Deep learning and featured-based classification techniques for radar imagery

Carole Belloni

THESE DE DOCTORAT DE

L'ÉCOLE NATIONALE SUPERIEURE MINES-TELECOM ATLANTIQUE
BRETAGNE PAYS DE LA LOIRE - IMT ATLANTIQUE
COMUE UNIVERSITE BRETAGNE LOIRE

ECOLE DOCTORALE N° 601
Mathématiques et Sciences et Technologies de l'Information et de la Communication
Spécialité : *Signal, Image, Vision*

EN COTUTELLE AVEC CRANFIELD UNIVERSITY
SIGNALS AND AUTONOMY GROUP
CENTER OF ELECTRONIC WARFARE

Par
Carole BELLONI

Deep Learning and Feature-Based Classification Techniques for Radar Imagery

Thèse présentée et soutenue à IMT Atlantique Brest, le 29/11/2019
Unité de recherche : Lab-STICC
Thèse N° : 2019IMTA164

Rapporteurs avant soutenance :

Jean-Paul Haton	Professeur émérite, Université de Lorraine
Matthew Richie	Professeur associé, University College London

Composition du Jury :

Président :	Salah Bourennane	Professeur, Centrale Marseille
Examinateurs :	Jean Paul Haton	Professeur émérite, Université de Lorraine
	Udo Uschkerat	Speaker Business Unit Defense, Fraunhofer Institut
	Thomas Merlet	Innovation Manager, Thales Optronique
Dir. de thèse :	Jean-Marc Le Caillec	Professeur, IMT-Atlantique
Dir. de thèse :	Nabil Aouf	Professeur, City University of London
Co-dir. de thèse :	Alessio Balleri	Professeur associé, Cranfield University

Invité(s)

Thomas MERLET	Innovation Manager à Thales Optronique

À mes parents…

Acknowledgements

I am extremely grateful to Prof. Nabil Aouf who provided constant encouragements and confidence in the work. His expertise provided key insights throughout the course of my study.

I express my sincere gratitude to Dr. Alessio Balleri for his patient guidance and his very relevant suggestions, always given with a smile. His knowledge and interest in radar was extremely valuable to me.

I would like to show my appreciation to Prof. Jean-Marc Le Caillec who provided helpful comments and suggestions.

My sincere thanks to Dr. Odysseas Kechagias for the many insightful discussions and multiple valuable feedbacks.

I would like to thank all the members of the jury, Prof. Salah Bourennane, Dr. John Economou, Prof. Jean Paul Haton, Dr. Thomas Merlet, Dr. Matthew Richie and Dr. Udo Uschkerat for their insightful comments and questions and for the time and interest they have shown to my work.

I am grateful to the MCM ITP and especially Thales who sponsored this research.

I express my heartfelt thanks to my parents, Isabelle and Philippe Belloni who supported me at all times and without whom I would probably not have enjoyed such opportunities.

A special thanks to Hugo Courtois who shared the PhD journey with me and nevertheless managed to motivate me every day.

Last but not least, I would like to thank all my friends at the laboratory for the laughs and support: Axel Beauvisage, Safiah Binti-Zulkifli, Brandon Corbett, Alejandro Dena,

Marco Di Fraia, Alexander Edward, Monica Estebanez, Luc Fourtinon, Gareth Frazer, Krasin Georgiev, Nasyitah Ghazalli, Amélie Grenier, Benjamin Griffin, Mathieu Issartel, Edward Jackson, Akhil Kallepalli, Leon Kocjančič, Alix Leroy, Hannah McGivern, Duarte Rondao, Raymond Vincent, Samuel Westlake, Sebastian Wirth, Ozgun Yilmaz.

Résumé long

Nous vivons actuellement un âge d'or de l'information caractérisé par l'augmentation drastique du volume des données produites ainsi que de leur complexité. Les principaux vecteurs de ces augmentations sont des transferts de données toujours plus simples et rapides, une multiplication et une amélioration des capteurs ainsi que le développement de meilleures technologies de stockage. Nous sommes actuellement capables de traiter tout au plus 20% des informations dont nous disposons. Mais dans le futur, l'acquisition de données devenant de plus en plus systématique car stratégique, ce pourcentage pourrait se réduire à seulement 2% [1]. L'analyse d'un tel volume de données ne peut se faire qu'en étant au moins partiellement automatisé. Des algorithmes issus du domaine de l'intelligence artificielle peuvent accélérer et simplifier le traitement de ce flux important de données. L'intelligence artificielle suscite un intérêt grandissant grâce à des résultats prometteurs, surpassant même parfois des experts dans leur propre domaine [2]. Ces algorithmes font déjà partie de nos vies quotidiennes à travers les recherches web, le filtrage de courriels, la gestion de flux d'actualités ou la recommandation de musique.

L'analyse de données concerne également les systèmes d'aide à la décision qui sont essentiels pour simplifier les choix opérateurs, en particulier dans des situations qui exigent des actions rapides ou avec des conséquences fortes. Le domaine d'application des ces systèmes est vaste, d'un tri préalable des appels aux urgences selon leur priorité, à une première analyse de l'environnement pour les pilotes de chasse. La demande concernant les systèmes d'aide à la décision est forte, en particulier dans le domaine de la défense. En effet, l'interprétation de menaces de plus en plus complexes est extrêmement difficileà réaliser, particulièrement dans un milieu où l'anticipation des actions est essentielle. Les algorithmes d'intelligence artificielle dans les systèmes d'aides à la décision donnent lieu

à des défis éthiques, légaux et technologiques. Le travail présenté dans cette thèse se concentre sur l'aspect technique.

Le travail proposé a été réalisé dans le cadre d'un projet visant à augmenter la quantité de données générées par des autodirecteurs Radiofréquence (RF). L'objectif principal de ce projet est de remplacer l'antenne orientable mécaniquement à l'avant de l'autodirecteur par une antenne réseau à commande de phase 3D [3]. Ces antennes ont de multiples avantages : une réduction des possibilités de pannes mécaniques ainsi qu'une réduction du volume nécessaire à l'antenne dû à l'absence de parties mobiles. La couverture du faisceau n'est également plus limitée par l'angle de rotation du système mécanique. L'orientation du faisceau est extrêmement rapide et peut se faire en quelques microsecondes [4]. De plus, les zones en dehors du champ de vision de l'antenne mécanique ne peuvent pas être analysées en même temps que la zone principale, ce qui empêche la potentielle détection de menaces venant d'autres directions que celle actuellement examinée. Toutefois, les antennes mécaniques ont aussi des avantages par rapport aux antennes réseaux à commande de phase : la cible étant dans la ligne de visée, l'énergie envoyée vers la cible est maximisée. De plus, tous les éléments radiants du réseau sont alignés sur une surface 2D pour laquelle les caractéristiques du faisceau sont bien connues. Le signal reçu sera donc plus facilement et mieux traité que pour un faisceau envoyé par une antenne réseau à commande de phase 3D, surtout si la direction d'émission n'est pas centrale.

L'avantage principal de l'antenne réseau à commande de phase 3D sur lesquels le travail de cete thèse se base est la possibilité d'envoyer et de recevoir différents faisceaux dans de multiples directions simultanément. Par conséquent, de multiples fonctions de l'autodirecteur peuvent être menées en parallèle. On se focalise ici sur la classification de cibles dans des images Radar à Synthèse d'Ouverture (RSO) qui pourrait être faite en même temps que la localisation de la cible principale. Ces nouvelles antennes donnent néanmoins lieu à de nouvelles difficultés, traitées par d'autres membres de ce projet. Certaines de ces difficultés concernent entre autres la forme de l'antenne, l'émission d'un faisceau par une antenne 3D et la différenciation des formes d'ondes pour les différents

signaux envoyés simultanément.

Un scénario potentiel mettant à profit cette technologie est décrit dans la Fig. 1.1. Un autodirecteur RF équipé d'une antenne réseau à commande de phase 3D mène deux tâches de front : l'autodirecteur suit la cible aérienne principale tout en envoyant des impulsions au sol durant toute sa trajectoire et jusqu'au point d'impact. Une fois reçus, ces signaux peuvent être traités de manière à fournir des images RSO de cibles potentielles au sol. Une grande antenne peut en effet être simulée avec une antenne réelle en mouvement grâce à des techniques de traitement du signal appliquées aux données radar acquises durant la trajectoire. Des images haute résolution peuvent donc être obtenues à partir d'une antenne réelle plus petite qui sans cette technique donnerait des résultats de résolution moindre.

