelet fusion**

**Figure 4-9: 2-D wavelet reconstruction [51]**

The figure is an illustration of the algorithm used for wavelet reconstruction. The filter coefficients are a function of the wavelet used. For multi-level decomposition the approximation image is iteratively subjected to the reconstruction.
The most critical step in the wavelet fusion algorithm is the data mapping rule. Figure 4-10 illustrates how different logic rules can be applied to a two input binary system. In (a) the target region is defined over a specific region; the concepts of false alarm and missed detection are also illustrated. In (b) two different signals are input. Each has different areas of missed detection and false alarms. A binary OR rule, an AND rule, and a combination rule are applied. The combination rule uses the AND rule in the non-target region and the OR rule in the target region.

Figure 4-10: Binary fusion rules
The image illustrates an example of binary data fusion. The target region is described as a ‘1’. The two sample inputs have various false alarms and missed detection regions. The OR rules increases the probability of detection, but has a higher rate of false alarms. Conversely, the AND rule lowers the number of false alarms at the expense of missed detections. The combination rule uses the strength of each to maximize the potential performance of the fusion.

An OR rule identifies the target region when either of the two inputs has identified this region; the drawback is that the output has the same number of false alarms.
as the two inputs combined. An AND rule eliminates false alarms in all cases except when both inputs exhibit false alarms; the drawback is that there are the same number of missed detections in the target region as the two inputs combined. It is clear that an OR rule is desirable in the target regions and an AND rule is desirable in the non-target regions; this is precisely the combination rule that is illustrated.

The probability of detection is related to the probability of a missed detection according to Eq. 4.35. The probability of false alarm is related to the probability of a correct no decision according to Eq. 4.36.

\[
p_d = 1 - p_m \tag{4.35}
\]

\[
p_n = 1 - p_{fa} \tag{4.36}
\]

The fused probabilities of detection and false alarm can be calculated using the OR rule, Eq. 4.37 and Eq. 4.38, and the AND rule, Eq. 4.39 and Eq. 4.40.

\[
p_{d}^{OR} = 1 - (p_{m1} \times p_{m2}) \tag{4.37}
\]

\[
p_{fa}^{OR} = 1 - (p_{n1} \times p_{n2}) \tag{4.38}
\]

\[
p_{d}^{AND} = p_{d1} \times p_{d2} \tag{4.39}
\]

\[
p_{fa}^{AND} = p_{fa1} \times p_{fa2} \tag{4.40}
\]
If the AND rule in used in the non-target region and the OR rule in the target region, the combinatorial rule probabilities are given by Eq. 4.41 and Eq. 4.42:

\[ p_d^C = 1 - (p_{m1} \times p_{m2}) \]  \[4.41\]

\[ p_{fa}^C = p_{fa1} \times p_{fa2} \]  \[4.42\]

Assuming that both inputs sources have the same probability, the following graphs, Figure 4-11, depict the performance of the three rules for a two input case. Notice that the AND rule suppresses the probability of false alarm while the OR rule increases the probability of detection. In both scenarios the combination rule exhibits both behaviors and simultaneously lowers the probability of false alarm and increases the probability of detection.

![Graphs](a) Scenario #1 (b) Scenario #2)

**Figure 4-11: Effects of fusion**

The graph depicts the effects of three binary combination rules for a two input system. The system inputs have unique ROCs for the two different scenarios. In both cases the combination rule is far superior to the other rules as it both suppresses false alarms and increases the probability of detection.
The main problem with the combination rule as described above is that the system assumes knowledge of the target region. If the target region is already known then processing of the radar signatures is unnecessary. Instead, it is necessary to estimate this region from the data.

The OR and AND rules have the same output when the inputs are the same, \(\{0, 0\}\) or \(\{1, 1\}\); the output is only different when the input values are different, \(\{0, 1\}\) or \(\{1, 0\}\). This indicates that when the input signals are different one of the two rules must be selected. If it is assumed that the input images are not binary but intensity images, high disparity in the two input images might indicate a possible rule change. The mapping is dictated by the disparity as given in Eq. 4.43. In an intensity based image a MIN operation is equivalent to the AND rule and a MAX operation is equivalent to the OR rule.

\[
|I_1 - I_2| > \beta_{H_1} \\
|I_1 - I_2| < \beta_{H_0}
\]

[4.43]

To illustrate the concept of disparity, it is assumed that an AND rules is always applied. High disparity will indicate possible cases when an OR should be applied. False alarms are typically randomly distributed within image. In these cases, neighboring regions will not show the same disparity as the false alarm region. However, when target self-occlusion occurs, large groups of neighboring pixels will exhibit high disparity. Therefore morphologic filtering can be used in combination with the disparity technique to indicate necessary changes in the fusion rule structure.
Figure 4-12 illustrates an example of how target self-occlusion is combated using this disparity technique. In the example, the regions of disparity are first calculated. The small isolated incidents of disparity are assumed to be false alarms – the AND rules is applied. The larger region of high disparity is assumed to be cause by target occlusion – the OR rule is applied. By using this disparity measure the combination rule can be applied without prior knowledge of the target region.

The remainder of the section outlines the fusion method implemented for the radar imagery. Each processing block illustrates the adjustable parameters or methods that most impact the outcome. Figure 4-13 outlines the data pre-processing steps. The
The first step is the formation of the individual ISAR images. Next, the images are decomposed using wavelet decomposition.

Figure 4-13: Data pre-processing overview
In the data pre-processing steps, the ISAR data is composed into an image and then subsequently decomposed using the appropriate wavelet functions.

After the initial images from both radars are decomposed, a comparison is performed. This comparison determines the information content of the two images. This can be performed according to an image disparity measurement, an entropy measurement, or a maximum likelihood ratio test (MLRT). Next, the comparison image is uniformly thresholded. The threshold value can be varied according to the desired output. Morphological operations are used to ensure consistency in the fusion mapping; the minimum and maximum rules are applied according to the desired mapping. Figure 4-14
illustrates the overall flow of the data fusion step. Finally, the fusion image is reconstructed using the individual wavelet images, Figure 4-15.

Figure 4-14: Data fusion rules

In the data fusion step, the decomposed images are compared using an image disparity measure. This disparity measure is used in determining the fusion map. The min and max operations are then used to fuse the two sets of wavelet coefficients.
4.4 Results & Analysis

4.4.1 Ray-tracing model experiments

The fit of a Gaussian distribution, signal-to-noise ratio, and intra-radar angle are all parameters that are investigated through various simulations of the ray-tracing model. In the first simulation, test object 1 is imaged at 5 dB for a radar location of (2, 0.5, 8). The results are averaged for the test object being rotated from 0° to 175° in 5° increments. The simulation in Figure 2-4 represents the data with no sub-regions. A least squares fit of a Gaussian curve is calculated. From the figure it is clear that this is a good first approximation.
Figure 4-16: Pixel distributions
The figure shows that the averaged image statistics are approximately Gaussian across the entire image.

Figure 4-17 uses 2 radial and 2 angular sub-regions. Three ROCs are being plotted - a theoretical ROC, a radar sub-block ROC calculated via the LRT, and one from thresholding the image. For 4 total sub-regions the overall performance of the system is seen to be high. The full likelihood ratio test is seen to be only slightly lower than the calculated theoretical value. This means that the assumption of Gaussian distributed values within the sub-regions is valid. Figure 4-18 confirms that the raw pixel values, averaged over the object rotations, are distributed in a Gaussian manner. Since this is the case, the thresholding method and the LRT match almost exactly.
Figure 4-17: ROC for 4 total sub-regions

The sub-block processing with 4 total regions seems to be advantageous over a single block. However, even though the average image statistics are approximately Gaussian, the individual image distributions deviate from this assumption and thus the ROCs are slightly less than the theory suggests.

Figure 4-18: Sub-block distributions

Blue represents estimated Gaussian for H₀, Cyan represents actual distribution for H₀, Red represents estimated Gaussian for H₁, and Green represents actual distribution for H₁. The average distributions are approximately Gaussian, but the realizations from a single image deviate from this assumption causing some estimation errors.
Figure 4-18 (b) illustrates that for one particular object orientation the pixel values are only roughly Gaussian. The fit is not nearly as good as the averaged values. Because of these deviations, each individual ROC will not exactly match the theoretical ROC; therefore, the averaged ROC for the LRT should be slightly lower than the theoretically calculated value. This reasoning matches with the results seen in Figure 4-17.

When the region is decimated into 4 angular by 4 radial sub-blocks, the overall system performance is seen to decrease. The decreased performance indicates that by subdividing the image, the spatially local image statistics are not as well behaved as the more general image statistics defined over larger regions spatially. The highlighted sub-block illustrates one example of how the local statistics produce a suboptimal result, Figure 4-19. Figure 4-20 shows that for the given highlighted sub-block one particular realization (b) is non-Gaussian even though the average case is roughly Gaussian (a). The effect of this is seen in both the likelihood ratio image as well as the detected object obtained for a specific threshold value, Figure 4-19. The performance of the sub-blocks is subsequently quite low.
Although the statistics vary across the image, the distributions over many of the sub-regions deviate from the Gaussian approximation in the above scenario. Since the...
local sub-region statistics do not obey the Gaussian assumption it is not desirable to divide the image into such localized sub-blocks (4x4). For subsequent simulations, the data was divided into 2 radial and 2 angular sub-regions.

Next, fusion was tested. The first simulation was used to validate assumptions made about the optimal SNR operating regions. The ROC improvement is defined as the improvement in probability of detection of the fused radar image over the probability of detection from the average of the individual radars for a fixed probability of false alarm. Figure 4-21 shows a sample of the test images ranging from -30 dB to 60dB. Figure 4-22 (a) illustrates that the biggest improvement is gained for radars with SNR around 0 dB. Figure 4-22 (b) shows that the average performance increase is most significant for low false alarm rates. Both are encouraging characteristics since these are areas where a system might operate.
Figure 4-21: Images with various SNR values
The image set shows target 1 image at various SNR values.

Figure 4-22: Improvement in the probability of detection for various SNR values
The graphs illustrate the improvement of baseline data fusion over the performance of a single image. The greatest improvement is for SNR values around 0 dB and for low probability of false alarm.
Next, the LRT was then investigated for 2 observations at 5 dB of noise and an inter-radar angle from 0° to 180°. The inter-radar angle is defined as the angle formed between two on-looking radar systems. Figure 4-23 illustrates the sample images for one of the two radars being fused. The results from fusion are reported in Figure 4-24. From the plots, the probability of detection increases as the inter-radar angle approaches 180°. The graph does not perfectly depict this trend as there is some fluctuation in the values. These fluctuations are likely due to simulation artifacts. An optimal inter-radar angle of 180° implies that the maximum improvement occurs for situations were the two radars are looking at opposite sides of the target.

Figure 4-23: Sample images for data being fused

The data represents images of target 1 at 5 dB. This is then fused with another radar at the same SNR, but different aspect angle.
The final simulation imaged target 2 over inter-radar angles from 0º to 180º. The SNR values were set to 5 dB and -3 dB. The average improvement as a function of false alarm is depicted in Figure 4-25. The improvement for -3dB SNR is more substantial then for +5dB. It can also be seen that the optimal improvement has shifted to a higher probability of false alarm as compared to test object 1. This result indicates that the more complicated geometry of object 2 is detectable, but will add additional error into the system.

When the individual radars convey a small amount of complimentary information, there are large gains by performing multi-look fusion. When the individual images already have high performance, there is a much smaller gain by having additional looks.
The tests using the ray-tracing method have illustrated many interesting points. The distribution of pixel intensities was Gaussian for large sub-regions. When the image is divided beyond 4 regions the Gaussian assumption breaks down. In order to detect and image the object, two tests were devised. The likelihood ratio test uses the Gaussian values fit from the image values, whereas the simplified test uses the image values directly. Under ideal conditions, perfectly Gaussian images, both tests will perform identically. However, the LRT outperforms the mathematically simplified test when the true underlying distributions of the images are only approximately Gaussian.

The values are not perfectly Gaussian distributed for a single realization, but the LRT forces the probabilities to be estimated using the Gaussian curve fit to the data. Since the true underlying distribution is Gaussian when averaged over many realizations, the LRT method more closely matches the optimal theoretical performance; it outperforms a simple thresholding of the raw observations. Conversely, when the

Figure 4-25: Improvements as a function of false alarm for target 2
The graphs depict the average improvement as a function of false alarm for test target 2. The improvements are greater for the lower SNR. For high false alarms in the 5 dB case there is actually a small dip in performance for fusion over the individual images. The effectiveness of fusion is a direct result of the quality of the input images.
underlying distributions are not Gaussian, the LRT underperforms and is not an ideal fusion method.

Finally, it was shown that by adding a second radar into the system, the additional observation can significantly improve the quality of the estimation being obtained. The estimation was most significantly improved when the second radar is at a 180° to the original radar. Even at sub-optimal inter-radar angles there is still significant improvement in the probability of detection for fixed false alarm rates.

4.4.2 MSTAR (linearity of the boundary)

Next, multi-look fusion was examined for the T-72 target data. To begin the analysis, two methods of target detection were again used – direct thresholding and likelihood ratios. First, the probability density functions are calculated for the appropriate regions. The probability of observing pixel values under a given hypothesis can be determined directly.

The conditional probabilities $P(x \mid H_0)$ and $P(x \mid H_1)$ were found using a frequency estimation measure. This method is a direct, unbiased estimate of the probability density functions. In order to get an accurate estimate, the number of pixels in the sample must be very large. For the T-72 target at 15° depression there are 273 images of size 138 x 139. Taking into account the regions of valid values (due to correctional rotation) the total numbers of target and non-target pixels are 158,340 and 3,921,645, respectively.
The distributions were calculated in two different ways. First, the 1D distribution was calculated from the images directly. Next, the images were preprocessed using a 10 x 10 Gaussian smoothing filter with standard deviation of 1.5. Gaussian smoothing is applied to reduce the effects of peaks at 0 and 255 for the noise and target respectively. In the smoothed images the saturation points are no longer observed, Figure 4-26.

Once the distributions were obtained, the likelihood ratio analysis and fusion techniques could be applied. Figure 4-27 is one such instance where this technique was applied. A multi-look system of 2 images was fused using the LR technique and spatial averaging. The corresponding ROCs are also reported.
The ROC’s obtained via the thresholding method and the LRT for the single SAR chip are statistically indistinguishable. Also, there is little variation in the ROC for the spatial fusion and the LR fusion. Both fusion methods, however, are seen to exhibit a substantial increase in performance over the single radar systems. Since this test only exhibits a single instance of fusion, a Monte Carlo simulation was performed. Two images were uniformly selected from the database and fused; this experiment was repeated over 5000 trials. Figure 4-28 (a) illustrates the histogram for the intra-radar angle, and (b) illustrates the improvement of the fusion algorithm over the average of the individual chips.

---

**Figure 4-27: Fusion analysis**

The fusion analysis above is for a 2 look system. The images were fused via a spatial average and a likelihood ratio. The likelihood ratio and spatial averaging perform almost identically, but both are an improvement over the initial images.
Both the likelihood ratio fusion method and the spatial average improved the image quality over the single looks. The most significant improvements in the probability of detection are for low false alarm rates; this result confirms the same findings obtained with the canonical dataset.

Next, the 2D distributions are obtained directly from the 1D distribution. Two images obtained at different look angles are assumed to be i.i.d, Eq. 4.44. This assumption holds since the distribution calculations were performed on an ensemble that contained all possible look angles. Figure 4-29 shows the 2D distributions for the original and smoothed images.

\[
p(x_1, x_2 \mid H_0) = p(x_1 \mid H_0) p(x_2 \mid H_0)
\]  

[4.44]
In the original 2-D distribution a saturation peak dominates the target distribution. Since the saturation points are biasing the distribution, smoothing was again applied to the input image. As discussed in Section 4.3 there are multiple types of boundaries that can be used in fusion. The linear boundary used in spatial averaging is compared to the conditional risk boundary in Figure 4-30. Both boundaries are illustrated for multiple operating points. The $x$ and $y$ axes in the images represent the pixel intensities of the two input images. The decision is either the pixel belongs to the target region (white) or the non-target region (black).

Figure 4-29: 2D pixel distributions

The figures represent the 2D pixel distributions of the T-72 at $15^\circ$ depression. Smoothing is applied to the input images to reduce the effects of the peaks at the tails of the distributions.
Finally, the two decision rules can be compared using ROC curves. The 1-D ROC curve is obtained by thresholding the 1-D distributions. The probability of false alarm and detection are obtained directly from integrating the appropriate target distributions. Figure 4-31 shows the theoretical ROC curves for the 1-D case, 2-D linear boundary, and 2-D non-linear boundary.

Figure 4-30: Decision boundaries
Two common types of decision boundaries are illustrated. The boundaries are shown over multiple operating points for a two look system. The white region represents the decision to classify a pixel as a target pixel, and the black region classifies a pixel as non-target.
From the figure above it appears that the nonlinear boundary is only nominally better than the linear one. The ROC for both 2-D cases are substantially better than that of the 1-D case. However, the gain in performance for using the nonlinear boundary is minimal. A set of images, Figure 4-32, was used as another illustrative test of the performance of these two boundaries.

Figure 4-31: Theoretical ROC curves
The figure illustrates the performance of the 2 fusion boundaries compared to the single look case. Again, both fusion methods improve upon a single look scenario, but give almost identical outputs.

From the figure above it appears that the nonlinear boundary is only nominally better than the linear one. The ROC for both 2-D cases are substantially better than that of the 1-D case. However, the gain in performance for using the nonlinear boundary is minimal. A set of images, Figure 4-32, was used as another illustrative test of the performance of these two boundaries.

Figure 4-32: Input test images
These two images are used as test images for multi-look fusion. They represent smoothed images of the T-72 at 15° depression.
Using the images above, Figure 4-33 shows one operating point for the linear and nonlinear decision boundaries. In (b) & (d), each pixel is represented by a 2D point in the vector space. The pixels are either correctly classified as a detection or noise pixel, or incorrectly classified as a missed detection or false alarm.

The probability of detection and false alarm for the linear and non-linear boundaries are (0.5172, 0.0024) and (0.6125, 0.0025), respectively. From this analysis it would appear that the non-linear boundary is better; however, when both methods are
operated over the entire range of operating points the performance is comparable, Figure 4-34. The illustration in Figure 4-33 is within the extremely small region of the ROCs where the non-linear conditional risk boundary is superior; generally this is not the case. Again the multi-look case significantly outperforms the single look scenario.

Figure 4-34: ROC for linear and conditional risk boundaries

There is only a very small region within the ROC where the 2D conditional risk is superior to the linear boundary. Generally, both rules again appear to be equivalent and far better then the 1D case.

Since the linear boundary is essentially equal to the optimal non-linear case in a Bayes framework, the next experiments focused on determining in what scenarios a biased linear boundary could be advantageous.

A range of input slopes, \( m \), and thresholds, \( T \), were applied to the probability distribution functions directly. The corresponding 3D ROC is displayed in Figure 4-35 (a). Figure 4-35 (b) illustrates the area under the ROC for various slopes. The negative slope reported in the graph is slope-1.
In the figure it can be seen that the optimal ROC is when the slope of the line is equal to 1. This means that without any other a priori information there is no reason to weight one image over another. The inherent characteristics of the distributions do not support a weighted linear boundary either. To further test this procedure, three cases of 2-look fusion were considered – equal information content, unequal information content, and severely disparate information content. Figure 4-36 illustrates the case were the two input images have approximately the same amount of information; the corresponding ROC is displayed. Figure 4-37 is an example of two SAR images where one is a better description of the target then the other.

In the graph the probability distributions were evaluated as a function of weighted averaging (sloped linear decision boundaries). The negative slope corresponds to the inverse of the slope not a negative slope. The optimal boundary is an equal weighting.

Figure 4-35: Evaluation of weighted averaging

(a) ROC

(b) Area under the ROC

In the figure it can be seen that the optimal ROC is when the slope of the line is equal to 1. This means that without any other a priori information there is no reason to weight one image over another. The inherent characteristics of the distributions do not support a weighted linear boundary either. To further test this procedure, three cases of 2-look fusion were considered – equal information content, unequal information content, and severely disparate information content. Figure 4-36 illustrates the case were the two input images have approximately the same amount of information; the corresponding ROC is displayed. Figure 4-37 is an example of two SAR images where one is a better description of the target then the other.
The area under the ROC, Figure 4-38, can again be used to assess the performance of the weighted boundary. Clearly in the case of equal information, the slope is still optimal around 1. The costs of weighing one image over the other decreases the performance; moreover the decay of information is equal for a bias towards either image. However, for the case of unequal information, there is a severe decline in performance as the boundary is biased towards image 1. There is little decline in
performance as it is biased towards image 2. This indicates that the information content in image 1 does not add as much in the fusion image as does image 2. Still, the optimal slope is 1.

In the final case, the test images have a large disparity in quality. The disparity in the quality of the images is so great that the weighting is clearly shifted towards the better image, Figure 4-39.
From this analysis, it was seen that when the ensemble probability distributions are used, the optimal slope of the linear decision boundary is 1. This means that each input images should be weighted equally. This decision rule is not optimal for all pairs of realizations, but should be used in most typical cases. The optimality of the rule is determined by the disparity of information between the two input images. The rule will fail when one image is so poor that it offers no new useful information content; in these rare cases fusion acts to degrade the overall quality of the output rather then improve it.

4.4.3 MSTAR (multi-radar configuration analysis)

The final analysis of a two radar system is to determine the optimal spatial configuration of the system. As with the canonical dataset, the improvement in probability of detection can be analyzed as a function of the intra-radar angle. Figure 4-39, shows the 3-D ROC and the ROC for a fixed probability of false alarm.

Figure 4-39: Two-look fusion with images of severely disparate information content

The two test images in this case have severely disparate information content. The second image has high noise and very few target features. As such it cannot improve upon the information that is already in image 1. For this extreme case there is no amount of fusion that will improve upon the data already in image 1.

From this analysis, it was seen that when the ensemble probability distributions are used, the optimal slope of the linear decision boundary is 1. This means that each input images should be weighted equally. This decision rule is not optimal for all pairs of realizations, but should be used in most typical cases. The optimality of the rule is determined by the disparity of information between the two input images. The rule will fail when one image is so poor that it offers no new useful information content; in these rare cases fusion acts to degrade the overall quality of the output rather then improve it.
The improvement seems to level off around 110° but peaks at 180°. This indicates that optimal fusion, with respect to the highlight region only, will occur for SAR images that are looking at opposite sides of a target. The results from the T-72 database coincide with those obtained using the canonical dataset.

Next, the optimal configuration for a 3 target system is investigated. The analysis focuses on determining which factors are most influential to the overall performance of the 3-look system - what is the optimal spacing of the radars? Are there specific looks that provide the best view of target specific features? The analysis framework is given in Figure 4-41. First, three SAR images with unique look angles are fused. Next, the ROC is generated according to the fused image. Then a Neyman-Pearson criteria is applied; the probability of detection is calculated for a fixed probability of false alarm. The Neyman-Pearson criteria, illustrated in Figure 4-42, is used to make the data space more

The improvement seems to level off around 110° but peaks at 180°. This indicates that optimal fusion, with respect to the highlight region only, will occur for SAR images that are looking at opposite sides of a target. The results from the T-72 database coincide with those obtained using the canonical dataset.

Next, the optimal configuration for a 3 target system is investigated. The analysis focuses on determining which factors are most influential to the overall performance of the 3-look system - what is the optimal spacing of the radars? Are there specific looks that provide the best view of target specific features? The analysis framework is given in Figure 4-41. First, three SAR images with unique look angles are fused. Next, the ROC is generated according to the fused image. Then a Neyman-Pearson criteria is applied; the probability of detection is calculated for a fixed probability of false alarm. The Neyman-Pearson criteria, illustrated in Figure 4-42, is used to make the data space more

The improvement seems to level off around 110° but peaks at 180°. This indicates that optimal fusion, with respect to the highlight region only, will occur for SAR images that are looking at opposite sides of a target. The results from the T-72 database coincide with those obtained using the canonical dataset.