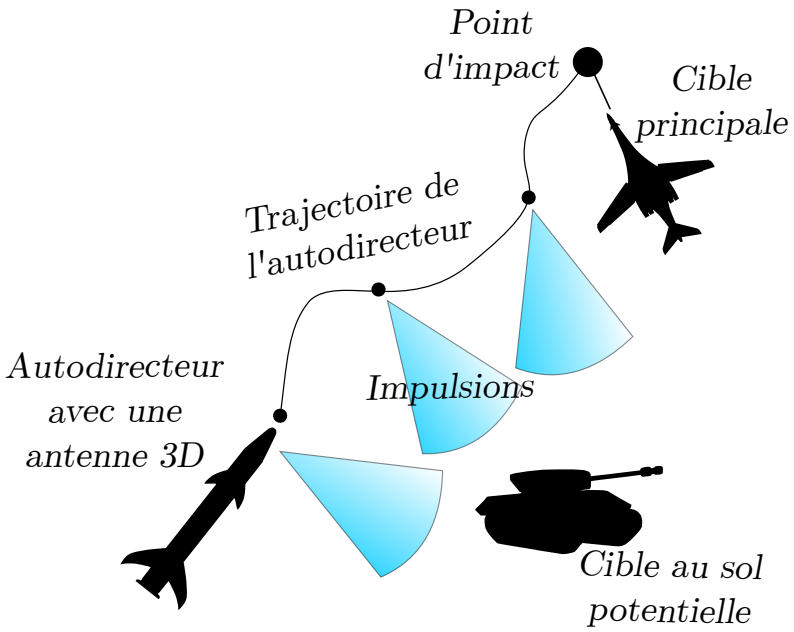


Fig. 1: Scenario de défense illustrant la classification de cible sur des images RSO.

Les images RSO présentent de multiples avantages par rapport aux données opto-électroniques. Ces données peuvent en effet être acquises dans des conditions météorologiques variées et durant la nuit.

Toutefois, l'interprétation des images RSO est complexe. Une formation est nécessaire pour que les opérateurs puissent les interpréter. En effet, les opérateurs doivent analyser des informations qui ne sont pas intuitives puisque les images RSO ne sont pas issues

du même phénomène physique que les images visibles, plus répandues. De manière à réduire leur charge de travail et à accélerer la classification de cible, de la Reconnaissance Automatique de Cible (RAC) peut être appliquée aux images RSO. La RAC peut aussi être motivée par la nécessité pour des systèmes comme les systèmes embarqués d'opérer sans intervention humaine.

Cette thèse vise à étudier et proposer l'implémentation de nouvelles techniques de RAC sur image RSO en les comparant avec des méthodes plus traditionnelles. Après la proposition d'une solution innovante reposant sur l'apprentissage profond, les raisons menant aux décisions prises par le réseau de neurones sont étudiées.

Objectifs de recherche et contributions

L'objectif de cette thèse est de donner des réponses aux questions de recherche suivantes:

Comment évaluer de manière impartiale les algorithmes de RAC sur image RSO? Une des difficultés majeures de la RAC sur image RSO est le nombre et la variabilité limitée des images disponibles. Cela a un impact non seulement sur le dévelopement des algorithmes de RAC sur image RSO mais aussi sur l'évaluation des performances de ceux-ci. En effet, certaines bases de données présentent seulement des différences minimales entre les sets d'entraînement et de test. Une nouvelles base de donnée RSO Inverse (RSOI) est proposée dans cette thèse. Cette base de donnée inclue 3 cibles pour un total de 1728 images. Ces images peuvent être divisées en 24 groupes et sont séparées entre les sets d'entrainement et de test de manière à ce que les conditions d'acquisition soient différentes. Ces conditions d'acquisition englobent la configuration de la cible, l'angle d'incidence et l'environement du laboratoire. L'arrière-plan est supprimé artificiellement pour diminuer le risque de corrélation.

Pour évaluer les progrès en terme de robustesse des algorithmes proposés, des tests sont réalisés sur des images dans lesquelles la cible est déplacée de manière aléatoire, ce qui permet de vérifier que l'algorithme ne dépend pas d'une localisation particulière.

Dans quelle mesure les techniques de classification du domaine visible peuvent-elles être transférées au domaine RSO?

Beaucoup de travaux ont été menés sur les images optiques car elles sont disponibles en grand nombre et utiles pour de nombreuses applications. Il est donc intéressant de voir si ce travail pourrait être réutilisé dans le le domaine RSO et dans quelle mesure cela pourrait être fait.

Une méthode de segmentation innovante basée sur des Modèles de Mélange de Gaussienne (MMG) est proposée. Les segmentations proposées précédemment reposaient sur des méthodes de seuils. Le score de Dice évaluant la performance de la segmentation a été améliorée grâce à cela de 11%. A notre connaissance, la segmentation à base de MMG n'avait pas été testée auparavant sur de la RAC d'images RSO.

Des classifications diverses reposant sur des caratéristiques visuelles sont comparées sur les images RSO en utilisant des descripteurs habituels du domaine optique. Certains de ces descripteurs, comme les descripteurs binaires n'avaient jamais été testés sur les images RSO. Ces descripteurs binaires sont, selon les résultats obtenus dans cette thèse, moins affectés par le chatoiement typique du domaine RSO et peuvent améliorer le seuil de classification de 40% par rapport à des descripteurs de gradients.

La grande sensibilité des descripteurs aux variations de l'orientation de la cible est également montrée et quantifiée.

La classification peut-t-elle être améliorée en prenant en comptes les spécificités des images RSO?

Les méthodes classiques d'augmentation de données visent à produire de nouvelles images en utilisant des transformations de l'image simples comme la translation, la symmétrie axiale, le recadrage ou l'ajout de bruit gaussien. Une méthode d'augmentation de données spécifique au domaine RSO est développée. Celle-ci est basée sur l'ajout de bruit basé sur une distribution de Weibull au profil de distance utilisé lors de la création des images RSO. Cette technique d'aumentation de données RSO permet une augmentation du score de classification sur la base de données proposées de 86% à 91%.

Après avoir montré l'influence de l'orientation de la cible lors de la classification par descripteurs, une nouvelle architecture d'apprentissage profond est proposée. Cette architecture attribue la tâche de classification à un réseau de neurones spécialisé dans la classification de cibles ayant une orientation particulière. Cette méthode dite "pose-informed" surpasse un réseau de neurones convolutionnel standard sur 4 des 5 datasets sur lesquels elle a été testée, avec en particulier une amélioration de 96% à 99% dans le cas du dataset le plus utilisé dans le cadre de la RAC d'images RSO.

Cet algorithme nécessite au préalable d'avoir correctement identifié l'orientation de la cible. La plupart des algorithmes estimant l'orientation des cibles dans le domaine du SAR le font à seulement modulo 180°, or la méthode précédente nécéssite également d'avoir différencié l'avant de l'arrière de la cible. Une association entre un réseau de neurones convolutif et une transformée de Hough est donc proposée pour accomplir cette tâche. Les erreurs dans l'estimation de l'orientation de la cible avec cette méthodes sont plus faibles que dans les méthodes proposées jusque là.

Peut-on appréhender les critères de classification des méthodes de RAC pour les images RSO issues du domaine de l'apprentissage profond?

Comme le raisonnement aboutissant à la décision de classification des réseaux de neurones ne peut être détaillé, des études sont proposées sur les explications possibles du comportement de ces réseaux. En effet, l'explicabilité des algorithmes d'apprentissage profond dans le domaine RSO restait limité à la visualisation des filtres des couches les plus basses des réseaux.

Des cartes de classification sont proposées pour mettre en évidence les zones d'intérêt communes à un groupe d'image pour le réseau. Cela apporte de nouvelles informations sur la localisation des caractéristiques les plus importantes pour une cible spécifique ou pour des cibles dans des orientations similaires.

L'ombre de la cible est peu utilisée selon nos expériences, du moins pour le réseau étudié. Des travaux de recherche supplémentaires visant à un meilleure utilisation des informations relatives à l'ombre pourraient donc améliorer les performances de classifi-

cation.

Une dernière étude montre que lors de son entraînement pour la classification de cibles, le réseau de neurones apprend également à différencier des caractéristques liées à l'orientation de la cible. Le réseau de neurones apprend donc des connaissances supplémentaires qui ne sont pas reliées directement à sa tâche principale.

Cela pousse à l'utilisation du transfer learning d'une tâche à une autre (utilisation pour une certaine tâche d'un réseau entraîné précédemment sur une autre tâche comme réseau de départ) plutôt qu'un simple transfert entre différentes bases de données. De ce fait, des images SAR auparavant ignorées pourraient être utilisées pour une première phase d'entraînement, sachant que les données RSO exploitables directement pour la RAC sont peu nombreuses.

Organisation de la thèse

Le thème et le contenu de chacun des 7 chapitres sont détaillés ci-dessous:

- Le chapitre 2 présente le processus d'acquisition des données RSO et RSOI. Le manque de données diversifiées est l'un des challenges principaux de la RAC sur images RSO. Un nouveau set de doonées RSOI est fourni dans l'objectif de permettre une évaluation plus juste des algorithmes de classification. Par rapport aux données publiques déjà disponibles, ce nouveau set d'images présente plus de différences entre les images destinées à l'entraînement et à l'évaluation des algorithmes [5, 6]. L'influence de la corrélation dû à l'arrière plan est également évaluée.

- Le chapitre 3 aborde des travaux préliminaires essentiels à la compréhension des algorithmes présentés dans les chapitres suivants. Ce chapitre traite essentiellement de vision assistée par ordinateur avec une introduction aux transformées de Hough et aux méthodes d'extraction de descripteurs d'image. Les réseaux de neurones et leurs méthodes d'entraînement sont aussi présentés.

- Le chapitre 4 compare différents méthodes de classification par descripteurs dans le but d'évaluer le potentiel des descripteurs conçus originellement pour des images optiques et non RSO. Certains de ces descripteurs n'avaient jamais été appliqués au domaine RSO. L'influence de l'orientation de la cible dans les méthodes de classification par descripteur est quantifiée [7]. Cette influence sera réutilisée dans le chapitre 5. Une segmentation basée sur du machine learning est proposée en amont de la classification par descripteurs. Cette segmentation consiste à modéliser le clutter par un MMG (Modèle de Mélange de Gaussiennes). On note une amélioration de la précision ainsi que du taux de rappel comparé aux méthodes par seuils d'intensité [8].

- Le chapitre 5 se focalise dans un premier temps sur le problème du manque de données RSO diversifiées avec la proposition d'une méthode d'augmentation de données produisant de nouvelles images avec l'addition d'une simulation de bruit propre aux données RSO [9]. La deuxième partie de ce chapitre montre que la détermination et l'utilisation de l'orientation de la cible dans la classification par réseaux de neurones profonds améliore la précision dans la plupart des cas. L'orientation de la cible est prise en compte dans l'architecture pose-informed proposée dans cette thèse [10]. L'orientation de la cible est déterminée en associant une transformée de Hough avec un réseau de neurones convolutionnel ce qui permet une amélioration de la précision par rapport aux algorithmes actuels de détermination de l'orientation sur 360°.

- Le chapitre 6 présente une analyse visant à l'explicabilité de réseaux de neurones profonds [11]. L'objectif est de comprendre les raisons poussant le réseau de neurones à une certaine décision de classification. Différentes analyses évaluent l'influence de zones particulières dans l'image dans la décision finale de classification. Ces zones peuvent être spécifiques à un certain type de cible, à son orientation ou à ce qu'elles représentent dans l'image comme la cible, son ombre ou le

clutter. La distribution des intensités composant les zones essentielles à la classification sont étudiées et comparées aux distribution de zones non nécessaires. Enfin il est démontré que les descripteurs appris par le réseau de neurones ne sont pas seulement spécifiques au type de cible mais également liés à d'autres variables de l'environement d'acquisition des données, bien qu'elles ne soint pas directement inclues dans la fonction que le réseau de neurones cherche à optimiser. Cela pourrait encourager l'apprentissage par transfert des réseaux de neurones sur différentes tâches plutôt que différentes bases de données.

- Le chapitre 7 suggère de possibles travaux supplémentaires compte tenu du travail de recherche présenté dans les chapitres précédents.

Contents

List of Figures

List of Tables

List of Equations

List of Abbreviations

ATR	Automatic Target Recognition
BRISK	Binary Robust Invariant Scalable Keypoints
BF	Box Filters
CAD	Computer-Aided Design
CNN	Convolutional Neural Network
DoG	Difference of Gaussians
EM	Expectation Maximisation
EOC	Extended Operating Conditions
FREAK	Fast Retina Keypoint
GMM	Gaussian Mixture Model
ISAR	Inverse Synthetic Aperture Radar
KL	Kullback-Leibler divergence
KNN	K-Nearest Neighbour
LoG	Laplacian of Gaussians
MGTD	Military Ground Target Dataset
MSTAR	Moving and Stationary Target Acquisition and Recognition
NN	Nearest Neighbour
ORB	Oriented FAST and Rotated BRIEF
PCA	Principal Component Analysis
RAM	Radiation-Absorbent Material
RCS	Radar Cross-Section

SAR	Synthetic Aperture Radar
SGD	Stochastic Gradient Descent
SGDM	Stochastic Gradient Descent with Momentum
SIFT	Scale Invariant Feature Transform
SOC	Standard Operating Conditions
SURF	Speeded Up Robust Features
SVM	Support Vector Machine
VNA	Vector Network Analyser

Chapter 1

Introduction

Contents

1.1 Overview and motivations

The information age has seen the amount of data produced explode and the information generated become increasingly more complex. This is mainly due to the facilitation and acceleration of data transfer, the improvement and multiplication of sensors and the development of storage technology. On their own, humans are currently able to process at their best 20% of the information at their disposal but in the future, with the acquisition of data becoming ever so strategic and systematic, this percentage could drop to only 2% [1]. The processing of such amounts of data calls for at least partial automation of data analysis. Algorithms relying on artificial intelligence (AI) have the potential to reduce time and stress strains created by the analysis of this flow of data. AI is currently gathering a lot of interest and already providing encouraging results, sometimes even outperforming experts in their own field [2]. Such algorithms are already partaking our everyday lives

with web searches, e-mail filters, news feeds filtering, music recommendation and the like.

Decision support systems are an aspect of data analysis that is gaining interest to simplify decision-making for operators, especially in time sensitive or high impact situations. This can range from the pre-sorting of emergency calls according to their priority, to an environmental pre-analysis for fighter pilots. The demand for decision support systems is especially high in the defence sector. Indeed, the interpretation of threats with a growing complexity is extremely challenging in a domain for which anticipation and accuracy are paramount. Such algorithms give rise to challenges across the ethical, legal and technological domains. The presented work will focus on the technological challenge.

The proposed work stems from a wider defence project that is going to increase the amount of data generated for RF-seekers. The global objective is to integrate a 3D electronically steerable antenna at the front of the seeker to replace the traditional mechanically steered antenna [3]. There are several advantages for a phased array antenna. There are less possibilities of mechanical faults and a reduction of the volume needed for the antenna as there are no moving parts. The beam coverage is not limited by the angle of rotation of the mechanical system. The beam steering is also much faster and can occur in a matter of microseconds [4]. For the mechanical antenna, the areas outside the field of view cannot be investigated at the same time, preventing the detection of potential threats in another direction than the one currently investigated. On the other hand, the target is kept in the line of sight of the antenna which maximises the energy sent towards the target. All the radiating elements of the mechanical antennas are aligned on a 2D surface and thus the beam characteristics are well known. The returned signal will be more easily and accurately processed than for a more complex beam sent by a phased array, especially in a direction that is not central.

The advantage of the 3D electronically steerable antenna that prompted this work is the ability to send and receive multiple beams in different directions at the same time. Multiple seeker functions can thus be achieved simultaneously. The focus here is on the

target classification by Synthetic Aperture Radar (SAR) images that could be carried out in parallel to the main target tracking. 3D electronically steerable antennas also give rise to other challenges, addressed by other people involved in this project. Some of these challenges concern the antenna shape, the emission of a beam with a 3D antenna or the differentiation of waveforms for the different signals transmitted simultaneously.

A potential scenario using this technology is given in Fig. 1.1. A RF-seeker equipped with a 3D antenna performs two tasks at the same time: The seeker can follow a primary air target while sending pulses to the ground throughout the trajectory until the impact point. The returns from the pulses sent to the ground can be processed to provide SAR images of potential ground targets. A large antenna can indeed be simulated with a constantly moving smaller physical antenna by using processing techniques on the radar data acquired during the trajectory. High resolution images can be thus obtained using a small antenna that, used on its own, would give results with a poorer resolution.