Next, the optimal configuration for a 3 target system is investigated. The analysis focuses on determining which factors are most influential to the overall performance of the 3-look system - what is the optimal spacing of the radars? Are there specific looks that provide the best view of target specific features? The analysis framework is given in Figure 4-41. First, three SAR images with unique look angles are fused. Next, the ROC is generated according to the fused image. Then a Neyman-Pearson criteria is applied; the probability of detection is calculated for a fixed probability of false alarm. The Neyman-Pearson criteria, illustrated in Figure 4-42, is used to make the data space more
manageable. Using this method, a single probability of detection value defines the performance for a set of 3 looks.

Figure 4-41: Analysis framework for a 3-look systems

The analysis framework inputs three SAR images at unique aspect angles. The data is then fused and the resulting fused image is analyzed. A single probability of detection output is reported as a performance measure of the particular combination.

Figure 4-42: Neyman-Pearson criteria

A Neyman-Pearson criterion fixes the probability of false alarm and reports the probability of detection. An alternative measure is the area under the ROC. Since the curves are well behaved, their Neyman-Pearson structure provides a computational advantage over the area measurement.

Since there are $27^3$ total possible 3-look combinations it is not feasible to do a brute force calculation. Also, a Monte Carlo simulation with uniform probability of selections will not produce the desired angular diversity. Instead the data was divided
into 15 angular bins. An image was selected randomly from each bin for all possible bin combinations. This process was then repeated over multiple iterations. This is illustrated in Figure 4-43. This approach guarantees randomly selected images, configuration diversity, and spacing diversity while minimizing computational complexity.

The images were then analyzed as a function of the mean intra-radar angle Eq. 4.45 and the intra-radar angle score Eq. 4.46.

\[
\overline{\Delta \theta} = \frac{1}{(N - 1)} \sum_{i=1}^{N-1} (\Delta \theta_i) \quad [4.45]
\]

\[
\vartheta = \frac{1}{N} \sqrt{\sum_{i=1}^{N} (\Delta \theta_i - \frac{360}{N})^2} \quad [4.46]
\]

Where \( N = 3 \). The mean intra-radar angle is simply the average of the angles between each of the three radars in the system. The intra-radar angle score is a measure of how
close the radars are to being equally spaced. For example, a mean intra-radar angle of 120° does not necessarily mean that the 3 radars are equally spaced. The intra-radar angle score is a measure of how close the system is to being equally spaced; a score of 0 means that the system is equally spaced. Figure 4-44 reports the probability of detection as a function of these two angular measures.

![Figure 4-44: Configuration dependent performance measures](image)

The peak mean performance for the mean intra-radar angle is 120°. This indicates that configurations with equally spaced radars are favorable. However, since non-equally spaced radars can have a mean-intra radar angle of 120° the performance as a function of intra-radar angle score is also important. From Figure 4-44 (b), the peak performance is for configurations with the lowest score. In the mean, equally spaced radars are optimal; although this is true in the mean, a variance is observed within the data. The high variance indicates that there are configurations that are not equally spaced that can also have high performance. Similarly, not all equally spaced radar combinations are equal.
Figure 4-45 and Figure 4-46 show two different configurations of 3 radars with equal spacing.

Looking at the ROCs of the individual images and the resulting fusion images illustrates that the quality of sets of equally spaced radars can vary. As seen in Figure 4-47, the quality of the 95° aspect image in the first set greatly improves the quality of that fusion image.
Not all sets of images with equal spacing have the same performance. There are also target specific features that influence performance. Since both configuration and object dependencies influence performance a new method is needed to determine an optimal configuration for the 3-look system.

A genetic algorithm (GA) was implemented to assess the performance of a 3-look configuration. Figure 4-48 outlines the various GA parameters that were set in the implementation; this is a very traditional implementation of a GA [52]. The probability
of detection output after fusion and the Neyman-Pearson criteria is the main evaluation metric used for individuals within the population.

---

**Figure 4-48: Outline of genetic algorithm implementation**

A traditional GA is used to analyze a 3-look radar system. Each individual within the population is a set of three aspect angles; this corresponds to a phenotype of a fused image. The fused image is then evaluated via an ROC and Neyman-Pearson criteria. Roulette wheel parent selection and 1-point crossover and swap are used for selection and mating. A replace all routine is used for generational advancement.

Under these parameters, the population evolved slowly and two distinct results were seen. The plots of Figure 4-49 illustrate the evolution of the population via histograms. At each generation the genotype of the population was histogramed; this corresponds to the number of individuals within the population that have that particular aspect angle. Initially, as seen in generation #28, the population evolves towards an evenly distributed radar configuration. However, as the population continues to evolve it converges to a more localized marginal configuration.
A comparison of the ROCs is made in Figure 4-50 for the optimal fusion image with 120° spacing, the sub-optimal fusion image with 120° spacing, and the fusion image obtained via the genetic algorithm. The fusion image calculated from the genetic algorithm has the best performance. There is, however, only a small gap in performance between it and the optimal 120° spacing set. The considerable drop in performance for the suboptimal 120° spacing dataset illustrates that there are significant tradeoffs between the influence of the radars’ configuration and target dependent features.

Figure 4-49: Evolution of the genetic algorithm

The histogram at each generation represents the number of individuals within the population that had a particular aspect angle. Two distinct solutions emerge. This illustrates the tradeoff between the influence of configuration and target dependent features.
The analysis above was concerned with systems using two and three looks. Operationally it may be possible to obtain more than this limited number of looks. Figure 4-51 shows the average ROC performance curves for two fusion rules as a function of the number of looks within the system.

Figure 4-50: Performance evaluation

The fusion as obtained by the GA results in the best image. Fusion with optimal 120° spacing is only slightly worse than the GA. There is a noticeable gap in the performance of the sub-optimal 120° spacing.

The analysis above was concerned with systems using two and three looks. Operationally it may be possible to obtain more than this limited number of looks.

Figure 4-51: Mean performance of multi-look systems

The ROC curves are reported for the two main fusion rules that were previously investigated, spatial averaging and probabilistic fusion. For both fusion rules, the addition of looks is a process of diminishing returns. After 5 looks there is little gained by adding another look. The probabilistic fusion saturates faster than the spatial average.
From the graphs, it is observed that the overall performance of the system increases for an increasing number of radars with diminishing returns. For up to 5 looks, the two fusion rules result in similar performance. After approximately 5 looks the probabilistic technique saturates quickly and the performance tops off. Diminishing returns are still obtained with spatial averaging, but small gains are made for over 5 looks.

4.4.4 MSTAR (advanced rules)

One of the ways that the above fusion rules can be enhanced is to compensate for the shadow regions. Since the shadow represents an area of target-self occlusion there is no useful highlight information that is gained in this region. The persistence weighted average approach is identical to the spatial average except in the shadow region. Within the shadow region, only pixels that are not within a shadow for the input images are fused. For example if a pixel is shadowed in 2 of 3 looks, the fusion value is equivalent to that of the third look.

Figure 4-52 shows an example where two input images are fused using the persistence rule. From the ROC, the persistent average has better information content then the regular spatial average. Also, a spatial average is seen to distort the relative magnitudes of the scattering centers where the input images have shadows. The persistent averages do not have this distortion; this improvement is seen only in the images themselves, not from the ROC.
The final technique used to analyze a small number of images is a component analysis. Figure 4-53 shows a set of 3 smoothed images used as a test set. A comparison of the decomposition via PCA and ICA is given in Figure 4-54. The test images were selected such that 2 inputs have redundant information (4° and 6°) and the third image contains more descriptive information.

In a persistent weighted average, fusion compensates for the shadow regions cause by target occlusions.

Figure 4-52: Persistence weighted average

The final technique used to analyze a small number of images is a component analysis. Figure 4-53 shows a set of 3 smoothed images used as a test set. A comparison of the decomposition via PCA and ICA is given in Figure 4-54. The test images were selected such that 2 inputs have redundant information (4° and 6°) and the third image contains more descriptive information.

Figure 4-53: Test input images

The first two input images represent images with redundant data since their aspects only differ by 2°. The third image offers a complimentary view.
The fused image from the decomposition corresponds to the dimension with the highest eigenvalue for both techniques; this corresponds to ‘Dim 1’ in Figure 4-54. The fused image and corresponding ROCs are given in Figure 4-55.

Figure 4-54: Component analysis

The dimensions of the component analysis correspond to the eigenvalues in decreasing order. For both techniques the third dimension is noise-like.
In this example, the dimensionality reduction using ICA outperforms that of PCA, and both dimensionality reduction techniques outperform spatial averaging. Since two of the images represent poor and similar information, the redundant, bad information overpowers the relevant information when spatial averaging is applied. ICA and PCA are better able to separate and use only a portion of that information. This scenario is a best case scenario for using the component techniques. The second test set, Figure 4-56, represents a 3-look system where each of the images has relevant and unique information.

Figure 4-55: Evaluation of fusion
The three test images were evaluated along with the three fusion image. Since the images are similar, images 1 and 2 have a similar quality metric. Image 3 contains the most information of the three inputs. The primary dimension of both the PCA and ICA contains more target information than the spatial average image. The area under the curve is reported in the legend.
For this example, the independent component associated solely with the largest eigenvalue is not necessarily a good estimate of the target. Useful, discriminatory target information is seen in the first two dimensions of the reduction, Figure 4-57.

Figure 4-56: Test input images

The dimensions of the component analysis correspond to the eigenvalues in decreasing order. The first and second dimensions contain useful target information.

Figure 4-57: Component analysis of ICA

When the information in the initial images is complimentary, the technique is not able to easily reduce the information into a single dimension. From an inspection of the data, input images 1 and 3 are seen to be slightly redundant, but input image 2 is complimentary of images 1 and 3. The secondary dimension in the ICA estimation is almost identically image 2. Therefore, dimensionality reduction in this problem needs to take the first two independent components into account. Dimensions 1 and 2 are
normalized and linearly combined to form a single fusion image for both PCA and ICA.

The resulting fusion images and ROCs are given in Figure 4-58

Figure 4-58: Evaluation of fusion

The three test images were evaluated along with the three fusion images. Since the quality of the 3 input images is high and represents complimentary data, all three fusion methods perform well. The spatial average outperforms either of the component techniques. The area under the curve is reported in the legend.

The performance of all 3 techniques appears to be equal. Again, regardless of the rule used, multi-look fusion increases the information content compared to the specific single-look images. Although the component techniques had comparable performance on the limited test sets, the difficulty is determining which dimensions, if any, to discard. Subsequently, if multiple dimensions are being kept, then fusion is not being achieved. A component analysis is better reserved for sets of data with large numbers of images. This concept is investigated in Chapter 5.
4.4.5 Point-Scatterer data experiments

For testing of the algorithms related to ISAR imaging, a radar image is simulated using the point-scattering model. This is a narrowband signal that uses a sinc point-spread function to simulate scattering centers. The amplitude of the sinc functions are varied according to the scattering strength. Additionally, target geometrics allow for occlusions of scattering centers for various look angles.

Initial tests were performed using 2 sets of two input images. The images were fused using a spatial averaging as well as single level wavelet decomposition. The purpose of the simulations was to compare the various images visually for quality and compare these observations to the evaluation criteria. Both images in test set A have a resolution of 2.39 units and are at 0° and 45° aspect. Test set B has mixed resolutions at varying aspects (Image 1: 1.21/ 2.42 units, 12° aspect / Image 2: 1.39/1.39 units, 45° aspect) and is a worst case scenario for fusion; the first image is far superior to the second image.

For test set A, both fusion methods produce results better than the initial input images, Figure 4-59. The output image from the wavelet fusion shows significant improvement over the spatial average. For the second test set, both fusion methods fail to create a new image that has better statistics than the initial inputs, Figure 4-60. As will be illustrated later, this is not a typical performance.
Fusion set A represents a typical set of images to be fused. The wavelet and spatial averaging methods both produce images better than the inputs.

Figure 4-59: Fusion of set A

Fusion set B is a worst case scenario for fusion algorithms; the disparity in information between the two images is severe. Both the spatial average and wavelet fusion rules result in an image that has higher power in the noise pixels than one of the original images.

Figure 4-60: Fusion of set B
Although the fusion was not successful in case B, the performance metric accurately depicts both the impulse intensity and the sidelobe levels. The target power scale space ratio (~SNR) is illustrated as a valid performance metric and will be used in subsequent analysis.

Next, wavelet fusion was evaluated as a function of look angle. To test the algorithms’ performance over a range of operating characteristics, test set A was again used. In this experiment, however, image 2 was varied over a range of look angles from 0° to 180°. The same experiment was also performed on test set B. Figure 4-61 and Figure 4-62 depict the results from these experiments. The graphs also show the improvement of fusion over the average of the input images.

![Figure 4-61: Variable aspect for fusion set A](image)

In this simulation image 1 is fixed, and the aspect angle of image 2 is varied. The wavelet fusion image method is superior to the individual images as well as spatial averaging.
For test set A it is clear that wavelet fusion is significantly better than spatial averaging for most aspect angles. Fusion is the most effective for images where the sidelobes do not overlap – 45° increments. In test set B it is observed that wavelet fusion is not always better than an average of the initial images, but is always superior to the worse of the two images. This is the case since the gap in performance between the two input images is large.

In order to more thoroughly compare wavelets to spatial averaging, 1000 trials were run for input images with random look angles, random range resolution, and random cross-range resolution. The point configuration was static for the simulation. In this test, the wavelet image was better than spatial averaging in 89.5% of the test cases.

Next, the point configuration was also varied. The number of points used was selected uniformly between 1 and 6. The locations of these points were uniformly varied over the entire image. Finally, the look angle, range resolution, and cross range

---

Figure 4-62: Variable aspect for fusion set B

In this simulation image 1 is fixed, and the aspect angle of image 2 is varied. The wavelet fusion image method is superior to the worse of the two individual images. The wavelet image is no better than that of spatial averaging.
resolutions were also uniquely varied. In 81% of the test cases, wavelets produced a better result than spatial averaging. Of the 1000 trials only 4 resulted in the fusion performance being less than the average performance of the initial input images. Figure 4-63 illustrates the improvements for each of the trials. The 81% improvement can be increased to 89% if a multilevel wavelet fusion is used.

Figure 4-63: Improvement for random point configurations

(a) Sorted according to wavelets  
(b) Sorted according to average

The improvement in the approximate SNR is reported for the wavelet and the average cases. In 81% of the trials the wavelet fusion resulted in a better image.

In order to more accurately model a real ISAR scenario, target occlusion was taken into consideration. In test image 1, Figure 4-64, the top-right scattering center is being occluded. The max, min, and mean functions are then applied to the approximation and detail images in the wavelet decomposition. Figure 4-65 shows all possible combinations of the rules being applied.
As predicted in the theory, the MIN rule should be used in the areas where there are no target features, and the MAX rule should be used in areas where the scattering

Figure 4-64: Single test image with occlusion point

The test images represent an ISAR image in which one of the scattering centers is being occluded.

Figure 4-65: Wavelet rules applied to test image with minimal occlusions

In the figure the various wavelet mapping rules are illustrated for a dataset where a single point is occluded in one of the images. When the min rule is applied to the approximation the occluded point is not imaged. However when the max rule is applied there is high power in the sidelobes.

As predicted in the theory, the MIN rule should be used in the areas where there are no target features, and the MAX rule should be used in areas where the scattering
centers preside. Therefore, a mask needs to be generated based on the disparity between the images. The disparity image is thresholded based on the blending parameter, $\beta$. The threshold is set based on the desired amount of redundant and complimentary information. Finally, morphological operations are used to process the thresholded difference mask to remove / reduce the effects of noise pixels. The flow is illustrated in Figure 4-66.

![Figure 4-66: Generating wavelet map](image)

The three images represent the process of generating the blending mask for a specified $\beta$. First, the disparity image is thresholded. Next, the noise pixels are reduced via morphological operations. The final output represents the fusion map used with the wavelet coefficients. The red represents where the max rule is used and the white represents where the min rule is used.

A MIN function is used as the default mapping function for the fusion of the wavelet coefficients. A MAX function is applied to the approximation image as based on the mapping process described in Figure 4-66 (c). Figure 4-67 shows the fusion results as a function of the blending parameter, $\beta$. 

---

**Figure 4-66: Generating wavelet map**

The three images represent the process of generating the blending mask for a specified $\beta$. First, the disparity image is thresholded. Next, the noise pixels are reduced via morphological operations. The final output represents the fusion map used with the wavelet coefficients. The red represents where the max rule is used and the white represents where the min rule is used.

A MIN function is used as the default mapping function for the fusion of the wavelet coefficients. A MAX function is applied to the approximation image as based on the mapping process described in Figure 4-66 (c). Figure 4-67 shows the fusion results as a function of the blending parameter, $\beta$. 

---
A comparison of Figure 4-67 and Figure 4-65 illustrates the effectiveness of the morphological blending mask. For low values of $\beta$, the image has more of the MAX properties. For high values of $\beta$, the image has more of the MIN properties. For $\beta$ of .5, the blending allows for the power of the scattering center to be imaged while simultaneously reducing the sidelobe power. Alternatively, correlation, variance, absolute value, or entropy measures could be implemented to replace the disparity measurement.

A second, more complicated, test set was generated to further test the mapping procedure. The initial images are given in Figure 4-68. The min/min, max/max, mean/mean, mapping/min images are given in Figure 4-69. Again the morphological blending fusion method is seen to be far superior to the traditional wavelet fusion techniques. The power of the scattering centers is maximized while the noise is suppressed.
Figure 4-68: Test images with multiple occlusion points

The test images represent an ISAR image in which multiple scattering centers are being occluded. The individual point scatterers’ properties are also shown.

Figure 4-69: Morphological wavelet fusion mapping

The image shows the output of fusion mapping for four different combinatorial rules. The min/min rule is unable to image all of the relevant scattering centers. The max/max has high noise components. The spatial average is the worst of both the min and max rules, whereas the morphologic wavelet fusion map for an optimal $\beta$ value is the best of both rules. The power at the scattering centers is maximized while the noise is minimized.
4.5 Conclusions

The use of data fusion in multi-look scenarios allows for a single image to be viewed instead of multiple individual chips. Ideally, the fused image will contain all of the target information that is contained within the individual chips. Specifically, this chapter analyzed data fusion as it relates to the highlight information within the images. The highlight information contains the reflectivity and geometry of the scattering centers that are unique to specific targets.

First a canonical dataset was examined using data generated from the ray-tracing model. An analysis of the pixels’ intensity distributions revealed that sub-divisions within the image can be advantageous in processing, but the distributions begin to break down as the number of sub-divisions increases. Probabilistic and thresholding methods were analyzed for their fusion capabilities. It was observed that when the initial images each have large amounts of information in a low noise environment, there was little to gain using fusion. Optimal fusion occurs in cases with manageable noise where the images contain complimentary information. In a 2-look SAR system, the optimal fusion occurred when the radars where separated by 180°.

Next, the T-72 target from the MSTAR database at 15° depression was examined. Again the spatial average fusion technique seemed to behave the same as the LRT probabilistic technique. A further analysis of Bayesian risk criteria revealed that the use of the pixel distributions and a non-linear decision boundary offered little gains in a practical implementation over the simpler linear decision boundary. Moreover, a
weighted linear decision boundary was only useful in cases where there was a large
disparity in the quality of the input images.

To better understand the gains obtained using fusion, the performance was tested
for various radar configurations in 2 and 3 look systems. Like the canonical database, the
optimal configuration of a 2-look system was radars at 180°. Although the peak occurred
at 180°, the largest improvements came from simply ensuring that the radars were spaced
apart by at least 110°. Similarly, the optimal mean configuration for a 3-look system was
equally spaced radars. There were, however, significant differences in the quality of the
fused images from multiple sets of radars that were equally spaced; in addition to the
configuration, the location of specific target features affects the performance. Using a
genetic algorithm an ideal set of 3 looks was obtained for the T-72. Fusion of additional
radars had diminishing returns, and offered very little performance gains after 5 looks.

One way in which the fusion of SAR images can be improved is to account for the
target shadows within the image. A simple weighted average, as a function of the
shadow location, applied to the fusion technique can be extremely advantageous. The
aspect dependent nature of the imagery makes is difficult to account for the variations
from one image to another. Two component analysis techniques, PCA and ICA, were
also implemented in an attempt to combat these variations. Unfortunately, the use of
these component techniques was limited since the number of input images is low. Since
the variations in the input images are high, the component techniques could not
significantly reduce the data’s dimensions. These techniques are more suited for cases
with large numbers of images.
Finally, the use of wavelets was investigated for simulated ISAR imagery. The wavelet mapping technique was seen to be superior to a traditional spatial average for images without occlusion. However, traditional mapping functions for wavelet fusion fail when occlusions occur. Target occlusion is common in ISAR imagery and therefore needs to be taken into consideration. Using a method that compares the intensities of the input images, a mapping function was developed that simultaneously maximized the power at all scattering centers while minimizing the power in the remaining noise pixels. The advantages of the mapping function were illustrated for two sets of images with simulated occlusion.
Chapter 5
Persistence Modeling

5.1 Problem Overview

The main goal of an NCTR system is to identify or classify an unknown target. There has been a considerable effort to develop such systems that utilize SAR and ISAR images. Traditionally, these algorithms operate as a function of the aspect angle since radar images will change drastically as a function of the target / radar configuration. Figure 5-1 illustrates a system of multiple looks of a single target as a function of aspect.

![Figure 5-1: Typical phenomenology for a target/radar configuration](image)

The image shows a common geometric interpretation of multiple looks of a stationary target in a SAR system. Each look contains pose information that is dependent on the aspect angle.

The radar images vary on both small and large angular scales. These changes can be understood by examining a single pixel within a set of angularly varying, registered multi-look images. On the small scale, there will be noticeable perturbations of the pixel value for changes as small as 1°. These variations are a function of the radar noise, error
in alignment, and minor geometric changes in the properties of a particular target feature.
The geometric changes in the scattering centers are seen more clearly on the large angular scales. First, there is target self-occlusion; target self-occlusion occurs because the target-radar geometry precludes the radar from being able to see all sides of the target for a single aspect. Therefore, under a certain set of aspect angles the pixel value will be low because it is part of the target shadow. Even when the feature is not in the shadow, the geometry of the feature changes for varying looks and therefore the scattering properties also vary. Figure 5-2 is an illustration of these changes as a function of multiple regions within the MSTAR T-72 data at 15° depression.

Figure 5-2: Illustration of small and large scale angular perturbations
The figure illustrates the variation in intensity of three different pixels within a set of T-72 images. Both small scale changes (noise-like variations) and large scale changes (average intensity) are illustrated in all three of the image regions. The locations of the points of interest (a) are overlaid on top of the mean image for all looks.

Although ordering of a set of multi-look images as a function of the aspect angle is intuitively appealing, there are some limitations to this approach. Instead, researchers
are looking for methods to generate either a single image or set of images that have target specific features [53] [54]. In this chapter, a persistence model is proposed that orders the images in a way such that the focus is on the scattering centers and not the angular dependencies. There are two main areas that the persistence model is compared to the traditional angular image set – visualization and classification.

Visualization is the first problem that is examined. For large sets of images, such as a template database, there is no direct way to visualize all of the data. Often times the data is viewed as a movie where the aspect is varied over time. The drawback to this approach is that symmetric target features are not seen at the same time. For example, the left and right sides of a tank cannot be viewed simultaneously. Using the persistence model, the data is varied as a function of the properties and thus features on both sides of a target are viewed simultaneously. Also, the variations within the traditional dataset are a function of the look angle whereas the persistence model varies as a function of the target. Section 5.3.1 outlines some of the visualization advantages of the persistence model.

Secondly, the persistence model can be used for classification. The persistence model is compared to the traditional model in terms of a template-matching classification structure. The advantage of a persistence model is in scenarios where only a small number of templates per target can be used. Since the persistence structure is a function of target features not the look angle, it offers a more intuitive approach for the problem of template reduction. Section 5.3.2 outlines some of the traditional methods that were implemented for classification and compares them to the performance on the persistence model.
5.2 Implementation

The general phenomenology of a SAR target/radar configuration is displayed in Figure 5-1. For each pass of a radar a single look generates a single image. Multiple looks build up the dataset to obtain information for different target poses. The set of $L$ total looks is defined by the vector $\Theta$, Eq. 5.1.

$$\Theta = \{\ell_0, \ell_{\Delta^r}, \ldots, \ell_{(L-1)\Delta^r}\}$$  \[5.1\]

Next, each look is defined by the radar image. Figure 5-3 shows the pixel representation of the radar image; however, instead of viewing each radar image as a 2-D image, each image is represented as a 1-d vector according to the appropriate indexed values. This representation causes no lack of generality. This is a common reshaping method. A typical target image from a random look, $\ell_n$, is described according to Eq. 5.2.

---

Figure 5-3: Radar image

The figure shows a radar image where each pixel is indexed by the relative $x$-$y$ spatial coordinate. The representation uses a single index derived from the subscript index. The columns are numbered from left-to-right followed by the rows from top-to-bottom.
Where \( x_i \) represents the pixel intensity of the \( i^{th} \) indexed pixel.

The persistence model is implemented under the following two constraints:

**#1:** All of the images defined by the look vector \( \Theta \) are pre-registered in the identical image space defined by \( I_{\ell_x} \).

**#2:** Information content and processing are constrained to single-pixel operations. Neighborhoods of pixels are ignored.

If the target chips are taken from a larger SAR image (as is typically the case in a battlefield intelligence scenario) then the geo-spatial co-registration problem has already been solved. Target detection and localization algorithms would output image chips according to the claims made in constraint #1. Centering is not required, nor is registration to a static known pose; co-registration as described in Chapter 3 is sufficient. A common pixel between uncommon images will be representative of the same target feature.

Constraint #2 will limit the ability to use certain types of algorithms in subsequent processing states, but is a good initial assumption. Many SAR algorithms do not take into account the local pixel neighborhoods. If pixel neighborhood information was desired, blurring could be used as a pre-processing step. Pre-blurring would allow for neighborhood operations, and the constraint would not be broken.

Since a neighborhood structure is not in use, spatially ‘close’ pixels, \( \{ \ldots x_{n-i}, x_n, x_{n+i}, \ldots \} \) are no longer constrained to lie next to each other in the image vector. The image vector can be reordered in an arbitrary way, Eq. 5.3.
The mapping of the initial pixel space to the reordered pixel space $(I_{\ell_s} \leftrightarrow \tilde{I}_{\ell_s})$ is useful to group like pixels. This mapping can be defined in terms of the pixel intensities, the spatial location, or any other desired function. It is crucial that the mapping is invertible. When the initial mapping is performed the index pointers must be saved such that an image can be reformed. The initial ordering allows the ‘image’ to be displayed. After processing it is therefore important to revert back to this index to properly display the output image.

Similarly, the look vector can also be reordered. Breaking this ordering however, carries a more substantial significance. The ordering, $\Theta = \{\ell_{0^*}, \ell_{\Delta^*}, ..., \ell_{(L-1)\Delta^*}\}$ allows certain aspect dependent features to be modeled. Many typical algorithms train models based on these dependencies to capture the targets’ statistics.

The goal is to reorder the pixels according to a new paradigm – persistence. The pose information according to aspect angle is partially redundant since it is also encoded in the registration process. The projection and reordering of the look vector can allow for the development of a robust and reliable target persistence model. Eq. 5.4 highlights the process for this mapping function.

$$\Theta = \{\ell_{0^*}, \ell_{\Delta^*}, ..., \ell_{(L-1)\Delta^*}\} \Rightarrow \Theta' = \{\ell_{n\Delta^*}, ..., \ell_{m\Delta^*}\}$$ \hspace{1cm} [5.4]

Previously, the data has been defined as either an image in the single-look space or a point in the multi-look space. In a full dataset however, the data will be a multi-
dimensional vector in a multi-dimensional look space. The data as viewed from its
geospatial aspect dependent representation is given by Eq. 5.5.

\[
D = \begin{bmatrix}
I_{\ell,\psi'} \\
I_{\ell,\alpha'} \\
\vdots \\
I_{\ell,(L-1)\alpha'}
\end{bmatrix} = \begin{bmatrix}
X_1^{\ell,\psi'}, X_2^{\ell,\psi'}, \ldots, X_{MN}^{\ell,\psi'} \\
X_1^{\ell,\alpha'}, X_2^{\ell,\alpha'}, \ldots, X_{MN}^{\ell,\alpha'} \\
\vdots \\
X_1^{(L-1)\ell,\alpha'}, X_2^{(L-1)\ell,\alpha'}, \ldots, X_{MN}^{(L-1)\ell,\alpha'}
\end{bmatrix} 
\] [5.5]

As stated above, the angle index of the pixel is unimportant. The mapping in Eq.
5.4 can be used, \( \Theta \Rightarrow \Theta' \). The data from Eq. 5.5 now becomes Eq. 5.6:

\[
D' = \begin{bmatrix}
I_{\ell,\alpha'} \\
\vdots \\
I_{\ell,\alpha'}
\end{bmatrix} = \begin{bmatrix}
X_1^{\ell,\alpha'}, X_2^{\ell,\alpha'}, \ldots, X_{MN}^{\ell,\alpha'} \\
\vdots \\
X_1^{(L-1)\ell,\alpha'}, X_2^{(L-1)\ell,\alpha'}, \ldots, X_{MN}^{(L-1)\ell,\alpha'}
\end{bmatrix} 
\] [5.6]

Because the location of the pixel is unimportant the pixel ordering can also be
broken. The mapping from Eq. 5.3 can be used, \( I_{\ell} \leftrightarrow \tilde{I}_{\ell} \). The data from Eq. 5.6 then
becomes Eq. 5.7:

\[
D'' = \begin{bmatrix}
\tilde{I}_{\ell,\alpha'} \\
\vdots \\
\tilde{I}_{\ell,\alpha'}
\end{bmatrix} = \begin{bmatrix}
x_1^{\ell,\alpha'}, \ldots, x_{MN}^{\ell,\alpha'} \\
\vdots \\
x_1^{(L-1)\ell,\alpha'}, \ldots, x_{MN}^{(L-1)\ell,\alpha'}
\end{bmatrix} 
\] [5.7]

The data of \( D'' \) can then be modeled using the most appropriate method
(deterministic, stochastic, etc...).

The generic model for generating a set of persistence images is given in Figure 5-
4 below. As an input, the model takes in a set of SAR images over a range of aspects.
Each image is represented as a vector; those vectors are ordered as a function of the look
angle. The angular constraints within that data are first broken followed by the pixel
constraints. A set of persistence images can then be output directly from the mapping
functions. However, a more sophisticated approach takes this data and applies a generative modeling strategy. Using the parameters intrinsic to that model, a set of estimated persistence images can also be obtained.

$D = \begin{bmatrix}
I_{1,1} & I_{1,2} & \cdots & I_{1,n} \\
I_{2,1} & I_{2,2} & \cdots & I_{2,n} \\
\vdots & \vdots & \ddots & \vdots \\
I_{m,1} & I_{m,2} & \cdots & I_{m,n}
\end{bmatrix}
\begin{bmatrix}
x_{1,1} & x_{1,2} & \cdots & x_{1,n} \\
x_{2,1} & x_{2,2} & \cdots & x_{2,n} \\
\vdots & \vdots & \ddots & \vdots \\
x_{m,1} & x_{m,2} & \cdots & x_{m,n}
\end{bmatrix}
$

$D' = \begin{bmatrix}
\bar{I}_{1,1} & \bar{I}_{1,2} & \cdots & \bar{I}_{1,n} \\
\bar{I}_{2,1} & \bar{I}_{2,2} & \cdots & \bar{I}_{2,n} \\
\vdots & \vdots & \ddots & \vdots \\
\bar{I}_{m,1} & \bar{I}_{m,2} & \cdots & \bar{I}_{m,n}
\end{bmatrix}
\begin{bmatrix}
x_{1,1} & x_{1,2} & \cdots & x_{1,n} \\
x_{2,1} & x_{2,2} & \cdots & x_{2,n} \\
\vdots & \vdots & \ddots & \vdots \\
x_{m,1} & x_{m,2} & \cdots & x_{m,n}
\end{bmatrix}
$

Figure 5-4: Persistence modeling

The goal of the persistence model is to take a set of SAR image inputs that have aspect angle dependencies and output a new set of images whose features are dependent on the physical features of the target rather than the look-angle dependencies. The flowchart above outlines the proposed approach to achieve that goal.

As seen in the figure above, the first step in the algorithm is to break the angular constraints. The persistence model tries to represent the intensity features of a particular scattering center. Therefore, a logical method in which to break the angular constraints of a particular pixel is to reorder the data based on the intensity. The $L$ values at each pixel are sorted according to the given intensity. Next, the pixel constraints must be broken. The likeness can again be simply defined in terms of the intensity. As an initial implementation, the mapping sorts the pixels based on the mean intensity.

Finally, the data was modeled using a deterministic approach. The ansatz data model is defined to be a cubic polynomial. The parameters are calculated using a least squares fit. The range is determined by the total number of input images, but can be
normalized in range of \([0 \ 1]\). Figure 5-5 outlines this initial implementation of the persistence model. The remainder of the chapter details the realization and testing of this model on the MSTAR SAR data.

\[
\text{Figure 5-5: Initial implementation of persistence model}
\]

The initial persistence model is implemented using sorting techniques based on intensity values as well as a deterministic modeling approach.

\section*{5.3 Results & Analysis}

\subsection*{5.3.1 Visualization}

The images in Figure 5-6 show the initial and processed datasets for the D7 MSTAR target. The data is mapped according to the process outlined in Figure 5-5.

\[
\text{Figure 5-6: Full data and mappings for D7}
\]

In the images above, the initial aspect dependent data is subjected to two sets of reordering to obtain the persistence data set. The data is taken at 15° depression.
In the initial data, the aspect dependencies are clearly seen. The shadow region, for example, is seen to snake through the pixels. After the initial reordering it is clear in which pixel locations the scattering centers are the strongest. The final reordering shows the grouping of like pixels.

All of the data in 5-6(c) is plotted as a function of the intensity. The result is shown in Figure 5-7. The graph shows that all of the data follow a general cubic trend.

A least squares approach was used to estimate the coefficients of a cubic polynomial for each pixel location. The input of the polynomial, $x$, is the intensity ranking according to the intensity mapping step. The cubic is fit and unique parameters are obtained for each pixel location. Figure 5-8 illustrates the 4 parameters.
Once the data is modeled according to the cubic polynomial, then a generative dataset can also be obtained. A comparison between this dataset and the input data reveals a good overall fit of the model.

Figure 5-9 shows that the generated model matches the input data quite closely. The overall absolute error is within an acceptable margin. Figure 5-10 shows the configuration of said error as a function of the target image. Notice that the maximum error occurs at clutter locations. This occurs because the cubic interpretation assumes that there are low and high regions within the data. If the pixel contains no high intensity information, the data deviates from the cubic assumption. There are also medium levels of error within the target region. This is also expected since it is the most complicated data to model and most vulnerable to noise.

Figure 5-8: Coefficients for cubic persistence model
The image illustrates the coefficient fit to each pixel location for a cubic polynomial. The coefficients were calculated using a least squares fit for all of the available data.
After the model parameters are obtained, the set of persistence images can then be generated. The $\alpha$ parameter defines the persistence from 0 to 1. The input to the cubic polynomial, $x$, is normalized based on the number of images in the initial dataset to obtain $\alpha$.

Figure 5-9: Modeled persistence data and corresponding error

The cubic polynomial fit (a) seems to estimate the data well. The absolute error of the persistence model is reported in (b). The maximum errors are at the tails of the model.

Figure 5-10: Maximum modeling errors

The maximum absolute errors for each pixel are displayed as an image. The most severe errors are seen to occur in the clutter region with some smaller errors within the target region.
Figure 5-11 shows the D7 target for 5 values of persistence. At $\alpha=0$, the overall shape of the target is identified. The cab top and motor hutch are clearly visible for low values of $\alpha$ because these target features are visible for a large range of aspects. As the value of $\alpha$ increases towards 1 the aspect dependent features emerge. For example, notice at $\alpha=1$, the plow and plow connections are easily discernable. Such features are obfuscated and spread over many images in the initial dataset; all of the features cannot be seen in a single image. In the set of persistence images, however, these connectors and features are easily seen.

![Persistence model images of D7](image)

Figure 5-11: Persistence model images of D7

The images above show a set of persistent data for varying levels of $\alpha$. Note that the features common in all aspects are seen for low values of $\alpha$, but angular dependent features emerge when $\alpha$ approaches 1.

The same tests were performed but for the T-72 tank chips. Again the persistence technique outputs a set of intuitive images. These images are seen in Figure 5-12. For the tank, one of the features that is visible from all aspects is the turret hatch on top of the tank. For $\alpha=0$ this is the main feature that is displayed. The sloped edges covering the treads are revealed as $\alpha$ increases. For $\alpha=1$ the features at the base of the turret are revealed. Notice that the tip of the gun barrel is visible from most aspects and displayed for $\alpha=0$. But the base of the gun is only visible in a small range and thus brightest for high $\alpha$ values.
The final visualization analysis that was performed is the use of a Principle Component Analysis (PCA) to reduce the dimensionality of the data space. The PCA data reduction is applied to the angularly dependent data as well as the persistence data. The main goal of this analysis is to determine the way in which each of the datasets varies; a set of basis images is obtained.

First, an eigenanalysis is performed to determine the number of images that need to be kept in order to adequately represent the variability of the data. Figure 5-13 shows that only 4 persistence images are needed to adequately capture the target features compared to approximately 20 of the angularly dependent data.

Figure 5-12: Persistence model images of T-72

The modeled set of persistence images for the T-72 is displayed for varying values of $\alpha$. The features within the image correspond closely with the tank’s unique features.
The four primary dimensions as well as the residuals are shown for the angular dependent data, Figure 5-14, and the persistent data, Figure 5-15. The residual image is defined according to all data not in the initial 4 dimensions. These figures clearly illustrate the difference in the decomposition of the angular images as compared to the persistence images. The 2nd thru 4th images in the angularly diverse data set show how the target points vary – top-to-bottom, left-to-right, and corner-to-corner. This variation corresponds to the look angle dependency of the initial data. Also the residuals are seen to be high. High residuals indicated that the 4 images are not sufficient to represent the entire dataset. This observation is also supported by the eigenanalysis.

Figure 5-13: Eigenvalues for T-72 datasets

The plots represent eigenvalues for the angularly dependent data and the persistence data respectively. The information is spread over more dimensions in the angular data.
In contrast to the figures above, the residuals from the decomposition of the persistence models are seen to be low. The data is concentrated in the primary eigenvectors. This means that data reduction is easier to achieve without loss of information. Also the 2\textsuperscript{nd} thru 4\textsuperscript{th} images represent the features of the targets. The top of the turret and sloped tread covers are seen as primary features in the T-72; the plow and connectors are seen as the primary variation in the D-7 images. This supports the use of the persistence space over the angular space since the image variations are a function of the target features, not the restrained look angle.

Figure 5.14: Dimensionality of the angularly dependent data

The images show the primary 4 dimensions and corresponding residuals for the angularly dependent data for each of two targets. The main variations within the data correspond to the look angle – top-to-bottom, left-to-right, corner-to-corner. Significant structures are seen in the residual image.
The features described above are further illustrated in Figure 5-16 below. The primary image is shown as an intensity image. The secondary image contours are overlaid with thick lines and the tertiary contours are illustrated with thin lines.
In Figure 5-16 (a) the various features of the tank can be seen. For example the tertiary data illustrates the base of the tank including the sloped tread coverings. The hatch is outlined in the secondary dimension. Notice that the wheels can be seen in the contour lines. This is a function of performing 2D algorithms to 3D object. Figure 5-16 (b) clearly illustrates the features of the bulldozer. The main structure is seen in the image. The secondary features such as the bumps on the hood and plow connections are easily discernable.

Using either the $\alpha$ interpretation or the PCA reduced data-space, the persistence modeling approach is seen to provide an intuitive method by which the target features are better visualized over the traditional angular dependent data-space. Although the techniques are compared using a visual analysis, it is not difficult to extrapolate and see
the potential use of such models in automated systems. The subsequent section tests classification using a reduced template set as one such example.

5.3.2 Classification

The following simulations and tests were used to assess the persistence model in terms of a classification framework. To facilitate the analysis, basic template matching types of classification structures were used. Although the classification structures are basic, they have been shown to be quite effective for the classification of SAR images [33].

Table 5-1 reports the settings that were fixed for these simulations. A reduction in image size was used to lower the overall classification rate of the default case. Small images are more difficult to classify and thus offer more of a challenge to the classifiers. This change therefore allows for a more though analysis. Figure 5-17 illustrates the set of 5 target templates that are being used in the experiments. The persistence model represents the data after restructuring only; deterministic modeling was not performed. As discussed previously, thresholding is an important step in reducing the effects of clutter; the classification algorithms should work on the scattering centers, not the background clutter. Figure 5-18 illustrates a thresholded version of the templates of Figure 5-17.
Table 5-1: Global simulation parameters

<table>
<thead>
<tr>
<th>Image Size</th>
<th>Image Vector</th>
<th>Template Data</th>
<th>Testing Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reduced from 64x64 to 16x16</td>
<td>203</td>
<td>15° depression</td>
<td>17° depression</td>
</tr>
</tbody>
</table>

Figure 5-17: Templates

The figures show the two different sets of length 203 template vectors used for the 5-target classification problem.
The application of the threshold is important to ensure that the data being classified is within the target region. It is however useful to look at both cases. The non-thresholded case is a good analysis of how well the algorithm can classify based on clutter and target regions. Although this does not show the practical applicability of the algorithms, it shows the overall classification capabilities.

In the simulations of Figure 5-19, the number of templates was varied. A nearest-neighbor template matching scheme was used for classification. The performance was reported as a function of the percentage of the total number of templates. There was no method implemented to intelligently select which images were available; it was randomly determined which templates to discard. The classification rate of the aspect dependent data decreases as the number of templates is reduced. The classification rate of the persistence model is relatively unaffected but the number of templates used.

Figure 5-18: Thresholded templates
The figures show the two different sets of length 203 template vectors used for the 5-target classification problem. A threshold is applied to eliminate the capability of the classifier to classify based on the background clutter.
Although the classification rate is initially lower using the persistence models, the number of templates can be easily reduced without sacrificing performance. When using the aspect dependent images, all of the aspects need to be present in the template set to optimally classify the target. There are many scenarios in which the use of such a populated template dataset is not practical.

Next the performance was tested as a function of the number of nearest neighbors. All available templates were used. The results are reported in Figure 5-20. In the aspect dependent case, only a small number of images (centered on that aspect) accurately match the test image. Therefore as the number of neighbors used to classify each test image is increased the overall performance drops; simply, there are only a few template images that match a single test image. For the persistence model, however, multiple template images match each test image. The templates all contain information from an

Figure 5-19: Classification performance as a function of the number of templates

A template matching scheme is used to classify a test instance to its closest match within the available template set. The classification rate is reported as the number of templates available per target was reduced. There is a clear drop in performance of the aspect dependent data, but the persistence data is unaffected. Multiple thresholds are used to illustrate the effects of eliminating the clutter.
assortment of aspect angles. Still, the overall classification rate with the persistence model is lower than that with the aspect dependent data in the presence of full templates.

For the remaining simulations only the thresholded case is being examined (30dB). Moreover a minimum distance classification metric is used. The test image \( t(i, j) \) is defined in vector form by \( \tilde{t}(k) \). The template \( T_n(i, j) \) is defined in vector form by \( \tilde{T}_n(k) \). The error for test image \( t(i, j) \) for the template corresponding to target \( n \) is given by Eq. 5.8.

\[
e_{T_n} = \sum_{k=1}^{203} (\tilde{t}(k) - \tilde{T}_n(k))^2
\]

Classification was performed according to Eq. 5.9

Choose \( n \) such that \( e_{T_n} = \min_{i \in [1, 5]} e_{T_i} \)
The threshold was applied twice – once to the templates and once to the test image. The first threshold is applied to the template directly, Eq. 5.10.

$$T_n^i(k) = \overline{T}_n(k) > \tau_i$$  \[5.10\]

The second threshold is applied to the error summation. Through the use of this threshold, the sum of squared errors is only calculated in areas of the image where the intensities are sufficiently high, Eq. 5.11.

$$e_{\tau_2}^* = \sum_{k=1}^{203} \delta(k)(\bar{t}(k) - T_n^i(k))^2$$  \[5.11\]

Where

$$\delta(k) = \begin{cases} 1 & \text{for } \bar{t}(k) > \tau_2 \\ 0 & \text{else} \end{cases}$$  \[5.12\]

For the case when target $T_n$ is represented by $M$ templates, the error is calculated as Eq. 5.13.

$$e_{\tau_2}^* = \sum_{i=1}^{M} \sum_{k=1}^{203} \delta(k)(\bar{t}(k) - T_n^i(k))^2$$  \[5.13\]

Figure 5-21 reports the training classification performance on each of the 5 targets in the system for a range of templates. The minimum error distance classifier is used; for the case of a single template this is equivalent to a single nearest neighbor classifier. For the angular data, the templates are spaced at approximately every 2° in aspect. For the persistence data, the template indexes are arranged from low-to-high $\alpha$. 
Clearly the individual aspect-dependent images are not very reliable for classification when only a single image is used. The angularly dependent images exhibited very random performance and there was no motivation to select one set of templates over another. This, however, is not the case for the persistence models. The persistence templates exhibit more desirable performance traits. Each target has a certain area within the persistence space that it is optimally classified. These marks are represented by black dots in the figure above.

Using the training performances as a guide, four sets of templates were manually defined. The number of templates used per target in these sets is 1, 2, 3, and 4 respectively. The index values are listed below. The performance on the test data set is reported in Table 5-2.

Set 1: \{111\}
Set 2: \{80, 150\}
Set 3: \{65, 121, 154\}

![Figure 5-21: Average classification rate on the training data using a single template](image)

For each template, the entire training dataset was classified based on that single template. From the plots it is clear that there exists no single template index in the angular data that is capable of classifying over the entire database. For the persistence model, however, there are multiple template indexes (α’s) that provide robust classification across the entire dataset.
Set 4: \{71, 103, 121, 168\}

Table 5-2: Classification using minimum distance classifier with persistence templates

<table>
<thead>
<tr>
<th>Set</th>
<th>T1</th>
<th>T2</th>
<th>T3</th>
<th>T4</th>
<th>T5</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>.8477</td>
<td>.1040</td>
<td>.9498</td>
<td>.3144</td>
<td>.9030</td>
<td>.6175</td>
</tr>
<tr>
<td>2</td>
<td>.8125</td>
<td>.5369</td>
<td>.7759</td>
<td>.6622</td>
<td>.9632</td>
<td>.7484</td>
</tr>
<tr>
<td>3</td>
<td>.7734</td>
<td>.5503</td>
<td>.7893</td>
<td>.6823</td>
<td>.9900</td>
<td>.7567</td>
</tr>
<tr>
<td>4</td>
<td>.6914</td>
<td>.6510</td>
<td>.7926</td>
<td>.7224</td>
<td>.9498</td>
<td>.7636</td>
</tr>
</tbody>
</table>

*The average performance takes into account a variable number of test instances per target.