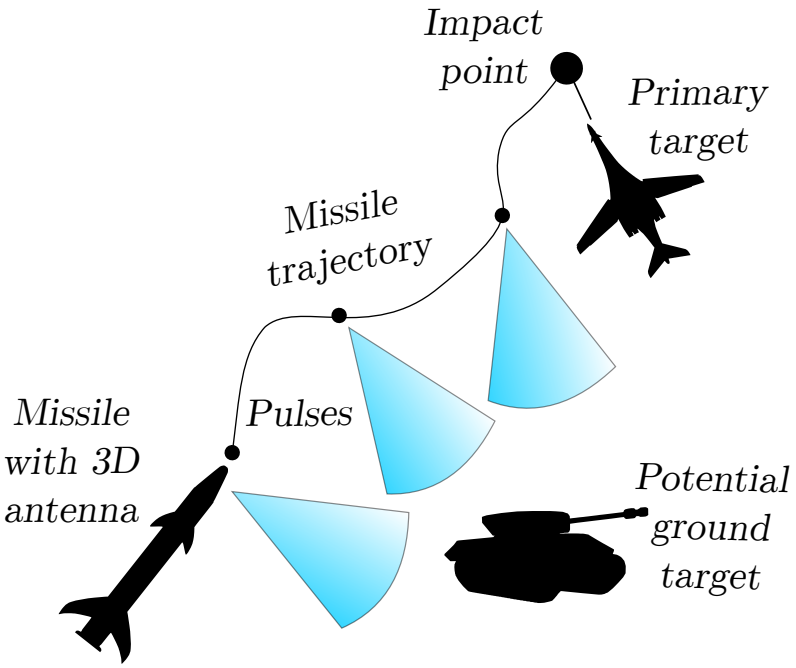


Fig. 1.1: Defense scenario requiring SAR ATR.

SAR data is interesting because of its multiple advantages over electro-optical data. It is indeed useful under a wider range of weather conditions and also during the night.

However, the interpretation of SAR data is challenging. So that operators can interpret

them, they will need to be trained. Operators have to analyse extra information that is hard to understand as SAR images are very different to interpret than usual electro-optical images. To reduce this workload and to speed up the recognition process, automatic target recognition (ATR) can be applied to SAR data. Another case motivating ATR would be if the interpretation of data has to be done on-line. No operators are involved in the understanding of the images generated by the embedded system and ATR is mandatory as there is no possible human intervention.

The presented work aims at implementing novel techniques alongside more traditional methods, in order to perform SAR ATR. In particular, after implementing a novel deep learning solution for SAR ATR, the reasons behind the decisions of the deep learning network are studied.

1.2 Research objectives and contributions

The objective of this thesis is to provide answers to the following research questions:

How to fairly evaluate SAR ATR algorithms?

A key challenge of SAR ATR is the lack of diversity and the small amount of images available. This has an impact not only on the development of SAR ATR algorithms but also on the performance evaluation of such algorithms. Indeed, some databases present only little differences between training and testing sets. In this thesis, a new ISAR database is proposed. This database contains 3 targets for a total of 1728 images. These images stem from 24 different group images and are dispatched between training and testing so that the acquisition conditions are partly different. These conditions include the target configuration, the depression angle and the lab environment. The background is then artificially removed to undermine potential undesired correlation.

In order to evaluate the robustness improvement made by the proposed classification algorithms, some test are performed on images that were randomly translated in order to investigate if the algorithm is location dependant.

To what extent can optical classification methods be transferred to the SAR domain?

A lot of work has been carried out on optical image classification as optical images are available in large numbers for diversified applications. It is interesting to see if the work carried out in the optical domain can be transferred to SAR ATR and, if so, to what extent.

An innovative GMM based segmentation is proposed. Previous segmentations in the literature were handled with threshold methods. The segmentation Dice score achieved in this thesis is improved by 11%. To the best of our knowledge, GMM segmentation has not been tried before on a SAR ATR database.

Various feature-based classification methods are compared on SAR data using well-studied descriptors in the visual domain, some of which, such as binary features were never tested on SAR data. Results show that binary features are less influenced by the speckle and can increase the classification rate by 40% compared to gradient features. The high sensitivity of descriptors to orientation variation of the target is shown and quantified.

Can classification be further improved by taking into account the specific characteristics of images in the SAR domain?

Classical data augmentation methods aim at creating new images by applying simple transformations such as translation, flipping or cropping of the image or the addition of a Gaussian distribution. A SAR specific data augmentation is developed by adding SAR specific noise based on a Weibull distribution to the range profiles. These range profiles are used to create the SAR images. Using this SAR data augmentation, the classification score relative to our database is improved from 86% to 91%.

After noticing the influence of the target orientation with the feature classification, a different architecture that attributes the classification task to a deep neural network specialised in classifying targets from a specific orientation range is proposed. This pose-informed deep learning method achieves better classification scores on 4 out of the 5 datasets it was tested on, with an improvement from 96% to 99% in the case of the most used SAR ATR dataset compared to a CNN without the orientation specificity. This al-

gorithm relies on a correct estimation of the target orientation. Most of the orientation determination algorithms in SAR have been focused on a 180° precision while the pose-informed architecture needs the full 360°. Thus, an association between a deep learning network and a Hough transform to perform this task is proposed. The orientation errors achieved are lower than alternative current available methods.

Can we better understand the classification criteria of deep learning SAR ATR methods?

As deep learning methods decision making flow cannot be detailed, explanations of the deep learning model's behaviour are investigated. The understanding of deep learning algorithms in the SAR domain was so far focused only on visualising low-level features of the network. Classification maps are proposed to highlight the common zones of interest of groups of images to the network. This provides new information on the location of the most important features of specific target classes and targets in specific orientation range.

It is also shown, at least for the studied network, that the target shadow is little used. Further work to better include shadow information could thus bring improvement of the current classification rates.

It is finally shown that, while training for target classification, the deep network also learns to discriminate features specific to the target orientation. The network learns additional knowledge that is valuable outside its main task. That encourages the possibility of training networks using transfer learning from one task to another rather than only from one database to another. SAR data previously ignored could be deemed useful in a first training instance considering the lack of data available for SAR ATR.

1.3 Organisation of the thesis

The topic and content of each of the 7 chapters is detailed below.

- Chapter 2 introduces the acquisition of SAR and ISAR data. It also presents the databases that will be used in the rest of the thesis. The lack of diverse data is one

of the main challenges of SAR ATR. A new ISAR dataset is presented with the objective to allow a fairer classification evaluation, alternative to what is available with already existing data, with more differences between training and testing set [5, 6]. The influence of background correlation of the datasets is also evaluated.

- Chapter 3 gives some introductory material necessary to understand the algorithms used in later chapters. This chapter focuses on the computer vision with the introduction of the Hough transform and feature extraction methods. It also introduces deep neural networks and their training procedure.

- Chapter 4 provides a comparison of several feature classification methods with the goal to assess the transferability of features originally issued from the optical domain (some of which have never been applied to the SAR domain). The influence of the target orientation in feature classification and the challenge of corner detection is quantified [7]. The fact that the target orientation influences classification scores is used in Chapter 5. A proposition of a machine learning segmentation precedes the feature classification. This segmentation is based on the clutter modelling using GMMs resulting in an improved standard precision and recall rate compared to existing threshold methods [8].

- Chapter 5 firstly focuses on the limited amount of diverse data available for SAR ATR with the proposition of a data augmentation solution that provides extra images with realistic SAR noise [9]. The second part of the chapter shows that determining and including the target orientation in the deep learning classification method improves the classification rates in most cases. The target orientation is taken into account by using the proposed pose-informed architecture [10]. The target orientation is determined using an association of Hough transform with a CNN and achieves a better precision than current algorithms on 360°.

- Chapter 6 presents an analysis to explain deep learning models [11]. The objective is to understand the reasons behind a classification decision obtained with deep

learning. Analysis are proposed to assess the influence of particular image areas in the final classification decision. These areas could be specific to the target class, target orientation or to what they represent in the image, i.e. shadow, target or clutter. The distribution of the areas leading to classification is also shown to be different than that of the other image areas. The features that are learnt by the network are shown to not only be specific to the target class but to other environmental variables as well, even if they were not included in the loss function. This could further the use of transfer learning across tasks rather than only across databases.

- Chapter 7 proposes some further work based on the research presented in the previous chapters.

Publications resulting from this work

Carole Belloni, Alessio Balleri, Nabil Aouf, Thomas Merlet and Jean-Marc Le Caillec. "SAR image dataset of military ground targets with multiple poses for ATR". In: *Target and Background Signatures III*. Vol. 10432. International Society for Optics and Photonics. Warsaw, Sept. 2017, 104320N.