The test performance closely matches the training performances described in Figure 5-21 (b). Additionally, set 4 shows that using only 4 templates and a minimum distance classifier the overall classification performance is quite high. The performance is equivalent to approximately 40 aspect dependent templates on a nearest neighbor classifier.

To get equivalent performance from the angular dependent images for such low numbers of templates a more complicated classification structure is needed. Since there is no motivation to choose one aspect-dependent template over another, a traditional dimensionality reduction technique is used to obtain a small number of templates. Using PCA, the basis images can be used as a classification structure as outlined in Figure 5-22.
Using this technique, the image $\tilde{t}(k)$ is estimated from the set of $M$ template basis images, $\tilde{T}^i_n(k)$ $i = 1, \ldots, M$ by Eq. 5.14. The pseudo-code for classification is also listed.

$$\tilde{t}(k) = \sum_{i=1}^{M} \alpha_i \times \tilde{T}^i_n(k) \tag{5.14}$$

For: All sets of target templates
- Using a least squares fit, estimate $\alpha_i$ from $\tilde{T}^i_n(k)$ and $\tilde{t}(k)$
- Generate an estimate, $\tilde{t}(k)$, using the $\alpha_i$’s
- Calculate the MSE between $\tilde{t}(k)$ and $\tilde{t}(k)$

Classify the test instance as the target whose templates resulted in the lowest MSE

This technique was tested for systems with 1-4 templates. Thresholding was applied to this classification structure the same way it was applied to the minimum distance classifier. The classification results are compared to the persistence minimum distance case in Table 5-3. For systems with only a few templates available, the persistence modeling and minimum distance classification scheme outperform the angularly dependent data with PCA basis classification.
5.4 Conclusions

The framework for persistence modeling was developed to turn a set of aspect dependent images into a set of images that are a function of the target features. Specifically, mapping functions are defined to transform the angularly dependent data into a space that is intensity-based and feature dependent. For testing purposes, the framework was implemented using simplified mapping functions. The framework allows for more advanced dynamic cost functions to be defined if needed. The application of the persistence model to visualization and classification were investigated.

Visualization using the persistence model works well. With angularly dependent data only part of a target is observable within any given single image; therefore, it is difficult to process visually the specific features of a target. The persistence model allows for visualization as a function of the targets’ features. This principle was illustrated for two different targets.

Although it was not designed as an optimal classification framework, the testing of a template classification structure reveals the persistence model’s potential. For systems with full templates, the persistence model could not perform as well as the

<table>
<thead>
<tr>
<th>Number of templates</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Classification Rate</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PCA (angular)</td>
<td>.5679</td>
<td>.5927</td>
<td>.6637</td>
<td>.7498</td>
</tr>
<tr>
<td>Min Distance</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(persistence)</td>
<td>.6175</td>
<td>.7484</td>
<td>.7567</td>
<td>.7636</td>
</tr>
</tbody>
</table>

Table 5-3: Classification in reduced template spaces
traditional framework. Under the assumption of a reduced set of templates however, the persistence model outperforms the angularly dependent dataset. For similar classification complexity costs and a classification rate above 75%, persistence modeling allows for a 10+ fold reduction in the number of templates needed. If a more complicated classification structure is implemented for the angularly dependent data, the persistence model still has a higher classification rate for systems with fewer than 5 templates. These experiments illustrate the potential of the persistence model to be used for classification in sensor systems where computational costs and storage are limited.
Chapter 6
Shadow Analysis

6.1 Problem Overview

SAR imaging is a technique of NCTR that has shown the ability to identify targets at long ranges in adverse conditions. These traditional SAR classification techniques focus on the EM scattering centers. A recent thrust, however, has been to use the shadow information present in SAR imagery [33] [55]. Current literature does not include any robust techniques that are capable of reliable classification using only shadow information. Classification via shadow information is difficult because the shape is dependent on the radar aspect angle, depression angle, and resolution. The purpose of this chapter is to investigate the feasibility and performance of shadow-only classification structures and assess their integration into NCTR applications.

In Section 6.2 a variety of classification methods are developed for the classification of shadows using shape based features. One class of techniques relies on using Procrustes analysis in the shape space. Procrustes is often used in medical imaging applications where certain classes of shapes exhibit a high degree of variability [56]. Classification structures of varying complexity are derived around the point-based Procrustes features. Alternatively an ensemble hidden Markov scheme is used in conjunction with a chain coding technique for classification.
Small scale tests were performed using a reduced number of images for a 2 target MSTAR system. The goal of Section 6.3 is to analyze the classification structures and determine the most appropriate classification method. Since, the HMM will be shown to have the greatest potential it was further analyzed for general performance, variable resolution, and variable depression angle.

The HMM classification structure was then more rigorously tested on a 5 target MSTAR system; the results are analyzed in Section 6.4. Specifically, the following were investigated: model implementation issues, model selection, ensemble averaging, single-look classification, multi-look classification, and integration with traditional classification systems. The final section, Section 6.5, investigated the problem of operation over a wide range of depression angles.

6.2 Classification Structures

6.2.1 Procrustes Analysis

6.2.1.1 Procrustes Overview

The first techniques that are being investigated are ones that operate on point-based shape features. These feature or landmark points should be homogenous across all images in the analysis. There are many techniques available that attempt to automatically solve the correspondence problem between sets of images [57] [58] [59]. However, for this research the point sets were obtained manually to minimize algorithmic complexity.
To begin, a shadow was automatically segmented out of a SAR chip. Next, points on the segmentation boundary that correspond to the key points within the shadow region were manually segmented. Figure 6-1 shows the key that was used to select the points uniformly in all images. Key point 1 in the shadow image is the leftmost-bottom point in the SAR image that borders the highlight region.

![Figure 6-1: Key point feature selection](image)

In order to perform the initial shape analysis, the shadow needed to be defined in terms of boundary key points. Three key points were used to triangulate the entire shape; four points were then defined according to the boundary between each of the initial key points.

Once the points are chosen, the image sets can be analyzed using Procrustes analysis [60]. Procrustes analysis allows the points to be transformed such that the new dimensionality of the data corresponds to a shape-space. The shape-space carries statistical information about the shape ensemble defined by their respective key points. The transformation is the minimization of the sum of squared errors of one point set transformed to another point set, Eq. 6.1.

\[
\min_{\hat{T}} \| \mathbf{x}_1 - \hat{T}(\mathbf{x}_2) \|^2
\]  

[6.1]
The first step in the alignment process of two shapes is to translate the objects such that their respective centers of mass lie at the origin. Next, the objects’ scales are normalized to 1. Normalization is performed by dividing by the square root of the sum of squares coordinate distances. The third step is to rotate one shape such that it is aligned with a given reference shape. The rotation matrix can be solved for using singular value decomposition, Eq. 6.2, where the corresponding eigenvectors and eigenvalues are given in Eq. 6.3 and Eq. 6.4. $A$ and $B$ are defined as centered and normalized pre-shapes with $N \times 2$ points each.

$$R = VW^T$$  \hspace{1cm} [6.2]

$$(A^T B)(A^T B)^T = VDV^T$$  \hspace{1cm} [6.3]

$$(A^T B)^T (A^T B) = WDW^T$$  \hspace{1cm} [6.4]

This procedure can be applied directly to any set of two images. When multiple images are being oriented, the process must be iterated. For multiple images, all the images in the dataset are centered and normalized. Next, an arbitrary shape is selected as the reference shape. Each shape in the dataset is aligned to this reference shape. A new reference shape is obtained by calculating the mean point for each of the key points. The new mean shape is then centered and normalized; the process is repeated until convergence is met [60].
After the Procrustes aligning is completed, the set of aligned shapes are described by Eq. 6.5. Using a parameterized model, the shapes can be defined in terms of the mean shape $\bar{x}$, a transformation matrix $P$, and a descriptor vector $b$.

$$x = \bar{x} + Pb$$  \[6.5\]

The mean shape is defined directly as a result of the Procrustes alignment procedure. $P$ is defined by the eigenvectors of the points’ covariance matrix. Instead of the $(x, y)$ point coordinates in the image space, the descriptor vector $b$ describes the shape in the new shape space. Principle components analysis can also be applied to reduce the dimensionality of the shape space. The shapes are therefore given by Eq. 6.6, where $\tilde{P}$ is the $m$ eigenvectors that correspond to the $m$ largest eigenvalues.

$$x \approx \bar{x} + \tilde{P}b$$  \[6.6\]

Once in this new reduced-dimensioned shape space, common shapes will reside in the same local area. It should therefore be possible to distinguish between two targets based on the shape description vectors. Sections 6.2.1.2-6.2.1.5 outline four methods of varying complexity, which may be capable of distinguishing targets based on these shape space vectors.

### 6.2.1.2 Minimum Distance Classifier

The simplest of the four classification strategies is the Procrustes distance classifier. From each set of target shapes, a mean shape is obtained using Procrustes analysis. This process is illustrated in Figure 6-2.
A new test instance is then aligned with mean shape 1 and the distance between points to mean shape 1 is calculated. The same is done for mean shape 2. The test instance is classified as target 1 if the distance to target 1 is less than that to target 2. Similarly, the target is classified as target 2 if the distance to mean shape 2 is less than the distance to mean shape 1 [61]. The distance measure being used is the Procrustes distance, the sum of squares distance after aligning the shapes.

Let $x_T$ denote the test image. Let $x_T^{\bar{x}_n}$ denote the Procrustes alignment of $x_T$ with $\bar{x}_n$. The minimum distance classifier can then be given by Eq. 6.7.

$$\sum_{T=1}^T |\bar{x}_1 - x_T^{\bar{x}_n}|^2 < \sum_{T=2}^T |\bar{x}_2 - x_T^{\bar{x}_n}|^2$$ [6.7]
6.2.1.3 Gaussian Likelihood Classifier

The next step in classifier complexity is to use the statistics of the shape space descriptors. Dimensionality reduction using PCA is performed in the shape space based on the $m$ largest eigenvalues of the covariance matrix. Figure 6-3 shows the flow diagram for obtaining the descriptors.

![Flow diagram](image)

Figure 6-3: Shape descriptors in the shape space

A set of target shadow shapes can be represented in a reduced dimensional shape space by combining the Procrustes alignment procedure and PCA.

An initial training image can be completely described in shape space 1 by, Eq. 6.8. The set of $M$ training images is described by Eq. 6.9.

$$b_n^1 \approx P^{-1} (x_n^1 - \bar{x}) \quad [6.8]$$

$$B^1 = [b_1^1, b_2^1, \ldots, b_M^1] \quad [6.9]$$
As an initial estimate, the descriptors are assumed to be Gaussian distributed. Therefore the statistics of the set of descriptors $B^1$ can be completely described by the mean and covariance. By definition the mean shape will have a zero shape descriptor in the shape space; the multivariate Gaussian is zero mean. Also because of the transformation used in the shape space, the covariance matrix will be diagonal. Using these factors the likelihood of observing a particular shape descriptor can be given as Eq. 6.10.

$$p(b_n^1 | B^1) = \frac{1}{(2\pi)^{m/2} \mid \Sigma \mid^{1/2}} e^{-\frac{1}{2} (b_n^1 - \mu_1)^T \Sigma_1^{-1} (b_n^1 - \mu_1)}$$  \[6.10\]

Now, let $x_t$ denote a test instance, $b_t^1$ denote the descriptors of the test instance in the shape space of target 1, and $b_t^2$ denote the descriptors of the test instance in the shape space of target 2. Assuming equal prior probabilities of observing the two targets the likelihood ratio classifier is given in Eq. 6.11.

$$p(b_t^1 | B^1) \approx \frac{1}{\sum_{T=2}^{T=1} p(b_t^2 | B^2)}$$  \[6.11\]

6.2.1.4 K-Nearest Neighbor Classifier

The next level of complexity used for classification is the K-nearest neighbor classifier. There are two basic methods to perform the analysis – single space reference, multi-space reference. The full reference space and template generation is shown in Figure 6-4.
The first method of single space reference is illustrated in Figure 6-5. A test instance is projected into the shape space of target 1 and the shape space of target 2. The distances from its descriptors in space 1 to the template descriptors (the descriptors of training set 1) are calculated. The distances from its descriptors in space 2 to the template descriptors of set 2 are also calculated. The class of the test instance is chosen to be the majority class of the $K$ smallest distances to the templates.

Figure 6-4: Template generation for K-nearest neighbors

The flowchart outlines the procedure of template generation for the nearest neighbor classification structure. Each template is projected into the shape space of all possible targets.
The main problem in performing K-nearest neighbors in this method is that there is no guarantee that the distances in one shape space are equivalent to the distances in another. Moreover, if signal space 2 has a lower variance than signal space 1, the distances are going to be lower from the projection into signal space 2. This will cause the classifier to unnecessarily bias signal space 2.

To correct the bias, all images are projected into the shape spaces of both targets, Figure 6-6. This is required because a test instance is of unknown class. Therefore to obtain a descriptor, the test instance must be projected into each shape space. One common space could be used for all images, but that would significantly reduce the class separability. In order to calculate the distance from this new extended shape vector, each individual template image must also be projected into all shape spaces. The distances from the test instance to each of the $M$ templates for targets 1 and 2 is given by Eq. 6.12 and Eq. 6.13 respectively.
The majority class of the $K$ minimum distances is used for classification of the test shape.

6.2.1.5 Multilayer Perceptron Neural Network

The final classification structure is a multilayer perceptron (MLP) neural network. The network was designed to have $T \times D$ input nodes, where $T$ is the number of targets and $D$ is the length of a shape descriptor in a single shape space. As in Figure 6-4, each training shape is projected into all shape spaces to obtain an extended classification shape.

Figure 6-6: $K$-nearest neighbors in a multi-space reference

In the multi-space reference, the test instance and all of the templates are projected into all possible shape spaces. These projections are necessary to ensure that there are uniform distances between each test shape and all templates. The tradeoff is increased computational cost.

\[
d_n^1 = \sum \left[ \|b^1_i, b^2_i\| - \|b^1_{1,n}, b^2_{1,n}\| \right]^2, \quad n = 1, \ldots M \tag{6.12}
\]

\[
d_n^2 = \sum \left[ \|b^1_i, b^2_i\| - \|b^1_{2,n}, b^2_{2,n}\| \right]^2, \quad n = 1, \ldots M \tag{6.13}
\]

The majority class of the $K$ minimum distances is used for classification of the test shape.
vector. This shape vector is the input of the MLP. Figure 6-7 shows the overall NN structure and a log sigmoid function. The network has 1 hidden layer with 60 nodes and 2 output nodes, one for each class. For target 1, the output is ideally [1 0], for target 2 the output is ideally [0 1]. The log sigmoid function was used as the discriminate function.

Figure 6-7: Multilayer Perceptron neural network classification structure

A MLP NN structure is outlined with 1 hidden layer. There are two output nodes, one corresponding to each output class. A log sigmoid function was used as a discriminate function. Shape-space vectors and the raw coordinates were used as test inputs.

In addition to the case described above, the raw data points were also fed into the neural network. The purpose of this experiment was to determine whether the shape vectors carried more discriminatory information then the registered point coordinates alone. The input layer was scaled accordingly.
6.2.2 Hidden Markov Modeling

6.2.2.1 Overview

Next, a statistical modeling approach was taken in an attempt to model and classify the target shadows. Similar to the Procrustes analysis, the extraction of the target shadow from a SAR image is the first step in the classification process. The input to the classification system is a smooth, background-free representation of the target shadow. When modeling the shadows, assuring that the shadow boundaries are smooth and free from clutter effects is a crucial step in the segmentation procedure. [33] showed that the clutter alone could produce a classification rate of over 95% for the MSTAR database. Therefore, it is imperative that the extracted shadow is independent from this clutter. The segmentation algorithm developed in Section 3.3.3 achieves this goal.

Once extracted from a SAR image, the shadow classification problem is analogous to other 2D shape recognition applications [62] [63]. A chain coding technique is used to represent the shadow’s shape; chain coding is used because it takes advantage of a SAR image’s resolution cells. Multiple Hidden Markov models (HMMs) are then trained for each target in the system. A subset of models is then selected such that the overall classification rate of a given set of targets is optimized. The overall training procedure is illustrated in Figure 6-8 (a); the details are provided in the subsequent methodology sections.

The classification process employs the same segmentation and chain coding procedure as training. The likelihood of observing a given test chain code is calculated for each target model in the system. For robustness, an ensemble system is being used
for the large scale tests; three models are selected to represent each target. The likelihood of observing a chain code from each target is simply taken as the average over the three target models. The target class is chosen to correspond to the model that has the maximum average likelihood. Figure 6-8 (b) shows an illustration of this procedure, and the details for each are provided in the respective sections below.

Figure 6-8: Statistical shadow modeling

Figure (a) is a flowchart representing the estimation process of a HMM for a single target. For each SAR image, the shadow is extracted and encoded using a chain code. Using this data a HMM is trained. For each target a set of such models is trained. The optimal 3 models are selected for each target and subsequently used in testing. The test procedure (b) requires the same image processing steps. The likelihood of observing the test image for a given target is averaged over the three target models. The test instance is recognized as the target with the highest likelihood.
6.2.2.2 Chain Code Representation

The 2D binary shadow information of a SAR image is coded using a chain code representation [64]. The chain code representation allows the shadow to be evaluated in the range and cross-range directions. This is convenient since the x and y axes of a SAR image are given by the cross-range and range respectively. No correlation or co-registration between images is needed since SAR processing naturally defines the image in the desired coordinate directions.

A chain code is defined by tracing along the edge of a discretized realization of the shadow profile. Movement from point to point on the contour is restricted to 1 of 8 different directions in the range / cross-range resolution cells. These 8 states correspond to the 8 observable states in a HMM. Figure 6-9 (a) shows the chain code definition, and (b) is an illustration of one such code. Although other representation methods exist (spline fitting, continuous curve approximations, invariant moments, etc…) chain coding is seen to be computationally efficient and a good first order discrete instantiation of semantic shape-based information [64].
6.2.2.3 Generating Target Models

As stated above, the discrete HMM consists of a set of hidden states $S$ and a set of observable symbols $V$, where the observable states are the chain code outputs. The HMM can then be defined by $\lambda$, which is the set of parameters $\{A, B, \Pi\}$; these are the state transition matrix, emission matrix, and state priors respectively. The state transition matrix represents the probability of going from state $S_i$ to state $S_j$, Eq. 6.14

$$A = \{a_{ij}\}, \quad a_{ij} = P[n_{t+1} = S_j \mid n_t = S_i] \quad [6.14]$$

Similarly, the emission matrix can be defined as observing the symbol $V_k$ when the system is in state $S_j$, Eq. 6.15.

$$B = \{b_j(k)\}, \quad b_j(k) = P[V_k \text{ at time } t \mid n_t = S_j] \quad [6.15]$$
Using these quantities, there are three general problems when dealing with HMM’s:

- **Evaluation**: Determine the probability that a sequence of visible states was observed given the model parameters \( \{A, B, \Pi\} \).
- **Decoding**: Determine the set of hidden states that most likely resulted from a given sequence of observed states and the model parameters.
- **Training**: Assuming the HMM structure is known, determine the model parameters from N sets of observed sequences \( \{O^{(i)}\}_N \).

The training mode is used to estimate a suitable model for a given target; \( \lambda \) is calculated such that \( P(\{O^{(i)}\}_N | \lambda) \) is maximized. For this implementation, the Baum-Welch re-estimation technique is applied. This technique is based on Expectation-Maximization and attempts to maximize the log likelihood of the model for the given training data [43].

Specifically, the training data is a set of \( N \) SAR images taken from the same depression angle at multiple aspects. Equally spaced aspects would be the most representative training sequence, but is not necessary for training. Once the chain codes are obtained for all training images, the model parameters can be estimated. This procedure is illustrated in Figure 6-8 and is repeated for each target to be classified. Given a set of training data, multiple learning sessions will yield classifiers with different capabilities because of random components and initialization in the training procedure. Multiple models, \( M \), are therefore trained for each of \( T \) targets.
6.2.2.4 Shadow Classification Structure

Once a set of models is trained and selected, a new test instance can be classified by evaluating the targets’ models. A test SAR image depicting an unknown target is again segmented and transformed into its respective chain code. The probability of observing the test sequence is calculated for each of the target models. The test image is recognized as target $k$ according to Eq. 6.16.

$$P(O^{(i)} | \lambda^k) = \max_n P(O^{(i)} | \lambda_n)$$  \[6.16\]

This structure allows for new targets to be added to the classification problem very easily. The only new training that needs to take place would be to train a HMM based on the new target.

To make the classification scheme more robust \[65\], the likelihood was calculated on 3 HMMs for each target. The log-likelihoods were averaged before a classification decision was made. The ensemble classification rule is therefore given by choosing the class $k$ according to Eq. 6.17.

$$P(O^{(i)} | \lambda^1_k, \lambda^2_k, \lambda^3_k) = \max_n \sum_{j=1}^{3} \log P(O^{(i)} | \lambda^j_n)$$  \[6.17\]

6.2.2.5 Model Selection

After training there are $M$ models for each of $T$ targets, Eq. 6.18.

$$\Lambda = \{\lambda^1_i, ..., \lambda^T_i\} \quad i = 1, ..., M$$  \[6.18\]
The goal of model selection is to choose the subset of models such that the classification rate is maximized. To accomplish this, the classification rate is evaluated over a validation dataset. From $\Lambda$, there are $M^T$ combinations of possible model sets.

All $M \times T$ models are used individually to calculate the likelihood of each of the possible validation images; the likelihood data is stored in a matrix. Next all possible combinations of models are evaluated for classification performance on the validation data. Since the likelihood was pre-calculated, the evaluation procedure consists of indexing a variable and finding the maximum value from a matrix. It is therefore computationally feasible to do a brute-force search for the optimal model combinations; the search operates on the order of minutes. The top 3 model sets are used in the HMM ensemble during the large scale testing; a single model was used for the small scale and preliminary tests. Alternatively, other optimization strategies are explored during large scale testing.

For the selection procedure outlined above the primary evaluation criterion was the overall system classification rate. This metric is used almost exclusively for reporting the end system performance. There are, however, other probabilistic quantities that can be employed to obtain varied system operations; these techniques are also evaluated during large scale testing.

The notation $c_i$ is used to describe the class output from the $i^{th}$ model. The class output from the classifiers is a set belonging to any of $N$ possible targets, Eq. 6.19. This is a closed target system in that a single classifier cannot output a ‘no decision’.

$$c_i \in \{\omega_1, \omega_2, \ldots, \omega_N\} \quad [6.19]$$
The two probabilistic quantities exploited during model selection are:

- \( p(c_i = j \mid \omega_j) \) is defined as the probability the \( i \)th model set chooses class \( j \) given that the correct class is class \( j \). This quantity is the probability of detection and is the most common measure reported.

- \( p(\omega_j \mid c_i = j) \) is defined as the probability the correct class is class \( j \) given that the \( i \)th model set chooses class \( j \). This quantity gives a measure of confidence that the output of a given model set is correct. Maximizing this quantity is equivalent to minimizing the number of false alarms for a given class. Maximizing the probability of the true class \( j \) being observed given that class \( j \) is output from the model set is equivalent to minimizing the number of times class \( j \) is decided given that \( j \) is not the true class.

Initially, the top model combinations were selected such that the probability of detection was maximized over all targets, Eq. 6.20.

\[
m^{\omega_n} = \max_{m^{\omega_n} \in \{m_1, \ldots, m_\omega\}} \sum_{j=1}^{N} p(c_i = j \mid \omega_j) \quad \forall \omega_n \quad n = 1, \ldots, N \quad [6.20]
\]

An alternative approach to selecting the 3 classifiers that have the best global performance is to select models that are maximized over a single target. Therefore, in this example an ensemble of 5 would be needed; one model set for each target being classified. Additionally, it is not necessary to only consider the probability of detection. The selection process can therefore be given in terms of a general criteria function \( r \), Eq. 6.21.

\[
m^{\omega_n} = \max_{m^{\omega_n} \in \{m_1, \ldots, m_\omega\}} r(c_i, \omega_n) \quad \forall \omega_n \quad n = 1, \ldots, N \quad [6.21]
\]

Notice that the \( \omega_n \) model is only evaluated on the target validation instances for class \( n \). Four different criteria function are investigated and are designated as MP, MP2, MP3, and MP4.
The first criterion is ‘loud-mouth models’ (MP). These models maximize the probability of detection for a single target as given in Eq. 6.22. The MP classifiers are ‘loud-mouth’ since they heavily bias their given target. The classifier is evaluated only on the validation instances of a single target. The classifier that gets the largest number of those test instances correct is chosen.

\[
m^{\text{MP}} = \max_{m_{i}^{n} \in \{m_{1}, \ldots, m_{5}\}} \ p(c_{i} = j \mid \omega_{j}) \quad \forall \omega_{n} \quad n = 1, \ldots, 5 \quad [6.22]
\]

The next criterion is ‘single-target cautious models’ (MP2). These models simultaneously maximize the probability of detection for a single target and minimize the probability of false alarm for that same target, Eq. 6.23. One of the drawbacks of using a loud-mouth model is its bias towards a single target. A model that always outputs target 1 would have a perfect probability of detection if evaluated on target 1; clearly this is not the most optimal model to select. So in addition to maximizing the probability of detection the number of false alarms is jointly optimized in MP2. It is important to note that although the confidence probability is maximized with respect to only a single target, it requires evaluation over all possible validation images.

\[
m^{\text{MP2}} = \max_{m_{i}^{n} \in \{m_{1}, \ldots, m_{5}\}} \ p(c_{i} = j \mid \omega_{j}) p(\omega_{j} \mid c_{i} = j) \quad \forall \omega_{n} \quad n = 1, \ldots, 5 \quad [6.23]
\]

The third criterion is ‘global, single-target cautious’ (MP3). These models simultaneously maximize the probability of detection for a single target, minimize the probability of false alarm for that same target, and maximize the probability of detection for the remaining targets, Eq. 6.24. In MP2 the probability of detection in calculated only over the validation instances of a single target; similarly, the confidence probability is maximized with respect to only a single target. The incorporation of the confidence
probability aids in classification performance over multiple targets, but is not a direct measure of it. MP3 maximizes over the same quantity as MP2, but also includes a global probability of detection term. This term ensures that while the performance of a single target is optimized the performance of the remaining targets is not compromised.

The final criterion is ‘global, single-target loud-mouth’ (MP4). These models maximize the probability of detection over all targets, but count all instances of a single target twice, Eq. 6.25. MP4 has the same single-target and global probability of detection terms as MP3. MP4 does not utilize the confidence probability. The MP4 selection criterion is therefore a measure of the global probability of detection, with an emphasis on the boundary corresponding to a single target.

\[
m^{\text{as}} = \max_{m_0^n \in \{m_1, \ldots, m_y\}} p(c_i = j | \omega_f) p(\omega_f | c_i = j) \sum_{j=1}^5 p(c_i = j | \omega_f) \quad \forall \omega_f \quad n = 1, \ldots, 5 \tag{6.24}
\]

The final criterion is ‘global, single-target loud-mouth’ (MP4). These models maximize the probability of detection over all targets, but count all instances of a single target twice, Eq. 6.25. MP4 has the same single-target and global probability of detection terms as MP3. MP4 does not utilize the confidence probability. The MP4 selection criterion is therefore a measure of the global probability of detection, with an emphasis on the boundary corresponding to a single target.

\[
m^{\text{gs}} = \max_{m_0^n \in \{m_1, \ldots, m_y\}} p(c_i = j | \omega_f) \sum_{j=1}^5 p(c_i = j | \omega_f) \quad \forall \omega_f \quad n = 1, \ldots, 5 \tag{6.25}
\]

6.2.2.6 Multiple Looks

Additionally, in modern battlefield situations it is common that a target is imaged by more then radar. Through the use of radars of opportunity and multiple radar bases, a target can be imaged from multiple locations. Multiple location processing improves target detection capabilities since the different aspects or ‘looks’ provide additional clues about the shape, dimensions, and special features of a target.
To incorporate such scenarios, the basic classification framework can be expanded. The ensemble likelihood of observing a single chain code $O^{(i)}$ given target model $k$ is defined by Eq. 6.26.

$$\Lambda_k(O^{(i)}) = P(O^{(i)} \mid \lambda_k^1, \lambda_k^2, \lambda_k^3)$$ \[6.26\]

This quantity is calculated through the HMM decoding process and ensemble averaging. During a multi-look scenario, multiple images are available and therefore multiple chain codes are used. The joint likelihood can be expressed as a product of the individual likelihoods since the images are assumed to be obtained independently, Eq. 6.27.

$$\Lambda_k(O^{(i_1)}, \ldots, O^{(i_p)}) = \prod_{j=1}^{p} \Lambda_k(O^{(i_j)})$$ \[6.27\]

### 6.2.3 Integration with Scattering Center Classifiers

The final aspect of evaluating the shadow-only classification system is to determine the feasibility and potential benefits of integrating it with a traditional highlight classifier [55] [66]. For example, assume the highlight classification system has a high level of performance and the shadow-only classification scheme has a slightly lower classification rate. The shadow-only classifier will not be useful if the instances it correctly classifies are simply a subset of the instances that the highlight classifier already correctly identified. Instead it is desirable to have the classification rates of the two classifiers be independent [67].
The independence of the classifier outputs can be easily tested. As seen in Figure 6-10 a set of test images are subjected to both types of classifiers. Each test instance is then correctly or incorrectly classified by each of the two classification systems.

\[ P(S \cap H) = P(S) \times P(H) \]  

**Figure 6-10**: Integration of classifiers

This figure shows the procedure used to determine the relationship between the classification rates of the two classification methodologies. It is desirable that the probabilities of correct classification from the two systems be independent.

The events of correct and incorrect classification for each system can be formalized as:

- \( S \): The correct classification of a test instance for the shadow-only classification system
- \( H \): The correct classification of a test instance for the highlight classification system

The probabilities of the events above are easily obtained from the testing dataset. Independence can then be tested as Eq. 6.28.

In order to test the above condition a nearest neighbor classification system was implemented for the highlight data. There is no assertion of optimality of this scattering center classifier. The purpose of the classifier is to show its performance and the
performance from the shadow classifier are complimentary. The classifier was similar to that used in [33]. The pseudo-code for this classification system is given below.

```
For: All images
- Rotate to adjust for known aspect angle
- Resize to 16x16
- Discard values lower than specified threshold

For: All test images
- Calculate distance of test image to all templates in the database
- Find the nearest 3 neighbors
- Classify as the majority class of the nearest neighbors. In case of tie, use class of single nearest neighbor
```

6.3 Small-Scale Tests

To initially test the feasibility of the classification structures listed in Section 6.2, a set of small-scale tests were performed. A two target system was selected from the MSTAR database; specifically the BTR60 personnel carrier and the T72 battle tank were used. A sub-sampled set of 42 images were selected from all possible images (25 tanks images and 17 personnel carrier images). The images ranged from 0° to 360° in aspect and were collected at 15° depression. These images can be found in the Appendix.

Section 6.3.1 overviews the Procrustes shape analysis performed on the sample shadows. Section 6.3.2 then performs the classification tests. From these tests it emerges that Hidden Markov modeling of the shadow shapes is the most suitable method. Therefore, further experimentation is performed to assess its robustness and feasibility for large-scale testing.
6.3.1 Procrustes Analysis

As described in Section 6.2.1.1, the first step in performing a shape analysis is to obtain the necessary point correspondences. Figure 6-11 shows a sample T-72 SAR chip, the estimated segmentation, and the selected shadow points.

![Figure 6-11: Point selection](image)

The key point selection was performed for all 42 images in the database. The raw data points for the segmented shadow regions for each class are given in Figure 6-12.

![Figure 6-12: Unregistered segmentation points](image)

The key point selection was a manual process.

The figure illustrates an example of key point selection on a segmented image.
Procrustes analysis was then performed for each set of images given above. The resulting mean shapes, and point correspondences are shown in Figure 6-13. Plots (a) and (c) show the individual point correspondences while (b) and (d) give the shape outlines. From both the scattered points and the line shapes it is clear that the points are not tightly distributed around the mean shape. The variance in the location seems to be slightly higher for the T-72 compared to the BTR60. This results indicates that the shadow of the tank varies more then the personnel carrier. This observation matches reality since the tank has significantly more structure then the BTR60.
After the Procrustes alignment was performed, the shape statistics for each target could be examined. Figure 6-14 shows the eigenvalues for both cases. There seem to be approximately 15 eigenvalues that contribute substantially to the shape. The eigenanalysis also mirrors the increased complexity of the T-72 over the BTR60.
After examining the eigenvalues, 15 components were kept when performing dimensionality reduction on the shadow information. Since a 15-dimensional space cannot be plotted, the coefficients are plotted as a line with 15 taps in Figure 6-15. Again the variations in the T72 are more apparent than in the BTR60. The ±3σ limits are illustrated for each dimension individually.

Figure 6-14: Eigenvalues for shadow points

From the eigenanalysis it is seen that approximately half of the points are significantly greater than zero. Also, more points are needed to describe the T-72 compared to the BTR60.

After examining the eigenvalues, 15 components were kept when performing dimensionality reduction on the shadow information. Since a 15-dimensional space cannot be plotted, the coefficients are plotted as a line with 15 taps in Figure 6-15. Again the variations in the T72 are more apparent than in the BTR60. The ±3σ limits are illustrated for each dimension individually.

Figure 6-15: 15-dimensional shadow shape space

The shape space for each target is shown in 15 dimensions. Each shadow is displayed as an individual line. All of the features are seen to be zero mean. The features that have the largest eigenvalues are also seen to have the greatest range and variance.
Once in the shadow shape-space, coefficients can be generated to obtain model shapes. Figure 6-16 and Figure 6-17 show the variations of the first and second modes for the T72 shadow. The first mode of variation is a compression and stretch. Large positive values for the primary mode compress the shadow horizontally and expand it vertically. Conversely, large negative values will compress the shadow vertically and stretch it horizontally. These variations are not absorbed into the rotation since the key point selection was fixed.

The second mode of variation appears to be the concavity of the lower-right subsection. Large positive values create a divot in the shadow, while large negative values create a protrusion similar to that of the gun turret. Another mode of variation is for the lower-left subsection. These variations could be absorbed together if a mirroring function was also incorporated into the alignment procedure.

![Procrustes Points T-72 Mean Shape Projected Shape](image)

(a) Large negative values (b) Neutral (c) Large positive values

Figure 6-16: First mode of variation

The first mode of variation within the T72 shape described the overall compression of the target. Large negative values compressed the shadow along the vertical axis, and large positive variations compressed the shadow along the horizontal axis.
Clearly, the shadow variations for the T72 have been captured in the Procrustes shape space. Procrustes analysis and PCA allow for the calculations of a mean shape, shape-space coefficients, and location statistics. The classification section discusses the appropriateness of these methods compared to Markov modeling and provides insight into the ability of these descriptors to uniquely represent the given targets.

6.3.2 Classification

Next, each of the 5 classification structures was tested. Since the training classification rates do not give a true indication of a classifiers’ performance, leave one out testing was performed. When testing a particular realization, all realizations except for that single realization were used in obtaining the shape statistics and training of the respective classification structure. The classification rate is then obtained by determining how many instances were classified correctly. Figure 6-18 below shows a plot of the testing classification rates.

![Figure 6-17: Second mode of variation](image)

The second mode of variation was the concavity of the lower-right section. A large negative value is similar to the variation caused by the turret of the gun.
The distance-to-mean-shape classifier only takes into consideration the average shape and not the variations in shape; it had the lowest classification rate. The shadow regions vary drastically with aspect and manifest in significantly different shapes; higher order statistics are needed. The multivariate Gaussian distribution takes some of the variations into account by using the first and second order statistics. Still, the overall performance is quite low with the likelihood classification structure. Similarly, the raw point coordinates input to a neural network offer little, if any, potential for classification.

The 3-nearest neighbor, multi-layer perceptron neural network, and hidden Markov model structures were able to classify the shadows with relatively acceptable rates of accuracy. Table 6-1 shows a comparison of these three methods against other recently published results. Both [33] and [55] used a spatial mask to obtain the shadow region. After masking, [55] uses an interpolated version of the spatial frequencies as the

Figure 6-18: Classification rates using leave-one-out testing

The plot shows the classification rates for the various classification structures. Since there were a limited number of instances available, leave-one-testing was used. The HMM classification structure is seen to perform the best. The Procrustes methods were seen to outperform the normalized point coordinates, but some methods (NN and MLP) outperformed others (distance to mean shape, and likelihood).
input to an SVM classifier. Their mask, however, does not filter out all of the scattering centers and contains significant clutter. Both factors indicate that their reported performance is not an indication of a shadow only classification method.

The methods presented in this chapter offer some amount of improvement over [33] and [55]. The algorithms were tested on the same data, but not the same subsets of that data. Therefore one cannot conclude with certainty that the newly developed classifiers are superior. It does however show that the use of Procrustes shape-space features are useful when classifying targets based on the shadow, and that a hidden Markov modeling structure shows even more potential.

The Markov modeling approach is more appealing over the Procrustes methods for a number of reasons. First, the classification rate using the HMM implementation was significantly higher compared to any of the classification structures utilizing the Procrustes based analysis. Next, the Procrustes analysis requires point correspondences between shapes. Since the shadows vary drastically from target to target and aspect to aspect the calculation of these correspondences is a limitation. Also, the Procrustes classification structures require knowledge of all of the targets in the systems; the projection of feature vectors into multiple targets’ shape spaces is required. This basic limitation makes altering the target set a non-trivial problem. Conversely, the HMM

<table>
<thead>
<tr>
<th></th>
<th>[33]</th>
<th>[55]</th>
<th>KNN</th>
<th>MLP</th>
<th>HMM</th>
</tr>
</thead>
<tbody>
<tr>
<td>BTR60</td>
<td>-</td>
<td>66.8%</td>
<td>82.4%</td>
<td>53.4%</td>
<td>88.20%</td>
</tr>
<tr>
<td>T72</td>
<td>-</td>
<td>69.6%</td>
<td>60.0%</td>
<td>84.0%</td>
<td>100.00%</td>
</tr>
<tr>
<td>Mean</td>
<td>58.8%</td>
<td>68.2%</td>
<td>69.0%</td>
<td>71.0%</td>
<td>95.20%</td>
</tr>
</tbody>
</table>
target models are easily portable and can be adapted for situations with varied operating conditions or target sets. Section 6.3.3 is a further investigation of the HMM structure using the small-scale dataset. The final implementations and performance analysis of this classification structure is thoroughly tested in Section 6.4.

**6.3.3 Markov Modeling**

Since the Markov modeling procedure seemed to be the most appropriate, additional testing was performed. The performance on the small-scale data was examined in more detail: the classification outputs were examined; the algorithm was tested as a function of resolution; the performance was analyzed for varying simulated depression angles.

**6.3.3.1 General Performance**

In the initial tests only 8 hidden states were used. The transition and emission matrices were therefore both 8 by 8. Figure 6-19 shows the converged matrices for both targets.
For both targets, the emission matrices are almost diagonal. This indicates that the observed parameters in the image closely match that of the hidden space. Although the variations from the diagonal are small, they are still important. The variations help indicate the levels of changes in the observed chain code. Additional hidden states would allow for increased model fidelity.

Figure 6-19: Markov transition matrices

The figures represent the converged probability matrices used in HMM processing. The 8 visible states correspond to the 8 states of the chain code. A set of 8 hidden states was also used. This number was determined experimentally. Additional hidden states should be used to obtain higher model fidelity for cases with greater numbers of targets.
The T-72 shadows have more physical structures than the BTR60. This is observed probabilistically in the increased disparity of the transition matrix of the T-72 compared to the BTR60. These variations in the transition matrix indicate that the curvature of the T-72 contains more variation. The BTR60 images appear to be generally rounder than those of the T-72 so the transition matrix is more diagonalized. Also, the presence of the turret in some of the T-72 images accounts for additional variations.

Using the matrices obtained from a leave-one-out testing scheme, the log likelihood of observing the given test sequences were obtained for each of the two target models. Figure 6-20 (a) shows the likelihood outputs for the two models for T72 test images; (b) shows the two likelihood outputs for BTR60 test image. For the T-72 test images, the probability of observing the given sequence is always higher under the T-72 model than using the BTR60 model. This corresponds to a 100% classification rate. For the BTR60 test images, there are only two instances where the BTR60 HMM does not yield the higher likelihood; this corresponds to an 88.2% classification rate. Combining the two targets, the HMM structure can achieve a classification performance of 95.2%. Not only are the instances correctly classified there appears to be good separation between the models.
6.3.3.2 Variable Resolution

One of the variables in SAR images is the resolution; therefore it is important to consider the classification performance under variable resolutions. Also, the use of reduced sized images further tests the discriminatory power of the classifier. Figure 6-21 shows 4 different resolutions of the same shadow shape. Clearly, as the scale is reduced the discriminatory features on the target become more difficult to detect.

Figure 6-20: HMM likelihood outputs

The overall classification rate under leave-one-out testing was 95.2%. Of all the instances, only 2 BTR60 images were incorrectly classified. In most of the correctly classified instances there is good separation between the models. In one of the two incorrectly classified instances the separation is small. These two observations indicate that there should be good confidence in the model outputs.
In the first test, the HMMs were trained using full scale images. The test image was then tested using a variable scale. The results are presented in Figure 6-22. As the scale of the test images were reduced, the classifier began to favor the T-72 image. The T-72 was almost always classified correctly, but at 80% scale the classification of the BTR60 images dropped to almost 50%. The chain code was no longer a representative model because of the scale variations in training and testing. Next, both the training and test data were scaled by the same factor. The performance is reported in Figure 6-23.
Figure 6-22: Performance analysis for a fixed training scale factor

The figure shows the leave-one-out testing results when the training data is fixed at 100% scale and the testing data has a variable scale factor. When the test data is subjected to a scale factor lower than 80% the performance on the BTR60 test set is unacceptable. This indicates that a fixed scale training model can allow for some changes in resolution, but cannot account for a drastic drop in resolution.

Figure 6-23: Performance analysis for a matched training scale factor

The figure shows the leave-one-out testing results when the scale factor for the training data is matched to the scale factor of the test data. For all of the scales, the performance is acceptable. As expected, the performance tails off slightly as the resolution is reduced.
When the training and test images were scaled equally, the performance of the classifier is still high. The initial images were approximately 140 x 140. Even at 14 x 14 the HMM classifier produced a 71% classification rate. This proves that the discriminatory power of the classifier is in fact in the overall shape, not in edge perturbations or fine edge details present only in high resolution images.

6.3.3.3 Variable Depression Angle

Next, the algorithm was tested under conditions of simulated varying depression angles. In the first experiment the models were trained at 15° depression and tested at incremental depression angles; the results are reported in Figure 6-24. Next, the models were trained and tested at various matched depression angles; the results are reported in Figure 6-25. The shadows were transformed using the method derived in Section 6.5.3.

![Image of Figure 6-24](image-url)

**Figure 6-24:** Performance analysis for a fixed 15° depression angle

The models were trained at 15° depression, but the test images were subjected to various scaled depression angles. The performance of the system is seen to be highest for closely matched depression angles. The performance is lowest for a depression angle of +15°. Although there is some decrease, the models offer good performance over a range of depression angles.
As with the case of variable resolution, the HMM performance decreases as the test images deviate from the trained models. When trained at 15º, the algorithm has satisfactory performance up to a depression angle of approximately 20º. This indicates that there is some robustness built into this method to account for small changes in depression. At large changes, however, the models trained at 15º depression no longer capture the discriminatory information of the targets. Again, analogous to variable resolution, the HMM classification structure performs well when the models are trained under the same condition as they are tested.

Banks of suitable models could be created for robust operation. Optimal model sets could be selected from the bank based on the targets of interest and the radar operating conditions. For example, if the training images were obtained at 15º depression, a bank of HMMs could be simulated at varying depressions through the
projective transformation. When a new test image came in, the HMMs could be chosen such that the models had scales and depression angles that corresponded to the test image. Due to the nature of the classification structure, maintaining such a model bank is feasible for practical applications.

6.4 Large-Scale Tests

The following sections outline the large-scale tests performed for shadow classification. The HMM technique from the previous section is used with some minor improvements for implementation on a large scale. A 5 target classification system is considered where data at a 15° depression angle are used for training/validation and 17° depression angle data are used for testing. The implementation section, Section 6.4.1, outlines model initialization, model size, and convergence issues. Section 6.4.2 deals with the problems of model selection and ensemble averaging. Section 6.4.3 details the classification performance over single and multi-look scenarios. And finally, Section 6.4.4 investigates how well the shadow classification algorithms can be integrated with traditional scattering center classification algorithms.

6.4.1 Implementation

One important issue for Markov modeling is the initialization of the transition and emission matrices. Three techniques were investigated – random, diagonalized, and histogram-based. The histogram-based method was seen to be the most desirable. This
method is common among many HMM applications. For this section, the training data is every other sample in aspect for the 15° depression data. The testing data is the remaining aspect samples from 15°.

First, a histogram of the chain codes across all possible states is obtained. This histogram represents an ideal emission matrix for a model with a single hidden state. However, the model contains $N$ hidden states. The histogram is replicated for each hidden state and subjected to zero mean additive noise with a fixed variance. The additive noise allows for variations in the convergence process and will lead to varied solutions. For high noise variances, the solution becomes similar to that of a random initialization and is unstable.

Initially, there is no reason for any given hidden state to transition to another particular hidden state; the transition matrix is diagonal. However, due to data complexity the hidden states will in fact transition from state one to another. This is captured by blurring the diagonal matrix. Similar to the emission matrix, the transition matrix is subjected to additive noise. The initialized and converged matrices are given in Figure 6-26. For random, diagonalized, and histogram-based initialization the converged matrices were similar to those given in (c) and (d). This form closely matches the initialization of the histogram based method. The models were allowed to converge until the change in likelihood was less than $10^{-3}$. 
Another aspect of HMM design is the number of hidden states. Figure 6-27 shows the classification rate as a function of the number of states on the training and testing datasets. The classification rate on the training data increases roughly linearly with increasing number of hidden states. This implies that increasing the number of hidden states will have a positive effect on system performance with respect to the training data. As expected, the data did not show the same linear trends on the test data.
The performance of the system seemed to level off at 15 hidden states. Increasing the number of states beyond this did little to increase the performance. Table 6-2 gives the convergence times for various model sizes.

Figure 6-27: HMM performance as a function of the number of hidden states

The classification performance is reported as a function of the number of hidden states. For each ‘number of hidden states’ the performance is reported as the mean performance averaged over 8 model realizations. The performance increases linearly for training, but levels off for testing.

<table>
<thead>
<tr>
<th>Number of hidden states</th>
<th>10</th>
<th>15</th>
<th>20</th>
<th>25</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time to convergence [m]</td>
<td>1.6</td>
<td>3.8</td>
<td>8.5</td>
<td>9.6</td>
</tr>
</tbody>
</table>

Although the classification rate seems to level off around 15 states in Figure 6-27 (b) there is the issue of model variability. In (b) the classification rate was averaged over only 8 of realizations. The purpose of the subsequent analysis is to determine if a single realization with 25 hidden states is statistically more likely to have a higher classification rate than a realization with 15 hidden states. First, an increased model sample size was
obtained. The number of models trained was increased from 8 to 30 for 15 hidden states and to 24 for 25 hidden states. The raw classification data for each case is given in Tables 6-3 and 6-4.