Carole Belloni, Nabil Aouf, Jean-Marc Le Caillec and Thomas Merlet. "Comparison of Descriptors for SAR ATR". In: *2019 IEEE Radar Conference (RadarConf19)*. IEEE. Boston, Apr. 2019.

Carole Belloni, Nabil Aouf, Thomas Merlet and Jean-Marc Le Caillec. "SAR image segmentation with GMMs". In: *International Conference on Radar Systems (Radar 2017)*. IET. Belfast, Oct. 2017.

Carole Belloni, Nabil Aouf, Jean-Marc Le Caillec and Thomas Merlet. "SAR Specific Noise Based Data Augmentation for Deep Learning". In: *2019 IEEE International Radar Conference*. IEEE. Toulon, Sept. 2019.

Carole Belloni, Nabil Aouf, Alessio Balleri, Jean-Marc Le Caillec and Thomas Merlet. "Explainability of Deep SAR ATR Through Feature Analysis". In: *IEEE transactions on aerospace and electronic systems* (2020). Submitted. Accepted with minor revision.

Odysseas Kechagias-Stamatis, Nabil Aouf, David Nam and Carole Belloni. "Automatic X-ray image segmentation and clustering for threat detection". In: *Target and Back-*

ground Signatures III. Vol. 10432. International Society for Optics and Photonics. Warsaw, Sept. 2017, 104320O.

Odysseas Kechagias-Stamatis, Nabil Aouf and Carole Belloni. “SAR Automatic Target Recognition based on Convolutional Neural Networks”. In: *International Conference on Radar Systems (Radar 2017)*. IET. Belfast, Oct. 2017.

Chapter 2

Generation of SAR and ISAR data for ATR

Contents

2.1 Summary

Frequency range	13 GHz to 18 GHz sampled with 4001 frequency points
Resolution	3.0 cm (range)×3.3 cm (cross-range)
Target class number	3 classes (T64, T72 and BMP1)
Number of images	1728 images from 24 distinctive sequences (Training: 864. Testing: 864.)

Table 2.1: Summary of MGTD (Military Ground Target Dataset) dataset

Evaluations of SAR ATR techniques are currently challenging due to the lack of publicly available data in the SAR domain. Existing SAR ATR algorithms have mostly been evaluated using the MSTAR dataset [12]. The various MSTAR dataset, acquired under various conditions are described in details. The problem with the MSTAR databases is that some of the proposed ATR methods have shown good classification performances even when targets are hidden [13], suggesting the presence of a bias in the dataset. An alternative to the standard 3 targets MSTAR dataset is proposed that minimises this bias.

In addition, a high resolution SAR dataset consisting of images of a set of ground military target models taken at various aspect angles is proposed. The dataset can be used for a fair evaluation and comparison of SAR ATR algorithms. The Inverse Synthetic Aperture Radar (ISAR) technique is applied to echoes from targets rotating on a turntable and illuminated with a stepped frequency waveform. The targets in the database consist of three 1.5-1.7 m long models of T64, T72, BMP1 tanks. The gun, the turret position and the depression angle are varied to form 24 different sequences of images. The emitted signal spanned the frequency range from 13 GHz to 18 GHz to achieve a bandwidth of 5 GHz

sampled with 4001 frequency points. The resolution obtained with respect to the size of the model targets is comparable to typical values obtained using SAR airborne systems. Single polarised images (Horizontal-Horizontal) are generated using the backprojection algorithm [14]. A total of 1728 images are produced using a 20° integration angle. The images in the dataset are organised in a suggested training and testing set to facilitate a standard evaluation of SAR ATR algorithms.

2.2 Introduction

As SAR ATR algorithms are currently developed, the availability of datasets that allow a fair evaluation of the performance is essential. If various ATR methods could be tested on the same datasets, the results could be better compared and improvements could be established more effectively. A dataset of this kind should represent a variety of operating conditions, such as different target viewing angles, different target configurations, layovers or occlusions induced by the surrounding environment, as well as target movements effects. With a more realistic dataset the likelihood of overfitting is decreased because the images are less similar with each other. The database that has been most often used to test SAR ATR algorithms is the Moving and Stationary Target Acquisition and Recognition (MSTAR) [12]. The problem with the MSTAR is that the independence between training and testing sets in the MSTAR has also been questioned [13], in both the 3 targets and the 10 targets database. It is essential to achieve reliable results in any recognition problems with independent training and testing dataset. As a result of these limitations, ATR algorithms could have shown artificially increased results and their performance not accurately reported. This chapter presents the commonly used MSTAR dataset and an alternative 3 targets dataset using MSTAR images. In addition to that, a new dataset dedicated to SAR ATR made with new ISAR images acquired in a laboratory is proposed [6]. A prior version of this new database has already been published [5]. The experimental set-up and all the environmental factors are presented in Section 2.6.1. Guidelines are

also given to correctly select the sequences to form independent and varied training and testing sets in Section 2.6.2. Following this discussion, an evaluation method is suggested, so that all algorithms are evaluated in the same way.

2.3 SAR and ISAR theory

The parameters used throughout this section relative to SAR and ISAR techniques are as follow:

Parameter	Description
r	Range of the studied scatterer.
x	Cross-range of the studied scatterer.
R_C	Range of the target centre.
d	Distance from the scatterer to the target centre.
θ	Angle relative to the polar coordinates of the scatterer as defined in Fig. 2.4.
Ω	Rotational speed of the turntable.
c	Speed of light.
f	Frequency of the signal.
Δf	Frequency step of the chirp.
K	Number of frequency steps.
N_p	Number of pulses during the integration time.
λ	Wavelength.
f_θ	Angular frequency.
f_t	Temporal frequency.
B	Frequency bandwidth.
τ	Pulse duration.
T	Processing time duration.
θ_a	is the integration angle.
Δr	Range resolution.
Δx	Cross-range resolution.
W_r	Range dimension of the ISAR image.
W_x	Cross-range dimension of the ISAR image.

2.3.1 Definitions

Resolution The resolution is the minimum distance needed between two scatterers so that the radar is able to tell them apart. Range resolution focuses on the difference of

(a) Transmitted signal on two scatterers.

(b) Received signal from the two scatterers.

Fig. 2.1: Illustration of the returned signal from two scatterers closer than the radar range resolution.

range distance between two scatterers. The radar is able to tell the scatterers apart only if the time to effectuate the round trip between the two scatterers is longer than the whole signal duration as shown in Eq. (2.1). Otherwise, the signals received from both scatterers are not separated but superimposed and the targets appear as a single target.

$$\Delta r = \frac{c \cdot \tau}{2} \tag{2.1}$$

In practice, two targets are distinguishable as long as the two signals do not overlap too much. The precise updated range resolution is obtained with the matched filter method that is used to localise precisely the received signal. The result of the matched filter of the chirp sent on the received signal is a sinc function with its main lobe centred on the received chirp location. The signals can slightly overlap as long as the part of the main lobes with a loss lower than 3dB do not overlap as they contain the main signal power. The width of the main lobe at a 3dB loss is $\frac{1}{B}$. By replacing the time needed between two signals to be resolved independently previously obtained τ with $\frac{1}{B}$, a range resolution defined as in Eq. (2.2) is obtained.

$$\Delta r = \frac{c}{2B} \tag{2.2}$$

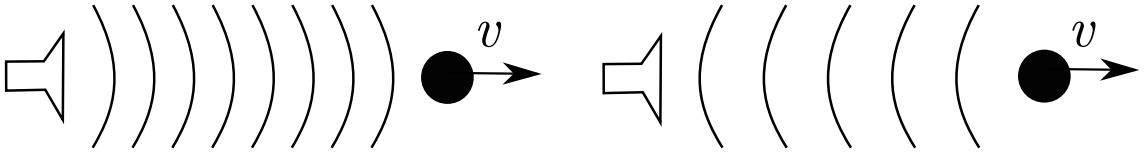


Fig. 2.2: Illustration of the Doppler effect for a target moving from the antenna.

Doppler effect The Doppler effect consists of a frequency shift of the signal when it is reflected on a moving target. Indeed, for a target moving at a speed v from the antenna, the received signal will be partly delayed and can be defined as in Eq. (2.3). The range r of the scatterer in this case depends on the time t and on the initial range $r(0)$.

$$s(t) = \exp\left(j2\pi f\left(t - \frac{2r(0)}{c} - \frac{2vt}{c}\right)\right) \tag{2.3}$$

The phase related to the Doppler shift here is:

$$\Phi_D = \frac{4\pi fvt}{c} \tag{2.4}$$

And the corresponding Doppler frequency defined as the derivative of the phase is:

$$f_d = \frac{1}{2\pi}\frac{d\Phi_D}{dt} = \frac{2fv}{c} \tag{2.5}$$

The Doppler shifts gives the opportunity for the radar to resolve two scatters if they move at different speeds. The speed difference between the scatterers must be different enough so that the radar is able to notice the frequency difference between the two signals.