Table 6-3: Model performance for 15 hidden states

<table>
<thead>
<tr>
<th>Model #</th>
<th>Target CR</th>
<th>Statistics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>10</td>
<td>80%</td>
<td>72%</td>
</tr>
<tr>
<td>11</td>
<td>81%</td>
<td>56%</td>
</tr>
<tr>
<td>12</td>
<td>64%</td>
<td>55%</td>
</tr>
<tr>
<td>13</td>
<td>81%</td>
<td>72%</td>
</tr>
<tr>
<td>14</td>
<td>76%</td>
<td>76%</td>
</tr>
<tr>
<td>15</td>
<td>71%</td>
<td>76%</td>
</tr>
<tr>
<td>16</td>
<td>78%</td>
<td>57%</td>
</tr>
<tr>
<td>17</td>
<td>77%</td>
<td>72%</td>
</tr>
<tr>
<td>18</td>
<td>66%</td>
<td>67%</td>
</tr>
<tr>
<td>19</td>
<td>77%</td>
<td>72%</td>
</tr>
<tr>
<td>20</td>
<td>76%</td>
<td>73%</td>
</tr>
<tr>
<td>21</td>
<td>74%</td>
<td>45%</td>
</tr>
<tr>
<td>22</td>
<td>77%</td>
<td>81%</td>
</tr>
<tr>
<td>23</td>
<td>79%</td>
<td>80%</td>
</tr>
<tr>
<td>24</td>
<td>71%</td>
<td>76%</td>
</tr>
<tr>
<td>25</td>
<td>77%</td>
<td>50%</td>
</tr>
<tr>
<td>26</td>
<td>77%</td>
<td>55%</td>
</tr>
<tr>
<td>27</td>
<td>73%</td>
<td>66%</td>
</tr>
<tr>
<td>28</td>
<td>65%</td>
<td>73%</td>
</tr>
<tr>
<td>29</td>
<td>79%</td>
<td>64%</td>
</tr>
<tr>
<td>30</td>
<td>80%</td>
<td>67%</td>
</tr>
<tr>
<td>31</td>
<td>75%</td>
<td>76%</td>
</tr>
<tr>
<td>32</td>
<td>76%</td>
<td>52%</td>
</tr>
<tr>
<td>33</td>
<td>54%</td>
<td>78%</td>
</tr>
<tr>
<td>34</td>
<td>79%</td>
<td>56%</td>
</tr>
<tr>
<td>35</td>
<td>73%</td>
<td>73%</td>
</tr>
<tr>
<td>36</td>
<td>60%</td>
<td>72%</td>
</tr>
<tr>
<td>37</td>
<td>77%</td>
<td>74%</td>
</tr>
<tr>
<td>38</td>
<td>77%</td>
<td>71%</td>
</tr>
<tr>
<td>39</td>
<td>84%</td>
<td>58%</td>
</tr>
<tr>
<td>40</td>
<td>85%</td>
<td>59%</td>
</tr>
</tbody>
</table>

71.57% 13.02% 67.83% 12.98%
The data in the tables above are represented via histograms in Figure 6-28 (a).

The mean and variance of the data was used under the assumption of Gaussianity to obtain the distributions in Figure 6-28 (b).

![Discrete histogram](a) ![Under assumption of Gaussianity](b)

**Figure 6-28:** Performance results for varying number of hidden states.

The histogram plot shows the general performance of the models trained using 15 and 25 hidden states. For the training data, the 25 state models are clearly superior. On the testing data, the two sets overlap. The same data was modeled as Gaussian and plotted in (b). The 25 state models have a slightly higher mean with lower variance than the models with 15 hidden states.

### Table 6-4: Model performance for 25 hidden states

<table>
<thead>
<tr>
<th>Model #</th>
<th>Training</th>
<th>Target CR</th>
<th>Statistics</th>
<th>Testing</th>
<th>Target CR</th>
<th>Statistics</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>72%</td>
<td>76%</td>
<td>50%</td>
<td>76%</td>
<td>74%</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>75%</td>
<td>82%</td>
<td>60%</td>
<td>82%</td>
<td>74%</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>78%</td>
<td>86%</td>
<td>70%</td>
<td>86%</td>
<td>77%</td>
</tr>
</tbody>
</table>

77.61% 11.84%
From the data, the 25 state model is better than the 15 state model for the training data. This however is not a significant conclusion; the performance on the test data is the important criterion. From the limited number of observations it is difficult to visually conclude which technique is superior. Therefore, in order to definitively declare one model-set better than the other a statistical analysis was performed. First, the concept of reliability was used to estimate the probability that one setup would produce a model with higher classification performance compared to the other. Two random variables, A and B can be defined:

- $A$: a random variable representing the classification performance for configuration 1
- $B$: a random variable representing the classification performance for configuration 2

The reliability measure is defined as $R = P(A > B)$. This can be estimated using data according to Eq. 6.29 where $N$ is the number of pairs $(a_i, b_j)$ such that $a_i > b_j$, for all $i = 1, 2, \ldots, r$, and $j = 1, 2, \ldots, s$ [68].

$$\hat{R} = \frac{N}{r \times s} \quad [6.29]$$

A second test, the Kolmogorov-Smirnov Goodness of Fit Test (KS Test), looks at the CDF of the two datasets [69]. The test then evaluates the distance between the CDF to determine if there is sufficient evidence to claim that the two datasets were not drawn from the same distribution. Table 6-5 shows the vernacular used according to the different significance levels, $\alpha$, of the test. The KS test is a non-parametric, distribution-free test and as such does not make any assumptions about the nature of the data’s distribution. The test is also most sensitive at the median of the dataset. In this case, median sensitivity (over tail sensitivity) is ideal since the test is used to determine whether the average model trained with 15 states is as good as the average model trained
with 25 states. The null hypothesis of the test is that ‘the two sets of classification rates come from the same distribution.

### Table 6-5: Vernacular for KS testing

<table>
<thead>
<tr>
<th>no evidence</th>
<th>little evidence</th>
<th>suggestive evidence</th>
<th>sufficient evidence</th>
<th>strong evidence</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\alpha=0.1$</td>
<td>$\alpha=0.05$</td>
<td>$\alpha=0.025$</td>
<td>$\alpha=0.01$</td>
<td></td>
</tr>
</tbody>
</table>

According to the reliability estimator, there is approximately a 99.17% chance that a 25 state HMM will result in a higher classification rate than a 15 state HMM for the training data. Similarly, according to the KS Test there is strong evidence to reject the null hypothesis; the two datasets do not come from the same distribution. This confirms the observations made earlier.

According to the reliability estimator, there is approximately a 68.61% chance that a 25 state HMM will result in a higher classification rate than a 15 state HMM for the testing data. According to the KS Test there is little evidence to reject the null hypothesis; the two datasets likely come from the same distribution.

For testing, even though the performance for a 25 state HMM has a 68.61% chance of being greater than that for a 15 state HMM there is little evidence to support the claim that the classification rates do not come from the same distribution. Using the timing simulations and the results from the performance analysis, Figure 6-29 shows a comparison of the two methods when the total number of models trained is fixed as compared to when the total training time is fixed.
For fixed training time, generating multiple models using 15 hidden states will result in better performance than generating models with 25 hidden states. In the time it takes to train 1 large model, 2.5 smaller models can be obtained. Since there is only a 68% chance of the large model resulting in a higher classification rate, it is more beneficial to generate larger numbers of the smaller models. This is subject to the condition of being able to pick the “best” model from all those trained. For a fixed number of models trained, however, it is more beneficial to use 25 hidden states. Since the number of models trained was fixed in this implementation, the models were trained with 25 hidden states.

Figure 6-29: Performance analysis

The performance analysis takes into account classification rate and training time. There is a 68% probability that a model trained with 25 hidden states has a higher classification rate than a model trained with 15 hidden states. A model with 25 hidden states takes approximately 2.5 times longer to train than a model with 15 hidden states. The analysis is subjected to the restriction that the ‘best’ models can always be selected; in practice that is not necessarily known.
6.4.2 Model Selection and Ensemble Averaging

After multiple models have been trained, the next step is to select the model(s) with the highest performance. In order to select the most appropriate model, however, an evaluation criterion is needed. The first criterion under consideration is the model likelihood as calculated on the training data. This quantity gives an indication of how well a particular model fits the data that was used to generate it. The second approach is to assess the classification rate on the entire training dataset. These two approaches are plotted in Figure 6-30 as a function of the classification rate on the validation data.

![Figure 6-30: Evaluation metrics on validation data for model selection](image)

The model likelihood and the training performance are illustrated as a function of the classification rate on the validation data. There is almost no correlation between the classification rate and the model log-likelihood. This indicates that although it gives an indication of how well the model fits the training data it does not give a good description of the system performance. The classification rate on the training data versus the validation data is seen to be linearly correlated.

The training performance seems to give a much better indication of the validation classification rate over the model log-likelihood. The model likelihood only takes into consideration how well the model matches the training data for that particular target. The training performance gives an indication of how well the system will classify data. The
drawback with using the training performance is that a model for each target is needed. A training performance classification rate cannot be obtained from a single model for a single target.

When further extrapolated to the test data, Figure 6-31, the validation data classification rate gives the best indication of test performance over either the model log-likelihood or the training performance. Again, in order to use a validation method, half of the data must not be used when training the models. The validation versus testing classification rate is linear but highly noisy. This indicates that the models with higher validation performance have a greater probability of having a high generalized classification rate on the test data, but does not guarantee it.

After all of the models for all of the targets are generated, the validation data is tested. The optimal model is selected by choosing the set of models that has the highest overall classification rate on the validation data. Since the process is computational efficient, it takes only a matter of minutes to do a brute force search through all possible combinations.
One hundred twenty models were trained in total for the 5 targets. The average model set got 422 of the 612 validation images correct. This corresponds to a classification rate of 69%. The selection procedure was run and the most optimal combination of target models got 477 images correct (78%); an increase in probability of detection of almost 10%. These 5 models were taken out of the mix and the second most optimal combination was obtained. This model performed at 75.8%; the procedure was repeated a third time for a model performance of 75.7%. Clearly, the selection process results in more optimal models compared to the average case.

To test the alternative selection criteria, the optimal sets of models were obtained using the four proposed optimization methods (MP, MP2, MP3, and MP4). The probabilistic quantities are reported as a function of the validation data set. Table 6-6 (a) and (b) report the probability of detection, the confidence probability, and the product of these two probabilities for MP and MP2. Table 6-6 (c) and (d) report the corresponding selection criteria probabilities for MP3 and MP4. Additionally the averages are given in (c) and (d) so a comparison can be made to the model selected versus the typical model.
As would be expected, the probability of detection of a single target in MP is greater than that in MP2. Notice in MP_T2, the probability of deciding T2 when the true state is T2 is 87%; the drawback is that 41% of the time when the classifier decides T2 the true class is something other than T2. In contrast MP2_T2 is more careful about false alarms and therefore when the classifier decides T2 there is a 75% chance that T2 is the true class, a significant improvement; there is only a slight reduction in the probability of correct classification.

After the models are selected they can be tested on the testing data. Table 6-7, 6-8, 6-9, and 6-10 report the classification statistics for each of the 4 sets of models selected.

Table 6-6: Type Caption Here

(a) Loud-mouth

| MP | P(c|w) | P(w|c) | Prod |
|----|-------|-------|------|
| T1 | 0.9231 | 0.6205 | 0.5744 |
| T2 | 0.8661 | 0.5851 | 0.5068 |
| T3 | 0.8296 | 0.7000 | 0.5807 |
| T4 | 0.8444 | 0.5561 | 0.4696 |
| T5 | 0.9516 | 0.5813 | 0.5532 |

(b) Single-target cautious

| MP2 | P(c|w) | P(w|c) | Prod |
|-----|-------|-------|------|
| T1  | 0.8681 | 0.8404 | 0.7296 |
| T2  | 0.8347 | 0.7465 | 0.6230 |
| T3  | 0.7482 | 0.9439 | 0.7062 |
| T4  | 0.7407 | 0.6849 | 0.5074 |
| T5  | 0.8952 | 0.8409 | 0.7572 |

(c) Global, single-target cautious

<table>
<thead>
<tr>
<th>MP3</th>
<th>Mixed</th>
<th>&lt;&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>T1</td>
<td>0.5427</td>
<td>0.3708</td>
</tr>
<tr>
<td>T2</td>
<td>0.4591</td>
<td>0.2912</td>
</tr>
<tr>
<td>T3</td>
<td>0.5386</td>
<td>0.3873</td>
</tr>
<tr>
<td>T4</td>
<td>0.3983</td>
<td>0.2399</td>
</tr>
<tr>
<td>T5</td>
<td>0.5528</td>
<td>0.3912</td>
</tr>
</tbody>
</table>

(d) Global, loud-mouth

<table>
<thead>
<tr>
<th>MP4</th>
<th>Mixed</th>
<th>&lt;&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>T1</td>
<td>0.6744</td>
<td>0.4958</td>
</tr>
<tr>
<td>T2</td>
<td>0.6423</td>
<td>0.4353</td>
</tr>
<tr>
<td>T3</td>
<td>0.6402</td>
<td>0.5070</td>
</tr>
<tr>
<td>T4</td>
<td>0.6195</td>
<td>0.3841</td>
</tr>
<tr>
<td>T5</td>
<td>0.7032</td>
<td>0.5709</td>
</tr>
</tbody>
</table>
Table 6-7: Loud-mouth models classification results

<table>
<thead>
<tr>
<th>Model</th>
<th>Target CR</th>
<th>Statistics</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>81%</td>
<td>52%</td>
</tr>
<tr>
<td>2</td>
<td>58%</td>
<td><strong>78%</strong></td>
</tr>
<tr>
<td>3</td>
<td>50%</td>
<td>58%</td>
</tr>
<tr>
<td>4</td>
<td>60%</td>
<td>49%</td>
</tr>
<tr>
<td>5</td>
<td>71%</td>
<td>51%</td>
</tr>
</tbody>
</table>

64.53%

Table 6-8: Single-target cautious models classification results

<table>
<thead>
<tr>
<th>Model</th>
<th>Target CR</th>
<th>Statistics</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>66%</td>
<td>60%</td>
</tr>
<tr>
<td>2</td>
<td>74%</td>
<td><strong>66%</strong></td>
</tr>
<tr>
<td>3</td>
<td>77%</td>
<td>68%</td>
</tr>
<tr>
<td>4</td>
<td>72%</td>
<td>59%</td>
</tr>
<tr>
<td>5</td>
<td>64%</td>
<td>60%</td>
</tr>
</tbody>
</table>

67.07%

Table 6-9: Global, single-target cautious classification results

<table>
<thead>
<tr>
<th>Model</th>
<th>Target CR</th>
<th>Statistics</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>69%</td>
<td>69%</td>
</tr>
<tr>
<td>2</td>
<td>74%</td>
<td><strong>66%</strong></td>
</tr>
<tr>
<td>3</td>
<td>68%</td>
<td>68%</td>
</tr>
<tr>
<td>4</td>
<td>69%</td>
<td>61%</td>
</tr>
<tr>
<td>5</td>
<td>64%</td>
<td>59%</td>
</tr>
</tbody>
</table>

68.07%
In MP, the bias towards a particular class is obvious; an increase in the classification rate of approximately 16% is observed for a particular target compared to that model’s overall classification rate. The cost of this increase is poor performance on the remaining targets. In contrast, MP2 tries balances probability of detection with increased confidence. As a result, the probability of detection for the single target is poor, but the probability of detection of the remaining targets is increased.

To increase the overall probability of detection further, the global probability of detection term was included in MP3. The model set is seen to have a slight overall increase in the probability of detection. In MP4 the single target classification rates are seen to jump up again as the confidence term is removed. The cost is a reduction of the overall classification percentages.

Finally, each of the 4 selection criteria was merged using the average likelihood method. The results are reported in Table 6-11 along with the case of 3 HMMS using the global classification rate. Although the four selection criteria have varied performance as individual classifiers, as ensembles all four have approximately the same performance. Moreover, this performance is equivalent to the performance of the 3 HMM case.

Table 6-10: Global, single-target loud-mouth

<table>
<thead>
<tr>
<th>Model</th>
<th>Target CR</th>
<th>Statistics</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>79%</td>
<td>71%</td>
</tr>
<tr>
<td>2</td>
<td>67%</td>
<td>76%</td>
</tr>
<tr>
<td>3</td>
<td>57%</td>
<td>65%</td>
</tr>
<tr>
<td>4</td>
<td>62%</td>
<td>48%</td>
</tr>
<tr>
<td>5</td>
<td>63%</td>
<td>67%</td>
</tr>
</tbody>
</table>

66.52%
In conclusion, it does not matter how it is obtained, but diversity in the ensemble’s models is important. With diversity, the ensembles perform significantly better than the individual classifiers. The ensembles also overcome the fact that the best generalized models might not be selected out of all possible combinations.

For this implementation, the 3 HMM ensemble is used with global classification rate model selection. However, a different optimization function may be more appropriate for another application. For example if the target type and number of targets varied drastically, the loud-mouth classification structure would be ideal. With this structure the optimal models can be selected as a function of the validation data of a single target; the models are not selected globally. Similarly, the other criteria may be more appropriate for other mission definitions.

6.4.3 Classification Results

6.4.3.1 Data

DARPA’s MSTAR dataset is again used for testing on the 5 target system. Before classification can be performed, shadow segmentation is needed. The automated
segmentation algorithm with smoothing of Section 3.3.3 was applied. Unfortunately, the segmentation algorithm was unable to obtain useable shadow segmentations for all of the testing and training images. Each of the segmentations was manually inspected. For the training data, the unusable segmentations were simply discarded. Both the raw and reduced test datasets were used to evaluate the classification algorithms. The numbers of images available before and after trimming the database are reported in Table 3-3.

Table 6-12: Database trim statistics

<table>
<thead>
<tr>
<th>Target</th>
<th>Depression Angle</th>
<th>15°</th>
<th>17°</th>
<th>15°</th>
<th>17°</th>
<th>15°</th>
<th>17°</th>
<th>15°</th>
<th>17°</th>
<th>15°</th>
<th>17°</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total number of SAR chips</td>
<td>15°</td>
<td>195</td>
<td>256</td>
<td>274</td>
<td>298</td>
<td>274</td>
<td>299</td>
<td>274</td>
<td>299</td>
<td>274</td>
<td>299</td>
</tr>
<tr>
<td>Non-valid segmentations</td>
<td>15°</td>
<td>13</td>
<td>1</td>
<td>19</td>
<td>29</td>
<td>3</td>
<td>5</td>
<td>4</td>
<td>10</td>
<td>25</td>
<td>14</td>
</tr>
<tr>
<td>Percentage discarded</td>
<td>15°</td>
<td>6.7%</td>
<td>0.4%</td>
<td>6.9%</td>
<td>9.7%</td>
<td>1.1%</td>
<td>1.7%</td>
<td>1.5%</td>
<td>3.3%</td>
<td>9.1%</td>
<td>4.7%</td>
</tr>
</tbody>
</table>

The reduced dataset at 15° depression is split in half. Ordered in terms of the aspect angle, samples are alternatively used in the training and validation datasets. The 17° depression data is used for testing.

6.4.3.2 Single-look Classification

After obtaining all the necessary design parameters, the system tests aimed to evaluate the classification performance as a function of the different target types and look angles; Figure 6-32 illustrates the classification results. For each test class, the number of test images is histogramed based on the look angle. The bin centers range from 0° to 360° in 10° increments. The number of test instances classified as each of the 5 targets is displayed according to the target color scale given in Figure 6-33.
From the graphs, it is clear that the performance is based on the look angle. For example, the T72 is almost always confused as BRPM from 150° to 210°. Figure 6-34 shows the overall classification rate for the T72. The performance is above 75% for...
angles not in either of the following ranges: $-45^\circ$ to $45^\circ$ and $150^\circ$ to $210^\circ$. This means that the T72 is not easily identified from its shadow profile when looking directly at either the front or the back. All images were at $0^\circ$ articulation.

Figure 6-34: Classification rate for T72

The plot shows the classification performance on T72 test images. It is clear that the target is easily identified when looking at either of the tank’s sides. When imaged from the front or back, the shadow does not contain enough discriminatory information for this target to be identified reliably.

Conversely, it is seen that the D7 is best identified in the ranges that the T72 has the worst performance. This is because the D7 has a unique profile from the front and back. This would not necessarily be the case if the shadow was not forced to be solid; in some aspects the D7 profile has a hole that cannot be accounted for using a chain coding system. The chain code system assumes that the target shadow is a solid 2D mass. As a result the coding scheme limits the performance for a small subset of images in this particular database.
6.4.3.3 Multi-look Classification

To test the performance under a multi-look scenario, two sets of tests were used: one with the full test data set; and one with the reduced test data set. Using the full data set gives an indication of how the system performs from end-to-end. The reduced data set eliminates the effects of bad segmentations and gives a better indication of how the classifiers specifically perform.

The images used in multi-look testing were uniformly, randomly sampled from all possible looks for a given target. The performances from the multi-look tests are reported in Figure 6-35, and Tables 6-13 (a) and (b). To obtain estimates for the untested datasets an exponential curve was fit.

![Graph showing classification rate of multi-look scenarios](image)

**Figure 6-35**: Classification rate of multi-look scenarios

The average classification rate for the 5 targets is reported for multi-look systems of varying sizes. Adding additional looks into the classification scheme is seen to have diminishing returns. The system, however, generally performs well for multiple looks. Specifically, the classification system performs over 90% at only 3 looks.
Clearly, the reduced dataset outperforms the full dataset for a single look. The improvement is approximately equivalent to the percentage of poor segmentations in the testing dataset, 3.96%; this means that most of the poorly segmented images are misclassified. The classification scheme has performance greater than 90% at 3 looks and the full system has performance greater than 90% at 5 looks. Both systems appear to have asymptotic behavior. The classification scheme reaches a predicted saturation at approximately 96.75% and the full system at a 91% classification rate.

Figure 6-36 shows the performance for a 2 look system, and reports the performance as a function of the angle between the two looks. The data shows that the optimal configuration of a 2 radar system has an intra-radar angle of approximately 90°. This indicates that systems looking at orthogonal faces will allow for the largest probability of correct classification.
6.4.4 Integration

The final simulations test the plausibility of integrating the shadow-only classification output with a traditional scattering center type classifier. For the highlight classification system, the entire 15° data is used for the templates and the 17° is again used for testing. The confusion matrices for each classifier are given in Figure 6-37.

Figure 6-36: Performance of a 2-look system as a function of configuration

For a system of 2 looks the overall classification rate is reported as a function of the angle between the radars. The configuration in (b) shows the optimal and suboptimal positions for radar 2. To simulate various combinations a Monte-Carlo simulation was used; the look angles of radar 1 and radar 2 were uniformly selected. The optimal angle is slightly under 90°.
As expected, the overall system performance of the scattering center classifier, 89.45%, is greater than that of the shadow-only classifier, 71.95%. Notice however that the two outputs compliment each other. For actual cases of class 1, the shadow-only classifier does not often mistake the test instance for class 2. The highlight classifier however has its worst performance for this same scenario. Clearly, there is information output from the shadow classification system that can be usefully integrated with the highlight classification system.

More generally, it is observed that probabilities of obtaining a correct classification from the two systems are in fact independent. Table 6-14 shows the joint probability space of both classifiers. From Table 6-14, $P(S \cap H) = 0.6464$ while $P(S) \times P(H) = (0.6464 + 0.0731) \times (0.6464 + 0.2481) = 0.6436$. 

---

**Figure 6-37: Confusion matrices**

Figure (a) shows the confusion matrix for the shadow-only classification system, and (b) shows the confusion matrix for the classifier operating only on the highlight region. Although the shadow-only system has a slightly lower overall classification rate, the performance of the two classifiers seems to be complimentary.
Since \( P(S \cap H) \approx P(S) \times P(H) \), it follows from Eq. 6.28 that these events are independent.

### Table 6-14: Joint probability space of shadow-only and highlight classification

<table>
<thead>
<tr>
<th></th>
<th>H</th>
<th>( \bar{H} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
<td>64.64%</td>
<td>7.31%</td>
</tr>
<tr>
<td>( \bar{S} )</td>
<td>24.81%</td>
<td>3.24%</td>
</tr>
</tbody>
</table>

### 6.5 Shadow Projections

#### 6.5.1 Introduction

It has been shown that in a narrow range of depression angles, the shadow within a SAR image can be reliably used for classification. To make this technique more operationally robust it is necessary to extend this idea to a larger range of depression angles. When the depression angle is shallow, the targets’ shadows will be long and exaggerated. Conversely, the shadows will be compressed for steep depression angles. Therefore, it is desirable to have a projection that can estimate the shape of a shadow at a depression angle different from the one it was obtained.

A homographic transformation is the most general of all the planar mapping functions. The transformation allows for changes in scale, rotation, skew, and perspective [70]. Figure 6-38 shows a pictorial example of a projective transformation. An application of this mapping should allow for the shadows in SAR images to be transformed from one depression space to another. This technique is only capable of
changing the overall shape of the shadow, not the structure. If a particular feature exists in only a small range of angles, a planar mapping will be unable to capture these features if they are unavailable in the initial image.

Using homogenous coordinates, the transformation of a single point in two-space can be expressed by the homographic matrix given in Eq. 6.30.

\[
\begin{bmatrix}
  x' \\
  y' \\
  1
\end{bmatrix}
= \begin{bmatrix}
  h_{11} & h_{12} & h_{13} \\
  h_{21} & h_{22} & h_{23} \\
  h_{31} & h_{32} & h_{33}
\end{bmatrix}
\begin{bmatrix}
  x \\
  y \\
  1
\end{bmatrix}
\]

[6.30]

After rearranging the terms, the new point can therefore be expressed by Eq. 6.31 and Eq. 6.32

\[
x' = \frac{h_{11}x + h_{12}y + h_{13}}{h_{31}x + h_{32}y + h_{33}}
\]

[6.31]

\[
y' = \frac{h_{21}x + h_{22}y + h_{23}}{h_{31}x + h_{32}y + h_{33}}
\]

[6.32]
There are 9 total unknowns in the above equations. However, the multiplication of a constant $k$ would not change the overall equations. Therefore there are only 8 degrees of freedom. In order to enforce this rule one of the parameters is set to unity $h_{33} = 1$. By imposing this constraint, multiplying through the denominator in Eq. 6.31 and Eq. 6.32 and rearranging the terms, Eq. 6.33 and Eq. 6.34 are obtained.

$$x' = h_{11}x + h_{12}y + h_{13} - h_{31}xx' - h_{32}yy'$$ \[6.33\]

$$y' = h_{21}x + h_{22}y + h_{23} - h_{31}xy' - h_{32}yy'$$ \[6.34\]

In order to solve the equations, a minimum of 4 point sets are needed. Additional points, however, can increase the accuracy of the estimated transform. Using 4 sets of point correspondences, the equations above can be expressed in matrix form, Eq. 6.35.

$$\begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -x_1x'_1 & -y_1x'_1 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -x_1y'_1 & -y_1y'_1 \\ x_2 & y_2 & 1 & 0 & 0 & 0 & -x_2x'_2 & -y_2x'_2 \\ 0 & 0 & 0 & x_2 & y_2 & 1 & -x_2y'_2 & -y_2y'_2 \\ x_3 & y_3 & 1 & 0 & 0 & 0 & -x_3x'_3 & -y_3x'_3 \\ 0 & 0 & 0 & x_3 & y_3 & 1 & -x_3y'_3 & -y_3y'_3 \\ x_4 & y_4 & 1 & 0 & 0 & 0 & -x_4x'_4 & -y_4x'_4 \\ 0 & 0 & 0 & x_4 & y_4 & 1 & -x_4y'_4 & -y_4y'_4 \end{bmatrix} \begin{bmatrix} h_{11} \\ h_{12} \\ h_{13} \\ h_{21} \\ h_{22} \\ h_{23} \\ h_{31} \\ h_{32} \end{bmatrix} = \begin{bmatrix} x'_1 \\ y'_1 \\ x'_2 \\ y'_2 \\ x'_3 \\ y'_3 \\ x'_4 \\ y'_4 \end{bmatrix}$$ \[6.35\]

If more then four sets of point correspondences are available for use in Eq. 6.35 the problem is ill-posed, but can be solved using planar least squares. Eq. 6.35 can be represented as a set of linear equations Eq. 6.36.

$$Ah = b$$ \[6.36\]
Where $A$ is $2N \times 8$, $h$ is $8 \times 1$, and $b$ is $2N \times 1$ for $N$ total points. The coefficients can therefore be solved using the method of least squares inverse, Eq. 6.37.

$$
A^T h = A^T b
$$

$$
(A^T A) h = A^T b
$$

$$
h = (A^T A)^{-1} A^T b
$$

[6.37]

Again, a minimum of 4 point sets are required to solve for all of the unknowns, but $N$ points can increase the accuracy of the method. Also, the constraint on $h_{33}$ could cause errors if the actual value were 0.

The technique outlined in Section 6.5.2 uses sets of point correspondences between two shadow shapes to estimate the appropriate homographic transformation. The robustness of this point matching approach is tested on canonical datasets. An overlap ratio factor is implemented to indicate the accuracy of the estimated target shadow [39]. The value is bounded between [0 1] and is given in Eq. 6.38.

$$
\Omega = \frac{A_D}{A_D + A_{FA} + A_{MD}}
$$

$A_D$ = Area of correct detection

$A_{FA}$ = Area of false alarm

$A_{MD}$ = Area of missed detection

[6.38]

Alternatively the overlap factor can be defined by Eq. 6.39

$$
\Omega = \frac{A_D}{A_{Target} + A_{Estimate} - A_D}
$$

$A_{Target}$ = Total area of target

$A_{Estimate}$ = Total area of estimate

[6.39]

For an $N \times M$ binary target image $T(i,j)$ and its estimation $\tilde{T}(i,j)$, the overlap ratio factor is given by Eq. 6.40.
Finally, in Section 6.5.3 a projective transformation is derived based on the target radar geometry. The derivation is tested by projecting the shadows obtained from real SAR imagery. This technique was also used Section 6.3.3.3 to test the classification method’s performance over its range of depression angles.

### 6.5.2 Point correspondences

#### 6.5.2.1 Implementation

The first method that is being tested for the projection of shadow shapes uses homographic transformations estimated directly from sets of point correspondences. Figure 6-39 illustrates a zoomed version of a shadow and emphasizes the edge and object points. The distinction between these two point sets is important in the subsequent development of the technique.
Clearly, \((x^i_{nm}, y^i_{nm})\) ∈ \((x^i_{nm}, y^i_{nm})\), the pixels along the edge of the initial object, are contained within the set of all object points. Since the initial shadow is guaranteed to be a solid rigid body (as required by the shadow model) the edge points will completely define the shadow. As such, these edge points are matched between two shadows and are subsequently used to estimate the homography. Shadow extraction and edge point selection are the first two steps of the overall transformation process as outlined in Figure 6-40.

Figure 6-39: Shadow object composition
The shadow is shown in red, and the background in blue. The shadow is made up of a set of discretized object points. The edge points completely define the shadow and are illustrated with green circles.
The homography is estimated using the **RANdom SAmple Consensus** (RANSAC) tool and the least squares homography estimator developed in Eq. 6.35 [71]. RANSAC is a tool that allows the estimation of a given set of parameters from $N$ points out of a total sample set $M$; where the total sample set is known to have outliers. The algorithm attempts to avoid the outliers in the given set of $M$ points, and compute the transformation on the best estimation of $N$ inliers. The flow of the algorithm is given as:

- Randomly take $P$ point correspondences from the entire sample set
- Calculate a homography based on these $P$ point sets
- Apply the homography and count the number of projected edge points, $C$, that are less than distance $d$ away from the edge point
- Repeat the above steps $N$ times
- Take the homography with highest number of inlier points, and use all $C$ inliers to calculate the final homography estimate

Figure 6-40: Shadow projection

The flow chart shows how a homographic transformation is estimated from two sets of shadow edge points. RANSAC is used in an attempt to eliminate bad sets of point correspondences. Once a suitable transformation is obtained, it can be used to transform either shadow into the depression space of the other image.
For estimation of a homographic transformation, at least 4 estimation point sets are needed. Eq. 6.41 and its ensuing derivation outline the number of runs required for RANSAC to have a desired level of performance.

\[
N = \frac{\log(1 - p)}{\log(1 - (1 - o)^s)}
\]

[6.41]

\(o\) = Probability that a single point is an outlier  
\(s\) = Number of points in a sample  
\(N\) = Number of entire samples performed (unknown parameter)  
\(p\) = Desired probability for getting a good sample

The probability that choosing one point is an inlier: \(1 - o\)

The probability of choosing \(s\) inliers in a row: \((1 - o)^s\)

The probability that 1 or more of the points were outliers: \(1 - (1 - o)^s\)

The probability that \(N\) total samples were not outlier free: \(\left(1 - (1 - o)^s\right)^N\)

The probability that at least one sample was free of outliers: \(1 - \left(1 - (1 - o)^s\right)^N\)

Finally, \(N\) can be solved for in the equation below:

\[1 - \left(1 - (1 - o)^s\right)^N = p\]

For example, if it is assumed that one of three possible correspondences for each point is correct, the probability that a single point is an outlier is 66%. Although only 4 point sets are required to calculate a homography, a least squares solution using 8 point sets seems to provide a more robust estimate. The desired probability of getting a good sample is 99%. Using sets of these parameters the number of samples required are: \(\{o = .66, s = 4, p = .99\} \therefore N = 343\) and \(\{o = .66, s = 8, p = .99\} \therefore N = 25,786\).

By changing the number of points used to calculate the homography, the total number of samples needed is drastically increased. However it will be illustrated that the least square estimate more accurately predicts the overall transformation when 8 point sets are used. After RANSAC is complete, a new homography is calculated using the
estimated inliers only when the point correspondences of the estimated inliers is known; otherwise the best known homography is used directly.

Since the homography was estimated on the edge points, one would reason that the projective transformation should then be estimated on the respective edge points only. However, the application of the projective transform to all object points is valid because the edge points completely categorize the initial shadow shape. Morphology can also be applied to smooth the final shape estimates.

Ideally, the two shadow point sets will have a 1-to-1 correspondence. In practice this correlation may not be known. Therefore each point in the original point set can be mapped to multiple points in the second point set as seen in Figure 6-41. Through RANSAC, the most optimal sets of points will be chosen. The drawback of using multiple correspondences is an increase in the number of necessary RANSAC iterations.

![Figure 6-41: Illustration of 3-to-1 point correspondence](image)

Since the direct point correspondences between two points sets may not be know, each point in one dataset is mapped to multiple points in the other.

One of the important tests of this method is to determine if a homography at one aspect can be applied to the same target at another aspect. The setup for testing the homographies over multiple aspect angles is given in Figure 6-42. A homography is
calculated for every aspect angle. Each of these homographies is then evaluated using
the overlap factor over all aspect angles.

6.5.2.2 Results

To test the transformation process, the 2.5-D canonical shadow model was used. Figure 6-43 shows two shadows obtained from the same aspect angle, but different
depression angles. The shadows are more then just scaled or rotated versions of each
other. The proportions in different parts of the object are substantially different. They
are related by a projective transform.
In order to morph one shape into the other three techniques were used. First points along the boarder were handpicked. Next, all 128 edge points were used to calculate a homography. Finally, RANSAC was used to identify a set of inliers; a homography was calculated as a function of the inliers. Figure 6-44 shows the set of all point correspondences and Figure 6-45 shows the resulting transformations.

![Image](image1.png)

(a) Simulated depression angle of 80°  
(b) Simulated depression angle of 85°

Figure 6-43: Shadows of varying depression angles

The effects of varying depression angle are quite clear. The shadow at a shallower depression angle appears to be stretched in comparison.

![Image](image2.png)

(a) Hand-picked  
(b) All possible  
(c) RANSAC inliers

Figure 6-44: Point correspondences

The figure illustrates three different methods for calculating a homography. The homography is based on various sets of point correspondences.
The calculated homography, using the handpicked, points is a good representation of the transformation from one shadow to the other. Using all possible points, the shapes match, but not quite as well as the hand-picked points. Not all of the points in the correspondence point set may lead to a meaningful homography; some of the point pairs are outliers. RANSAC can be applied in an attempt to identify a more optimal transformation. Since the shapes were simpler, RANSAC eliminated only a small number of outliers and had similar performance to using all possible points. As the shapes become more complex the disparity between the techniques increases.

In addition to testing the algorithm with the same 1-1 correspondences, 3-1 point correspondences were also used. It is known that the 1-1 points do not directly match for
all sections of the image. So in addition to the 1-1 matched, each point in the source image is mapped to 3 points in the reference image. This allows for shifts of correspondences along the boundary of the image. The total number of points to choose from in the RANSAC algorithm is increased by a factor of 3. The percentage of outliers is at least 66% since each point in the reference image only matches to 1 point in the source image. Table 6-15 outlines the performance under various parameter configurations of RANSAC for two distinct aspect angles.

Table 6-15: RANSAC performance

<table>
<thead>
<tr>
<th>Aspect Angle</th>
<th>DΦ 1 map from DΦ 2</th>
<th>DΦ 2 map from DΦ 1</th>
<th>DΦ 1 map from DΦ 2</th>
<th>DΦ 2 map from DΦ 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hand Picked</td>
<td>0.8851</td>
<td>0.8230</td>
<td>0.8851</td>
<td>0.8230</td>
</tr>
<tr>
<td>RANSAC</td>
<td>0.8577</td>
<td>0.8862</td>
<td>0.8455</td>
<td>0.8451</td>
</tr>
<tr>
<td>(1-1 / vector eval) s=4</td>
<td>RANSAC</td>
<td>(1-1 / boundary) s=8</td>
<td>(1-1 point corr.) s=8</td>
<td>(1-1 point corr.) s=8</td>
</tr>
<tr>
<td>RANSAC</td>
<td>0.8711</td>
<td>0.8421</td>
<td>0.8797</td>
<td>0.8451</td>
</tr>
<tr>
<td>(3-1 point corr.) s=8</td>
<td>RANSAC</td>
<td>(3-1 point corr.) s=8</td>
<td>(3-1 point corr.) s=8</td>
<td>(3-1 point corr.) s=8</td>
</tr>
<tr>
<td>RANSAC</td>
<td>0.9005</td>
<td>0.7423</td>
<td>0.8000</td>
<td>0.7936</td>
</tr>
<tr>
<td>(1-1 / vector eval) s=4</td>
<td>RANSAC</td>
<td>(1-1 / boundary) s=8</td>
<td>(1-1 point corr.) s=8</td>
<td>(1-1 point corr.) s=8</td>
</tr>
<tr>
<td>RANSAC</td>
<td>0.9002</td>
<td>0.7181</td>
<td>0.7868</td>
<td>0.8078</td>
</tr>
<tr>
<td>(3-1 point corr.) s=8</td>
<td>RANSAC</td>
<td>(3-1 point corr.) s=8</td>
<td>(3-1 point corr.) s=8</td>
<td>(3-1 point corr.) s=8</td>
</tr>
</tbody>
</table>

In order of performance, the methods ranked – Hand Picked, 3-1 / s=8, 1-1 / s=8, 3-1 / s=4, 1-1 / s=4. Clearly the use of 8 point sets improves the estimate over the selection of only 4. Also, mapping the point correspondences 3-to-1 increased the performance.

Using the optimal RANSAC parameters as calculated above, the aspect dependence of the homography were evaluated. A simple object with symmetric properties was tested; the object is shown in Figure 6-46.
As described in Figure 6-42, a grid was obtained to evaluate the overlap error metric over all possible combinations of homographic calculations and test angles. Figure 6-47 illustrates the results for test object 1. The test angles varied from 0° to 345° in 15° increments.

The majority of the iterations yielded an overlap factor greater than .7. This means that the homography applied to a registered image was a good approximation regardless of which aspect angle it was calculated at. The projected test shadow is most like the actual shadow when the homography is applied at 0° and ±90° increments. This
result is pleasing as it illustrates the symmetry of the object, but does not restrict the use of a given homography to a symmetrical object.

The same analysis was repeated for a simple tank-like object with single axis symmetry. The results are given in Figure 6-48.

There are certain aspect angles where the homographies do not seem to apply well. This is due to the fact that the shadows at certain aspect angles simply do not have the same shape. The median analysis shows that the transformation calculated at a particular aspect is the least applicable at ±90°. This result is indicative of the object’s single axis symmetry.

A final simulation set was derived where the object had no symmetry and hidden features. Hidden features mean the object has a discernibly different shape at the two depression angles. Figure 6-49 shows the test object and some sample shadow projections.
When new structures arise, the algorithm fails to correctly project the shadows. A reason for the failure is the lack of good point correspondences between the shapes. Since one shadow has a significantly different shape, the points no longer lie in the same regions of the two images. In order to use the point-based transformation method reliably across a range of target types, a more advanced point matching technique needs to be applied. However, under certain constraints, the point-based transformation method showed good potential for performing the necessary transformation.

6.5.3 Radar geometry

The second approach for obtaining shadow projection is a derivation of the necessary transformation matrix using the target-radar geometry. A geometric optics
interpretation of the radar shadow in range is given in Figure 6-50. Multiple radars are assumed to have constant ground range.

![Shadow geometry diagram](image)

**Figure 6-50**: Shadow geometry

The above shadow geometry is the starting point for the shadow projection transformation derivation. Multiple radars are assumed to have varying depression, but fixed ground range.

Using the geometry above, the radar geometry can be related to the shadow geometry by using similar triangles, Eq. 6.42.

\[
\frac{T}{S} = \frac{H - T}{R} \tag{6.42}
\]

Therefore the length of the shadow can be written as Eq. 6.43, where the height of the radar can be given in terms of the radar range and depression angle Eq. 6.44.

\[
S = \frac{R \times T}{H - T} \tag{6.43}
\]

\[
H = R \tan \theta_D \tag{6.44}
\]

A configuration of two radars at different depression angles will have shadows of variable lengths. Figure 6-51 below shows the associated geometry.
The length of the second shadow can be defined in terms of the first shadow and the shadow length multiplier, \( p \), Eq. 6.45.

\[
S_2 = pS_1
\]  \[6.45\]

Using the geometric relations derived above, the multiplier can be given as Eq. 6.46.

\[
p = \frac{S_2}{S_1} = \frac{R \times T}{R \tan \theta_D - T} = \frac{R \tan \theta_D - T}{R \tan \theta_2 - T}
\]  \[6.46\]

In most realistic imaging scenarios the first term in the numerator and denominator is going to be much greater than the target height, \( R \tan \theta_D >> T \). Additionally, in this example the ground range is kept constant. Therefore the final expression for the variable shadow lengths is given by Eq. 6.47.

\[
p \approx \frac{\tan \theta_D}{\tan \theta_2}
\]  \[6.47\]
This expression confirms the observation that as the depression angle is decreased the overall extent of the shadow downrange is increased. Similarly, as the depression angle is increased the shadow will almost vanish.

Next, the shadow cross-range perturbations can be calculated. The target has variable heights and thus the cross-range shadow is a realization of this profile. However, to simplify the derivation, the shadow is defined by its bounding box; a bounding box is the smallest possible rectangle that encloses the region. The geometry for the two depression angles is given in Figure 6-52.

![Cross-range geometry](image)

Figure 6-52: Cross-range geometry

The cross-range geometry is given in terms of the bounding box of the respective shadow regions. Using this geometry, the cross range extent can be given for the two depression angles as Eq. 6.48.

\[
\tan \theta = \frac{S'_{w1}}{S_1 + R} = \frac{S'_{w2}}{S_2 + R}, S' = \frac{S}{2}
\]  

[6.48]
The shadow at depression angle 2 can therefore be expressed in terms of the observed shadow at depression angle 1 according to Eq. 6.49.

\[
S'_{w2} = S'_{w1} \frac{S_2 + R}{S_1 + R} = S'_{w1} \frac{pS_1 + R}{S_1 + R}
\]  \hspace{1cm} [6.49]

Given a shadow bounding box under a particular depression angle, a new shadow can be projected into its respective bounding box for a new depression angle, Eq. 6.50 and Eq. 6.51.

\[
BB_1 = \{(−S'_{w1},0),(−S'_{w1},S_1),(S'_{w1},0),(S'_{w1},0)\}
\]  \hspace{1cm} [6.50]

\[
BB_2 = \{(−S'_{w1},0),(−S'_{w2},S_2),(S'_{w2},0),(S'_{w1},0)\}
\]  \hspace{1cm} [6.51]

The coordinates of the second bounding box are calculated via Eq. 6.45 and Eq. 6.49. Given these two sets of four points, the projective transformation can be calculated. The projection of any point is given by Eq. 6.30. Forcing \( h_{33} = 1 \), the homography can be solved as Eq. 6.52.

\[
\begin{array}{cccccccc}
- S'_{w1} & 0 & 1 & 0 & 0 & 0 & S'_{w1}S' & 0 \\
0 & 0 & 0 & - S'_{w1} & 0 & 1 & 0 & 0 \\
- S'_{w1} & S_1 & 1 & 0 & 0 & 0 & - S'_{w1}S'_{w2} & S_1S'_{w2} \\
0 & 0 & 0 & - S'_{w1} & S_1 & 1 & S'_{w1}S_2 & - S_1S_2 \\
S'_{w1} & S_1 & 1 & 0 & 0 & 0 & - S'_{w1}S'_{w2} & - S_1S'_{w2} \\
0 & 0 & 0 & S'_{w1} & S_1 & 1 & - S'_{w1}S_2 & - S_1S_2 \\
S'_{w1} & 0 & 1 & 0 & 0 & 0 & - S'_{w1}S' & 0 \\
0 & 0 & 0 & S'_{w1} & 0 & 1 & 0 & 0 \\
\end{array}
\begin{bmatrix}
  h_{11} \\
  h_{12} \\
  h_{13} \\
  h_{21} \\
  h_{22} \\
  h_{23} \\
  h_{31} \\
  h_{32} \\
\end{bmatrix}
= \begin{bmatrix}
- S'_{w1} \\
0 \\
0 \\
S_2 \\
S'_{w2} \\
S_2 \\
S'_{w1} \\
0 \\
\end{bmatrix}
\]  \hspace{1cm} [6.52]

Alternatively, the equations could be solved by forcing \( ||h|| = 1 \). Finally, the image (at the initial depression angle) is transformed by the calculated homography to obtain an estimate of the shadow at the alternative depression angle. All of the quantities
of Eq. 6.52 can be calculated as a function of the initial shadow’s bounding box. The shadow of Figure 6-53 is used as the reference to project into other depressions, Figure 6-54. Although there are some slight differences, the technique does a good job in capturing the overall essence of the shadow in an alternate radar-target geometry.

Figure 6-53: Reference SAR shadow at 15° depression
A shadow at 15° was used as the base reference and projected in alternative depression spaces.

Figure 6-54: Projection of 15° shadow to varied depressions
The 15° shadow was projected in 17° and 19° depression. There are some minor differences in the leading edge of the shape, but overall the technique works well. Some of the variations are due to the fact that the actual shadows were collected in a real environment. Even at the same depression angle and aspect, multiple chips of the same target will exhibit variations. The changes are due to clutter, noise, and the segmentation procedure. Overall, the derivation is seen to hold for radars of constant ground range.
6.6 Conclusions

SAR images are effectively being used for NCTR. Currently, the highlight information in the image is the main focus when designing target classification structures. A recent trend however has been the inclusion of the shadow information since it contains a projection of the target’s profile. This work has shown that in both single and multi-look scenarios the shadow-only classifiers can provide robust and reliable identification. Moreover, the technique developed was shown to have complimentary performance which allows for optimal integration with existing highlight techniques.