2.3.2 SAR principle

The SAR main principle is to simulate a longer antenna by transmitting several pulse regularly to the ground, as a delayed phased array. This require to know the movement of the plane acquiring the images, and the targets trajectories if they are moving [15]. It is assumed in this thesis that the targets and the antenna do not move at the same time. In the case of ISAR images, it is the target that is moving and the antenna that is still, which is similar to the SAR image configuration after a referential change.

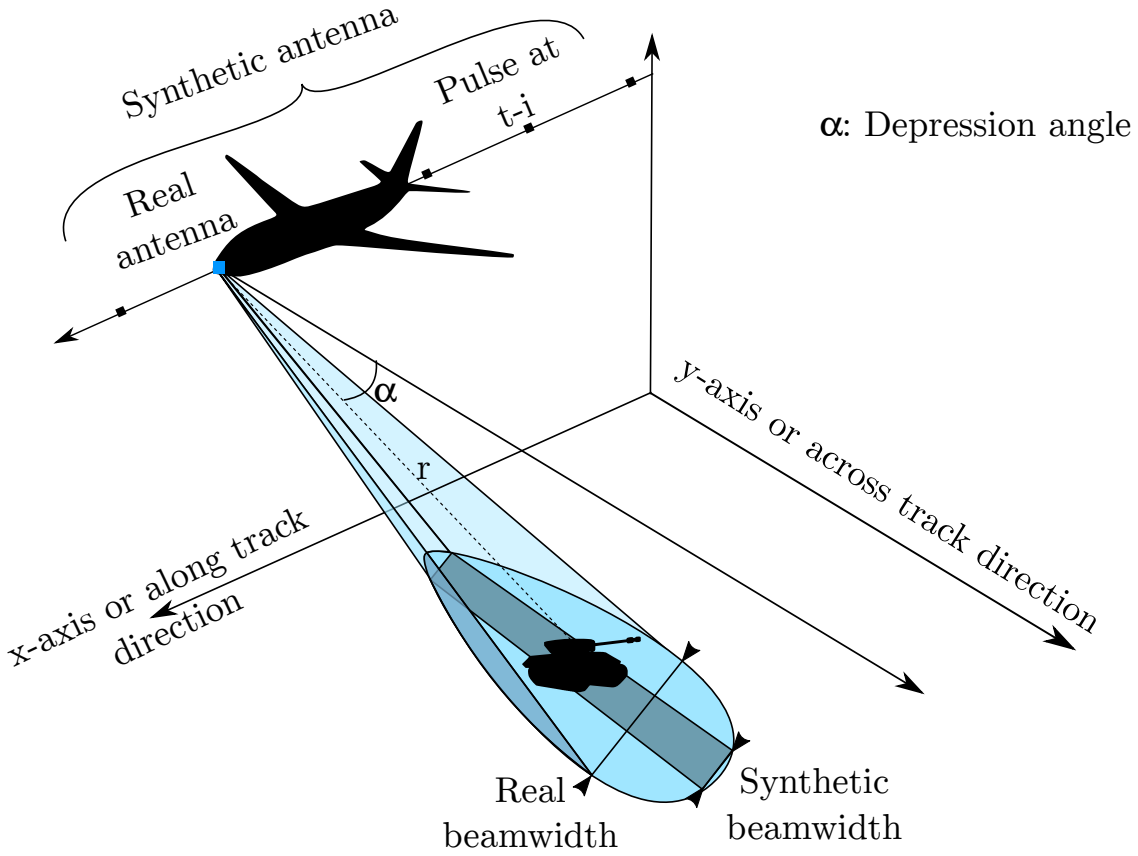


Fig. 2.3: Setup to acquire SAR images.

In a classical SAR scenario, the antenna is assumed to move on a straight line at a constant speed, while sending signal regularly to the ground at a certain depression angle as shown in Fig. 2.3. All the measurements are stored to be compiled later on and simulate a longer antenna. The combination of all the measurements is possible by using some Doppler shifts measurement that give information on the location of each zone at the time of measurement. Indeed, the Doppler shift is related to the plane location in regards to the imaged zone with the speed contributing to the Doppler shift varying from one extreme to another between the beginning and the end of the acquisition.

2.3.3 ISAR principle

In the ISAR model, it is the target that moves in front of a fixed antenna. The target here rotates on a turntable as seen in Fig. 2.4. The following explanations are based on [16]. The cross-range position of scatterers are deduced from the Doppler shift of the received

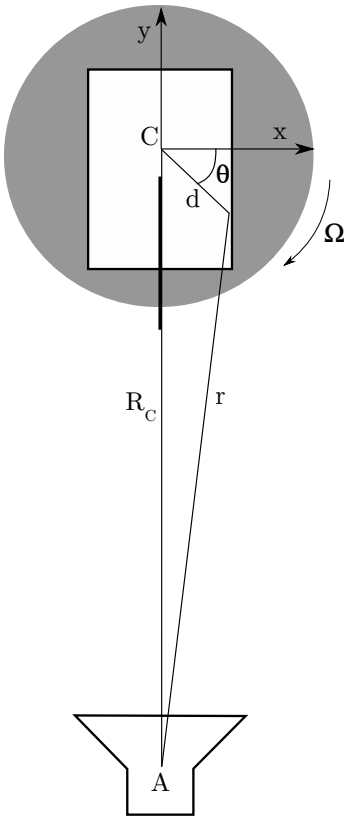


Fig. 2.4: Setup to acquire ISAR images.

signal. In order to compute the Doppler shift, the range between each scatterer composing the target and the antenna has to be calculated throughout the movement. Starting from the geometric configuration presented in Fig. 2.4, the range of any source point on the target is deduced in Eq. (2.6).

$$r = R_C - d\sin\theta \tag{2.6}$$

The received signal by the antenna issued from the studied scatterer can be written as the transmitted signal with a round-trip delay due to the range found in Eq. (2.6) as in Eq. (2.7).

$$s(t) = \exp\left(j2\pi\omega\left(t - \frac{2r}{c}\right)\right)$$

$$s(t) = \exp\left(j2\pi\omega t - j2\pi\frac{2R_c}{\lambda} + j2\pi\frac{2d\sin\theta}{\lambda}\right) \tag{2.7}$$

Using Eq. (2.7), the Doppler frequency is calculated by differentiating the phase associated with the Doppler shift Φ_D in Eq. (2.8). The frequency shift can be associated with the cross-range location of the scatterer x.

$$\Phi_D = 2\pi\frac{2d\sin(\Omega t)}{\lambda}$$

$$f_t = \frac{1}{2\pi}\frac{d\Phi_D}{dt} = \Omega\frac{2d}{\lambda}\cos(\Omega t) \tag{2.8}$$

$$f_t = \Omega\frac{2x}{\lambda} \tag{2.9}$$

Similarly, the angular frequency can be calculated if the phase is differentiated against θ instead of time to obtain Eq. (2.10).

$$f_\theta = \frac{1}{2\pi}\frac{d\Phi_D}{d\theta} = \Omega\frac{2d}{\lambda}\cos\theta \tag{2.10}$$

$$f_\theta = \frac{2x}{\lambda} = \frac{f_t}{\Omega} \tag{2.11}$$

This particularity means that there are two options to compute the ISAR image, either having the target rotate at a constant speed, or doing the rotation by step and transmitting the signal when the target is still. In both cases, the cross-range cell of the scatterer can be retrieved from the frequency shift.

Cross-range resolution

The cross-range resolution can be deduced from Eq. (2.8). The resolution of the Doppler frequency using a matched filter can be approximated by $\frac{1}{T}$. The cross-range resolution can thus be deduced in Eq. (2.12).

$$\Delta x = \frac{\lambda}{2\Omega}\Delta f_t = \frac{\lambda}{2\Omega T} \tag{2.12}$$

$$\Delta x = \frac{\lambda}{2\theta_a} \tag{2.13}$$

Maximum alias free image dimensions

Range dimension The number of range cells is the number of frequency points K as generated by the VNA during the acquisition. Thus, with Eq. (2.2), the range dimension of the maximum alias free image can be deduced as in Eq. (2.14).

$$W_r = \Delta r \cdot (K-1) = \frac{c}{2} \cdot \frac{K-1}{B} = \frac{c}{2 \cdot \Delta f} \tag{2.14}$$

Cross-Range dimension The number of cross-range cells is the number of pulses sent N_p. Thus, with Eq. (2.12), the cross-range dimension of the maximum alias free image can be deduced as in Eq. (2.15).

$$W_x = (N_p - 1)\Delta x = \frac{\lambda}{2}\frac{(Np-1)}{\theta_a} = \frac{\lambda}{2 \cdot \Delta\theta} \tag{2.15}$$

Backprojection algorithm

The intensity of each point in the ISAR image is retrieved with the backprojection method. It is a method more computationally heavy that the matched filter method. However, the

intermediary computation of the range profile is interesting to visualise some of the work that is done on the images such as the approximated background removal in Chapter 2 or the data augmentation in Chapter 5.

The distance d between the target centre and the scatterers is calculated firstly for each pixel and each pulse. Each range bin of the range profile can be calculated as an inverse Fourier transform of the received signal with the inclusion of the delays induced by d. Ultimately, the intensity at a certain range is computed as the summation of the values of the range profiles of all pulses at this specific range.

More details as well as an example of Matlab code can be found in [14].

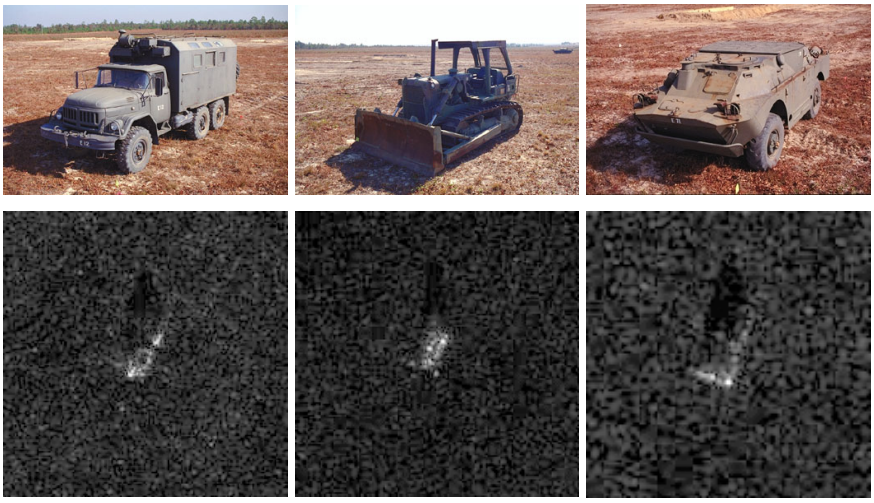
2.4 Description of the MSTAR dataset

The MSTAR public dataset was developed by the U.S. Defense Advanced Research Projects Agency (DARPA) and the U.S. Air Force Research Laboratory (AFRL) [12, 17]. This database was collected under Horizontal Horizontal (HH) polarisation in X-Band with a $30\,\text{cm} \times 30\,\text{cm}$ resolution.

2.4.1 Targets

The database is composed of 10 different targets shown in Fig. 2.5 with their visual and SAR representation. The various targets range from different categories with a bulldozer (D7), a truck (ZIL), a rocket launcher (2S1), an air defence unit (ZSU), armoured personnel carriers (BRDM2, BTR60, BTR70, BMP2) and tanks (T62, T72). The particularity with the BMP and T72 targets is that there are several tanks representing the same target. Moreover, for the T72, various variants are available as seen in Fig. 2.6 with fuel barrels, a skirt or a reactive armour.

[1] Photos of the BMP2 and BTR70 from the MSTAR dataset were not found and are replaced with alternative photos of the same tank models. These photos were taken by Vitaly Kuzmin (`https://www.vitalykuzmin.net`) and are licensed under the Creative Commons Attribution Non Commercial No Derivatives 4.0 International License.



(a) ZIL (b) D7 (c) BRDM2

(d) BTR60 (e) BTR70 [1]

Fig. 2.5: Target photos and corresponding SAR image.

(f) BMP2 [2] (g) T62 (h) T72

(i) 2S1 (j) ZSU

Fig. 2.5: Target photos and corresponding SAR image (continued).

2.4.2 Datasets

Class	Training (17°)		Testing (15°)	
	Serial number	image number	Serial number	image number
BMP2	sn_c21	233	sn_c21	196
	sn_9566	232	sn_9566	196
	sn_9563^{B}	233	sn_9563^{B}	196
T72	sn_132^{B}	232	sn_132^{B}	196
	sn_812	231	sn_812	195
	sn_s7	188	sn_s7	191
BTR70	sn_c71^{NA}	233	sn_c71^{NA}	196

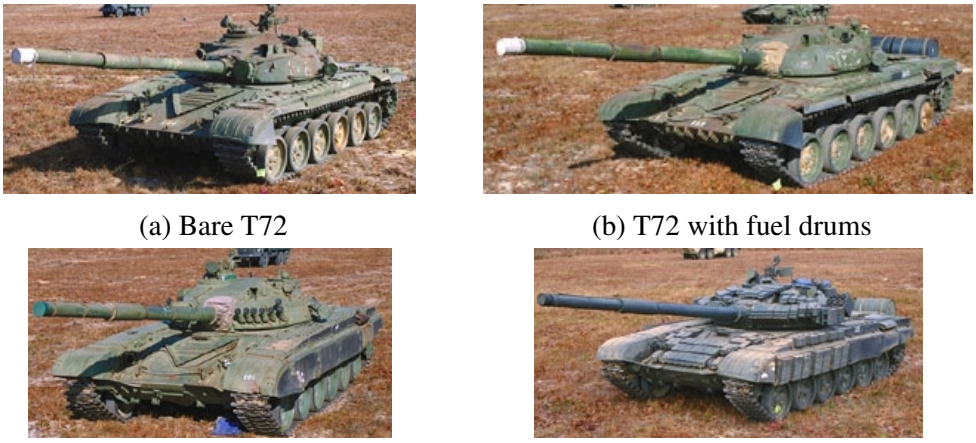
Table 2.2: MSTAR dataset SOC - 3 targets. Referred as *Dataset A* in Section 4.4.

Class	Training (17°)		Testing (15°)	
	Serial number	image number	Serial number	image number
BMP2	sn_9563^{B}	233	sn_9563^{B}	196
BTR70	sn_c71^{NA}	233	sn_c71^{NA}	196
T72	sn_132^{B}	232	sn_132^{B}	196
BTR60	sn_k10yt7532NA	256	sn_k10yt7532NA	195
2S1	sn_b01^{B}	299	sn_b01^{B}	274
BRDM	sn_E71^{NA}	298	sn_E-71NA	274
D7	sn_92v13015NA	299	sn_92v13015NA	274
T62	sn_A51^{F}	299	sn_A51^{F}	273
ZIL	sn_E12^{NA}	299	sn_E12^{NA}	274
ZSU	sn_d08^{B}	299	sn_d08^{B}	274

Table 2.3: MSTAR dataset SOC - 10 targets.

In the nomenclature designating the sequences, *sn_X*, X designates the serial number of the target [18]. The sequences produced make possible to test ATR algorithms in standard operating conditions (SOC) with a constant target and two different depression angles (2° difference) between training and testing as suggested in [18]. The standard conditions make possible to test the ATR on 3 or 10 different targets such as in Tables 2.2 and 2.3. The MSTAR is also composed of data making the ATR algorithm deal with more realistic changes with different target variants, target configurations or bigger change in

[2] Photos of the BMP2 and BTR70 from the MSTAR dataset were not found and are replaced with alternative photos of the same tank models. These photos were taken by Vitaly Kuzmin (`https://www.vitalykuzmin.net`) and are licensed under the Creative Commons Attribution Non Commercial No Derivatives 4.0 International License.



(a) Bare T72 (b) T72 with fuel drums

(c) T72 with a skirt (d) T72 with a reactive armour

Fig. 2.6: Various variants of the T72.

the depression angle (13° difference). These stronger changes between training and testing make up the extended operating conditions (EOC) datasets as seen in Tables 2.4 to 2.6. In all these tables, *F* refers to the presence of fuel drums on the target, *S* to the presence of a skirt, *R* to the presence of a reactive armour, *B* means that the tank is bare, without fuel drums, skirt or reactive armour. *NA* means that the presence of such variances are not applicable and the absence of precision means that the relative information could not be found. There are three EOC datasets available. The depression EOC or EOC 1 dataset has a high difference in depression angle between the training and testing set (13°). For the variant EOC or EOC2 dataset, the targets of the same class have differences from the manufacturer and have been built to different blueprints. Concerning the configuration EOC or EOC3 dataset, something is removed or added to the target such as fuel barrels, skirt [18]. These EOCs are commonly used in deep learning on the contrary to EOCs proposed in [19] as not all data are available [20–22]. A few other EOCs have also been proposed [23].