First, Procrustes analysis was investigated and proved to be a useful tool for the examination of shadow regions in T72 and BTR60 SAR images. After alignment and calculation of the mean shape, the shapes were projected into their respective shape space. PCA was performed for dimensionality reduction. Once the shape descriptors were obtained, the classification of the targets was performed. Additionally, a method of hidden Markov modeling was used in conjunction with a chain coding technique to classify the targets.

From the results, the performance of the Procrustes mean classifier on the shadow region is seen to be comparable to random guessing; there is not enough discriminatory information in only the average shape to classify the targets. When the variance was taken into account the Gaussian classifier performed better, but its performance was still not sufficient. The KNN and MLP classifiers both worked satisfactorily. The highest classification rate was obtained using the HMM classifier. Compared to other techniques, this method showed the most promise.
Further testing was performed to determine the HMM classifier’s robustness to scale and depression angle. Under fixed training conditions, the algorithm can tolerate slight changes in resolution and depression angle. For large changes however, the trained models no longer sufficiently represent the targets. If the models are trained under the same conditions as the test images, high performance can again be obtained. This indicates that a bank of models could be trained such that the operational parameters of the test image could be matched.

To improve the performance when applied to large scale problems, an ensemble HMM system was developed. A novel approach for model selection was developed along with a variety of search optimization choices. The performance of the system can be tweaked based on the mission objects. For example, the probability of false alarm for a particular target can be minimized while the probability of detection of another target is simultaneously maximized.

In general, the shadow classification algorithms allow for the development of a system that exploits the shadow information in a SAR image for ATR. For a single look, the shadow-only classifier ensemble had a 74% classification rate, but with as few as 3 looks the average performance was greater than 90%. For only 2 looks, the classification rate also reached over 90% for certain radar configurations. These results indicate that reliable classification based on shadow features in high resolution SAR images is practical. Moreover, the classification output is independent from the output of a classifier operating on the scattering centers. This indicates that the shadow methods can be used to enhance the more traditional methods of ATR using SAR images.
Finally to make the techniques more robust, shadow projection methodologies were investigated. Point-based shadow projection techniques are limited in their effectiveness by the accuracy in which point sets from two images can be matched. When good correspondences between points can be obtained, the RANSAC least squares algorithm was able to output a homography capable of projecting the shadows between depression angles. The applicability of this transformation was limited in aspect by the nature of the target features. As an alternate approach, a transformation was derived as a function of the target-radar geometry. The projection is calculated as a function of a shadow’s bounding box in one depression angle and the desired new depression angle. The technique produced the desired effects when applied to the MSTAR data.

The application of these shadow techniques to future NCTR applications shows great promise. By combining these classification algorithms with traditional scattering center classifiers, reliable target recognition with SAR images can be performed at further ranges and in scenarios with lower signal-to-noise ratios. Additionally, through the use of the secondary processes (model selection and shadow projection) these techniques can be adapted to work with a wide variety of mission objectives.
7.1 Further Research / Investigations

The tools developed in this research focused mainly on airborne radar systems looking at ground and air based targets. As such, the implementation of these techniques in unmanned aircraft is a natural extension of this work. Section 7.1.1 outlines future research that integrates this work with higher level mission controls – navigation and scheduling of sensing tasks. Alternatively, these techniques can be ported to other arenas such as maritime surveillance. The implications of this move and additional research are discussed in Section 7.1.2.

7.1.1 Applications in Unmanned Aerial Vehicles

Unmanned aerial vehicles (UAVs) are growing in popularity as a sensing vehicle in many combat situations. UAVs can be controlled from a remote location or they can fly autonomously with specific mission objectives [72]. The research presented in this dissertation can easily be implemented and expanded to aid in classification or navigation using a UAV.

For example it was determined that, without any a priori knowledge, highlight fusion is optimal for looks that are spaced 180° in aspect and shadow classification fusion is optimal for looks that are spaced 90° apart. As such, a UAV routing algorithm can
attempt to image targets such that these two criteria are met. Figure 7-1 is an illustration of a scene containing multiple targets. Two UAVs are routed such that each target is imaged at the most optimal sensing locations while avoiding the taboo regions within the flight space. In the example pictured, each target is imaged at multiple locations. Some targets are imaged at 180° spacing with two looks while others have three looks at 90° and 180° spacing to the original image.

![Directed UAV flight paths and sensing locations](image)

Figure 7-1: Directed UAV flight paths and sensing locations

In the illustration above two UAVs are routed such their collective flight paths are optimized. The flight path is governed by taboo regions that must be avoided, and targets that require sensing. Due to operational constraints it is not feasible to continually sense a target. Therefore, multiple sensing locations are selected for each target to optimize the respective fusion. The images obtained from the sensing locations could be subsequently input into either the highlight fusion or shadow classification algorithms.

A dynamic routing technique could also be applied. A classification system with confidence levels could be developed based on highlight processing and the shadow classification method. After a UAV initially images a target, it would attempt to classify the target. When the confidence level is high, the target ID would be used to schedule the
UAV’s next task. However if the ID is not known with suitable confidence, a subsequent image would be obtained. The secondary image could be obtained such that the fusion with the first look is optimized. This optimal fusion angle would be determined by the radar configuration as well as the specific features of the target’s most likely identifications.

In other mission scenarios, UAVs are tasked to image only a specific swath of the battlefield. For such tasks, circular SAR (CSAR) techniques allow for three dimensional target imaging [73]. To obtain this three dimensional image, however, complete circular aspects at multiple elevations are needed; this is often a prohibitive amount of data to be collected. Alternatively, the persistence model could be applied to obtain a 2-D estimate with much lower data requirements. Moreover, circular flight paths like the one in Figure 7-2 (a) require difficult maneuvering. Linear flight paths like (b) are often easier to fly.

Figure 7-2: UAV flight trajectories

When a UAV images a fixed target, images are often obtained in a circular aperture. This flight path, however, offers a challenge to the operator or navigation system. It is often easier to circumnavigate a target using a piecewise linear trajectory. Unique trajectories can require specialized processing.
In this example, the non-linearities in the flight path could be detrimental to CSAR algorithms. Instead of 3-D imaging, many individual 2-D images could be formed using narrow aspects. Additional processing would be required to reduce this dataset to a more manageable size for subsequent analysis. Either the highlight fusion algorithms or the persistence model would be well suited to solve this challenge. Through additional research, a new persistence mapping function could be implemented to help achieve mission goals such as change or anomaly detection.

### 7.1.2 Applications to Maritime Targets

The algorithms developed primarily dealt with terrestrial and air-based targets. The transition to maritime targets requires additional research. The sea-clutter, for example, is constantly moving and can represent a more complex background for SAR images than land [74]. Additionally, maritime targets themselves are also in motion. This motion must be combated in SAR, but is advantageous in ISAR. Although the time-varying nature of the ship’s movement allows for the formulation of an ISAR image, there are additional considerations [75].

As a UAV or other sensing modality images a ship, the look aspect will change over time. Similarly, the angular rotations of the target will also be dynamic. These concepts are illustrated in Figure 7-3 below. Due to these variations during capture, the data are better represented via a sequence of time-varying ISAR images rather than a single image [76]. Each frame will contain variable statistics due to the varying target
aspect as well as variable angular velocities. ISAR image fusion is an efficient method to account for the time varying data and can be used effectively.

Suppose a sequence of X-band $I/Q$-data vectors are collected when the target is rotating with constant angular velocity on the supporting plane from observing aspect $0.1^\circ$ to $82.199^\circ$. Traditional ISAR image generating methods would use the direct FFT to transform this complex data matrix to an ISAR image as shown in Figure 7-4. Note the high degree of blurring due to the large extent of the aperture. This amount of blurring will clearly obscure the important target features and make recognition and identification virtually impossible. However, the data matrix can be divided into continuous pieces

Figure 7-3: Data capture framework for maritime ISAR

An expansion of the research presented in this dissertation is to maritime targets. In the scenario illustrated above, a sensing platform illuminates a maneuvering target for an extended period of time. If a single image was obtained from the given data, the results would be highly undesirable. Instead, persistence or fusion could be applied to a sequence of images derived from sub-aperture images.
along the aspect angle. Upon applying an ISAR image fusion technique, the result is greatly improved as shown in Figure 7-5. The image fusion technique results in significantly less blurring and better resolution, thereby enhancing target recognition and identification using its strong scattering centers. Additional research would be needed to determine the optimal method for parsing the data.

Figure 7-4: ISAR images using all available data [77]

In this example the Slicy target was imaged during approximately a quarter of a rotation. Since the target exhibited a high degree of rotation, the scattering centers appear blurred. The target rotation will not be known and therefore the radar time-on-target to generate a single image will not be known.
The final expansion of this research is the ability to calculate three dimensional target transformations. Since maritime targets will rotate about all three axes, the projection of their point scatterers is nontrivial. The shadow projection methodology utilized a fixed ground plane in which the shadows were observed. The same projection does not exist for ships because of their multi-axis range of motion. Figure 7-6 illustrates multiple looks of a ship target that has undergone variable rotation.

Figure 7-5: ISAR fusion images using sub-apertures and fusion [77]

By dividing the initial data into 30 sub-apertures, the individual images can be fused. The target’s features are better resolved using fusion as compared to full aperture image formation. Additional research would be needed to determine the most appropriate sub-apertures in which to segment the initial data.
Since the ship rotates about all 3 axes, the projection of point sets is more complex than terrestrial based SAR imaging. The point scatterers will be scaled as a function of the Doppler frequency and the view angle will change according to the changes of angular velocity [78]. More complicated three dimensional projective transformations are needed to accurately account for these variations in maritime targets [79].

7.2 Goals and Accomplishments

Synthetic aperture radar and inverse synthetic aperture radar have a long proven history of operation for the purpose of non-cooperative target recognition. As battlefield capabilities advance, new opportunities emerge for exploiting the available data. One
such new capability is the ability to image a target from multiple looks. These new looks allow the systems to operate at a capacity greater than what would be possible with only a single radar image. The processing of these radar images within the multi-look paradigm was broken into four main subdivisions – pre-processing, highlight analysis, persistence modeling, and shadow analysis.

The goal of all the algorithms was to enhance the capabilities of NCTR using the SAR / ISAR imagery. Specifically, the goal of pre-processing was to align the images in a common projection plane, and segment the shadow, highlight, and clutter regions within a single radar chip. The purpose of the highlight analysis was to determine the benefits of combining multiple images and define a method of integration that optimized those benefits. The objective of the persistence model was to represent a large number of aspect images as a function of the target’s features. Finally, the shadow analysis aimed to prove that the use of shadow features could lead to a robust and reliable classification structure.

To begin, a number of canonical datasets were created. Ray-tracing methods were used to model the target-radar geometry and separately create an estimate of target reflectivity and target shadow. A 2-D height map was used to define the targets in a 2.5 dimensional space. A point scattering model was used to simulate isolated scatterers in an image similar to ISAR; occlusion was also included into this model to simulate the target’s physical structure. The canonical datasets were used for validation and testing of many of the algorithms. In addition to these three models, two real datasets were also used. The ERAD’s Slicy model was a well-conditioned dataset they obtained in laboratory conditions using a scaled target model. The final database, DARPA’s
MSTAR database, consisted of SAR data collected on real-world targets in a test range. A variety of targets (personnel carrier, reconnaissance vehicle, battle tank, cargo truck, and bulldozer) from this dataset were used for testing and analysis.

Before the main processing steps could be implemented, two necessary preprocessing steps were required – registration and segmentation. Image registration onto a common projection plane is important for many of the image-level algorithms. Registration ensures that the pixels in multiple images correspond to the same target features. Although general registration methods were not investigated, techniques were implemented to ensure the necessary registration for the given databases. A novel automated segmentation algorithm was developed that improved upon current systems’ capabilities. By combining diffused smoothing and Gaussian mixture modeling, the segmentation technique was able to reliably obtain good estimates of the highlight, background, and clutter regions within a single target chip.

Since many of the traditional SAR NCTR algorithms operate on the reflectivity returns of the scattering center, the highlight regions within the images were analyzed. Information integration was applied for systems that obtained multiple images of the same target at varying depression angles. The paradigm was examined for small numbers of looks. First, an image analog of an ROC curve was derived to evaluate the information content within SAR images. Next, the efficacy of varying fusion rules was tested.

The greatest gains in integration were not for complex probabilistic rules, but ones that took into account the nature of radar imaging. Two methods were proposed for counteracting the target shadow and occlusion – persistence averaging, and difference
mapping for wavelet fusion. The persistence averaging method fused two images linearly, but accounted for the areas of shadow within the image. The difference mapping rule used a disparity measure to estimate object occlusions; based on these estimates, a wavelet mapping function was defined. This rule allowed for the lowering of sidelobe power while simultaneously maximizing the power of target scatterers imaged by either radar. Both techniques offer novel improvements over traditional methodology for image fusion in radar imagery.

The final highlight analysis focused on determining the operational gains of multi-look imaging. The improvement of multi-look imagery was investigated as a function of both target number and target configuration. In a two-look scenario, the optimal configuration contains radars looking at opposite sides of the target, $+180^\circ$ aspect. The performance, however, was a function of configuration and target specific features. These two types of dependency were illustrated on a three-look system. In the absence of prior knowledge, the best configuration is equally spaced radars. However for a specific target, certain aspects contain superior information. Fusion was also illustrated to be a case of diminishing returns; on the data examined, little information was gained by fusing an excess of 5 looks.

Next, a persistence modeling framework was developed for handling cases when large amounts of aspect data was available. Specifically, the model focused on cases with rich aspect data from a single depression angle. The goal of the persistence model was to resolve rarely seen target features. The persistence modeling framework transformed an aspect dependent set of images to a target feature dependent set of images. The gains of the persistence model were illustrated for visualization and
classification. Visualization with the persistence model illustrated the intuitive benefits of the model over traditional aspect dependent imagery. Finally, the model was used to perform classification under the constraints of a reduced template dataset. The persistence model allowed for an order of magnitude decrease in the number of templates needed; alternatively it offered a more simplistic classification structure compared to traditional methods for a fixed number of templates and classification rate.

The final analysis focused on the shadow features within the radar imagery. Multiple classification algorithms were tested and a hidden Markov model ensemble was shown to be the most advantageous. Although the classification rate using a single-look was only slightly above 70% for a 5 target system, with as few as two looks the system had performance over 90% for certain radar configurations. In direct contrast to the highlight region, the optimal configuration for shadow classification fusion was seen to be orthogonal radars, ±90° in aspect. The shadow classification was similar to the highlight data in that it exhibited diminishing returns from adding additional looks. Two supplementary techniques were also developed to make the implementation more robust – model selection and shadow projection. The model selection criteria allows for optimal classification structures to be obtained dynamically as a function of mission specifications. The derivation of shadow projections allows for the estimation of shadow shapes from varying depression angles.

Both the highlight and shadow techniques supported the goal of exploiting multi-look radar imagery for NCTR. Multiple new algorithms were developed that utilized this framework to image, classify, or otherwise enhance the input data. The most notable accomplishments were an automated segmentation technique, novel image fusion rules, a
persistence framework, a shadow classification technique, and operational analyses. The utilization of these tools will allow surveillance, target, and imaging systems to: operate at longer ranges; operate in lower signal-to-noise ratios; function with greater performance; or otherwise perform at a higher level.
References


Appendix

G.1 ERADS Slicy Header

C 0, PEC (all metal) Slicey
C 3, 9/17/2004 Date
C 4, 2:07 PM Time
C 18, 9/17/2004 1:46 PM Date and Time of Cal
C 52, 5.7 Est run duration (Hrs)
C 28, ADXAA005000XCF040917A000SMS.DAT File Name
C 32, DX Target code
C 31, AA GP Code
C 1, Free space pylon. GP Desc
C 8, 2048 N Freq steps in full sweep
C 9, 0.000030 (S) Lock-in TC
C 50, Yes Lockin-Autoscale used
C 53, Yes Full BW calibration
C 47, 64 # Bins in target gate
C 48, 64 # Bins in cal gate
C 49, 4# Bins in in-scene gate
C 11, -5.0000 Primary Axis Begin
C 12, 370.0000 Primary Axis Range
C 13, 0.022117 Primary axis increment
C 40, 0.0 (S) Prime Axis settle time
C 16, 16730 # primary axis points
C 17, 5.000 Secondary Axis position
C 23, 160.000000 (GHz) CR Ctr Freq
C 24, 10.500000 (GHz) CR BW
C 104, 0.562 (*) CR Range Bin Res
C 201, 10.000000 (GHz) FS Ctr Freq
C 202, 0.665250 (GHz) FS BW
C 203, 8.9999 (*) FS Range Bin Res
C 25, 1# Scans/Avg
C 26, FS Aspect Prime Axis
C 27, Depression Cylinder Secondary Axis
C 29, 16.00 Scale Factor
C 30, Yes In-scene Cal used
C 33, C = SMS-160A @ NGIC Dataq location
C 34, Multi-axis stage Stage setup used
C 35, 0.300 Deg bistatic
C 36, Far-field on 72 inch reflector with 3dB zone of about 22 inches (two way distance) 29.3 ft at 16th scale. Measurement Field
C 37, Jack Sizemore Operator
C 42, X FS Radar Band Code
C 43, 1.264 (in) dia of Flat plate
C 44, 27.78 (dbsm) RCS of Flat plate
C 45, 1.143 (in) dia of dihedral
C 46, 26.03 (dbsm) RCS of dihedral
C 101, The typical target orientation procedure consists of: first the target is moved to either 0, 90, 180, degrees azimuth in respect to the radar beam. Then a visible HeNe laser that is co-aligned with the radar beam is back reflected from transmit to receive horns then an azimuth sweep of .1 increment from 85 to 95 degrees checking highest flash point return, then the azimuth is defined. Depression is locked at 0 degrees after calibration and a recheck of 0 degrees using a digital protractor. Target orientation determination
C 102, The calibration was performed using the calibration stage, which provides precision positioning of a calibrated metal dihedral object in the beam. A flat plate (single bounce) object is also used in the calibration. The flat plate is of a cylindrical cup shape design, which slips tightly over the dihedral to provide a flat polished surface to be co-aligned with the beam path of the dihedral. Typical alignment of the calibration objects is performed by back-reflecting the beam from a visible HeNe laser, which has been co-aligned with the radar beam, into the system receiver horn. Calibration object size is 651m2 dihedral and 1140 m2 flat plate. Cal procedure
C 103, Slicey with half round defined at 0 degrees azimuth. Target Configuration
C 60, No MT G Enable
C 298, For more information please contact: 434-951-1450
C 299, This data was generated by the National Ground Intelligence Center for Public Release
C Cal type = 4, RCS = 599.722290
C Cal type = 1, RCS = 401.002289
C Cal type = 3, RCS = 401.002289
C Cal type = 5, RCS = 401.002289
C Cal type = 7, RCS = 0.000000
C Processed in Linear Pol. 0¡ rotation
C Format after header= Aspect (F4) Length(I4) Data (C4) (HH, HV, VH, VV) repeated to end
G.2 MSTAR Header

[PhoenixHeaderVer01.04]
PhoenixHeaderLength= 01593
PhoenixSigSize= 00155049
PhoenixSigNum= 0001
PhoenixHeaderCallingSequence=
HeaderVersionNumber= 8CM
native_header_length= 0
Filename= hb14935.0024
Chip_MD5_CheckSum= 18a0fb30f28123f097f45697fa824e0f
ParentScene= hb14935
NumberOfColumns= 138
NumberOfRows= 139
TargetType= t72_tank
TargetSerNum= A64
TargetAz= 0.062515
TargetRoll= 1.526880
TargetPitch= 359.995209
TargetYaw= 358.931763
DesiredDepression= 15
DesiredGroundPlaneSquint= -90
DesiredSlantPlaneSquint= -90
DesiredRange= 5000
DesiredAimpointElevation= 39
MeasuredDepression= 14.968750
MeasuredGroundPlaneSquint= -90.088959
MeasuredSlantPlaneSquint= -90.085938
MeasuredRange= 4993
MeasuredAimpointElevation= 37.285000
MeasuredAircraftHeading= 160.109375
MeasuredAircraftAltitude= 1326.968994
RadarMode= mode 5 - spot light
SensorCalibrationFactor= 42.995998
RadarPosition= bottom
Range3dBWidth= 0.295200
CrossRange3dBWidth= 0.309400
SceneCenterReferenceLine= 160
X_Velocity= 45.943359
DataCollectors= Sandia National Lab
CollectionName= MSTAR Collection 2
Scene 1
SensorName= Twin Otter
Classification= UNCLASSIFIED
MultiplicativeNoise= -10 dB
AdditiveNoise= -32 to -34 dB
CenterFrequency= 9.599000 GHz
CrossRangeWeighting= -35dB_Taylor
RangeWeighting= -35dB_Taylor
DynamicRange= 64 dB
Bandwidth= 0.591 GHz
RangeResolution= 0.304700
CrossRangeResolution= 0.304700
RangePixelSpacing= 0.202148
CrossRangePixelSpacing= 0.203125
AverageImageCalFactor= 1.501722
Polarization= HH
TargetSeasonalCover= only growing vegetation
TargetWaterContent= dry
PhoenixLines= 0059
[EndofPhoenixHeader]
G.3 Small Scale Test Data

Figure A7-1: SAR chips of T-72

The SAR chips are representative images from 360° aspect of the T-72 target. The images are taken at 15° depression and are used for small scale testing of the classification algorithms.
Figure A7-2: SAR chips of BTR60

The SAR chips are representative images from 360° aspect of the BTR-60 target. The images are taken at 15° depression and are used for small scale testing of the classification algorithms.
Vita – Scott Papson

**Education:**
Rowan University 201 Mullica Hill Road Glassboro, NJ
BS Electrical and Computer Engineering GPA: 3.93 May 2003
MS Engineering with Specialization in Electrical GPA: 4.00 May 2004

The Pennsylvania State University University Park, PA
Ph.D. Electrical Engineering August 2007

**Related Experience:**
*The Johns Hopkins University Applied Physics Lab, Senior Research Staff* Laurel, MD, 20723
September 2007
Member of the Missile Defense Radar Engineering Group.

*The Pennsylvania State University, Research Assistant* University Park, PA, 16802
July 2004 -
August 2005
The focus of research was non-cooperative target recognition for radar imaging systems. Specifically, statistical signal detection and image processing techniques were applied for multi-look synthetic aperture radar (SAR) and inverse synthetic aperture radar (ISAR) data integration.

*Rowan University, Research Assistant* 201 Mullica Hill Road Glassboro NJ, 08028
Summer 2003 -
June 2004
MS research focused on evolutionary computational methods and data fusion algorithms for use in virtual reality environments. The initial application of the algorithms was non-destructive evaluation of gas transmission pipelines. Other responsibilities included supervision of undergraduate research teams and supervision of the virtual reality laboratory.

*Rowan University, Clinic Projects* 201 Mullica Hill Road Glassboro NJ, 08028
Fall 1999 -
May 2003
Rowan’s Clinic experiences allowed for the refinement of steps in the design process through practical application projects; important team leadership characteristics were also obtained. Through these different projects, theory learned in the classroom was applied to real world applications.

*Delsys Pharmaceuticals, Intern* 11 Deer Park Suite 118 Monmouth Junction NJ, 08852
May 2002 -
August 2002
Work included: integral involvement in implementation a hardware system; analysis of software components; expansion of a software system designed to run specialized hardware; and design of an optical measurement system.

**Society Memberships:**
IEEE, SPIE, Tau Beta Pi Honor Society, Penn State University Graduate Fellow (04/05)

**Research Interests:**
Image Processing & Computer Vision – target recognition, shape analysis, feature extraction
Data Integration – sensor & feature level fusion, high-level multi-domain conceptual fusion
Visualization – advanced scientific visualization for analysis, data presentation, frameworks for user interaction
Comparative Metrics – algorithm assessment, system simulation and comparisons, development of information measurements