Proposition of an alternative MSTAR dataset for the 3-target ATR problem

An alternative partition is proposed in Table 2.7, named dataset B, to the standard one for 3-class targets SAR ATR in Table. 2.2, named dataset A. The series chosen in dataset B have been chosen so that the serial number is different between training and testing

Class	Training (17°)		Testing (30°)	
	Serial number	image number	Serial number	image number
2S1	sn_b01^{B}	299	sn_b01^{B}	288
BRDM	sn_E-71NA	298	sn_E71^{NA}	289
T72	sn_132^{B}	232	sn_A64^{B}	288
ZSU	sn_d08^{B}	299	sn_d08^{B}	288

Table 2.4: MSTAR dataset EOC 1 - Depression variant

Class	Training (17°)		Testing (15° & 17°)	
	Serial number	image number	Serial number	image number
BMP2	sn_9563^{B}	233	sn_9566	196+232=428
			sn_c21	196+233=429
BRDM	sn_E-71NA	298	-	-
BTR70	sn_c71^{NA}	233	-	-
T72	sn_132^{B}	232	sn_812NA	195+231=426
			sn_A04^{F,S}	275+299=573
			sn_A05^{S}	274+299=573
			sn_A07^{S}	274+299=573
			sn_A10^{S}	271+296=567

Table 2.5: MSTAR dataset EOC 2 - Version variant

for the BMP2 and T72. The targets are thus not entirely identical in both sets and even if not moved during the acquisition, the background will still be different between the training and testing set. The correlation between two sets should be thus reduced and the dataset choice is justified by applying a template based ATR method to dataset A and B with the full images and background-only images in which the target was removed after segmentation in Section 4.4.2 [13, 24].

2.5 Reasons for creating a new dataset

Algorithms performing ATR rely on a training set to recognise targets in a testing set. To ensure a fair analysis of the performance of these algorithms and avoid any possible bias in the results, training and test images should be taken from independent sets of data.

It has been shown that the MSTAR [12] contains data with a high degree of correlation between images in the training and testing set due to the presence of correlated

Class	Training (17°)		Testing (15° & 17°)	
	Serial number	image number	Serial number	image number
BMP2	sn_9563^{B}	233	-	-
BRDM	sn_E-71NA	298	-	-
BTR70	sn_c71^{NA}	233	-	-
T72	sn_132^{B}	232	sn_s7	191+288=419
			sn_A32^{F,S,R}	274+298=572
			sn_A62^{F}	274+299=573
			sn_A63^{F}	274+299=573
			sn_A64^{B}	274+299=573

Table 2.6: MSTAR dataset EOC 3 - Configuration variant

Class	Training (17°)		Testing (15°)	
	Serial number	image number	Serial number	image number
BMP2	sn_c21	233	sn_9566	196
			sn_9563^{B}	196
T72	sn_132^{B}	232	sn_812	195
			sn_s7	191
BTR70	sn_c71^{NA}	233	sn_c71^{NA}	196

Table 2.7: MSTAR dataset - 3 targets alternative. Referred as *Dataset B* in Section 4.4.

background [13, 25]. Indeed, it has been demonstrated that the recognition rates of algorithms tested on the MSTAR are high even when the target to recognise is artificially hidden. Moreover, data released to the public and included in the guidelines [18] contains only two targets (BMP, T72) with a complete training and testing sets, and one, the BTR60, with only one sequence for each set. In order to evaluate the correlation on the MSTAR datasets, a simple ATR method is applied on full SAR images as well as on SAR images with the target hidden and compare the results.

2.5.1 Nearest neighbour classification

Nearest neighbour classification method

The training and testing set are chosen according to Section 2.4.2. The nearest neighbour classification method consists in comparing the image to be classified in the testing set to every image in the training set. To be able to compare all the images, they are scaled

to the same size. The scale chosen was 64×64 pixels to have a quick matching but still some precision to the image. This is done using a bicubic interpolation for the images of different sizes. That means that the pixels without a proper intensity assignation get as an intensity a weighted average of the 4×4 neighbouring pixels with assigned values. Once both images are the same size, the global intensity distance is computed between both images as in Eq. (2.16).

$$D(I_1, I_2) = \sum_{x=1}^{n} \sum_{y=1}^{m} |I_1(x,y) - I_2(x,y)| \tag{2.16}$$

where I_k is one of the image to compare.
$I_k(x,y)$ is the intensity of I_k at row x and column y.

This distance is computed for each test image with all the available training images. Out of all these distances, the class of the test image is chosen to be the same as the class of the target in the training image with the lowest distance to the test image.

The full images are classified as well as images with the central area covering the target hidden as in Fig. 2.7 so that the target representation does not interfere in the classification. The target is hidden using a mask which is a black square of 34×34 pixels added in the middle of the image. The partially hidden images are classified using training images with the central area covered as well. Thus, two classification scores are computed for each dataset with SOC (3 and 10 targets) and EOC (4 targets with different depression angles, variants, configurations): one for the full images and one for the masked image.

Results of the nearest neighbour method on full SAR images and partially masked SAR images

The classification scores are already high, especially for the SOC datasets, with a very basic classification method as seen in Table 2.8. Even when the target is hidden, the NN method still manages to classify well up to 6 times the score attained by a random classifier. The scores are lower for the EOC datasets which means the correlation affects

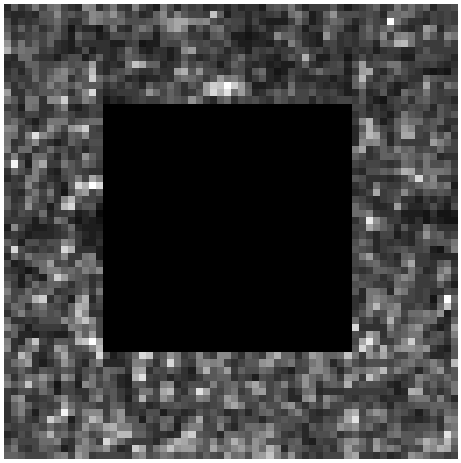


Fig. 2.7: SAR image with the target masked.

	SOC 3 targets	SOC 10 targets	EOC 1 (Depression)	EOC 2 (Variants)	EOC 3 (Configuration)
Full image	99,89%	87,30%	70,90%	75,04%	72,18%
Target hidden image	83,32%	61.01%	30,50%	49,99%	59,63%
Random classification	33,33%	10,00%	25,00%	25,00%	25,00%

Table 2.8: Classification results using the NN method on full SAR images and SAR images with a hidden target.

less the dataset with bigger environmental changes between the training and testing.

2.6 Description of the Military Ground Target Dataset (MGTD)

2.6.1 Data acquisition

Experimental setup

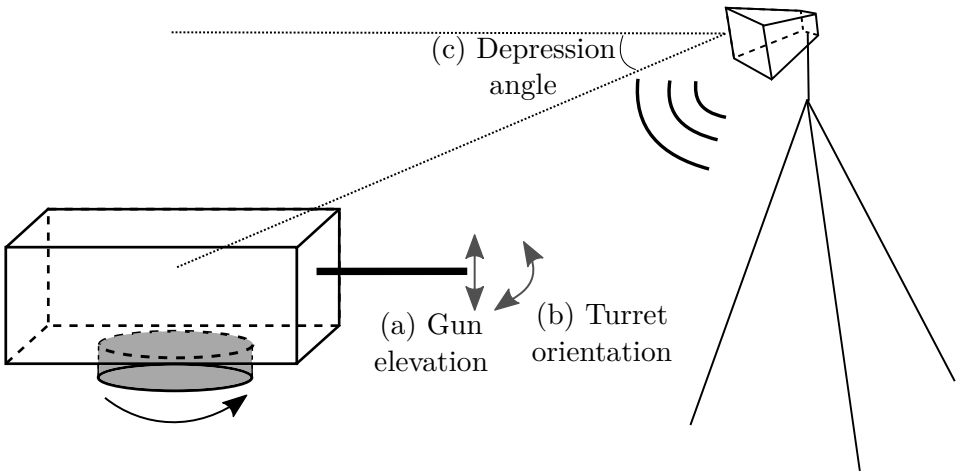


Fig. 2.8: Experimental setup. The antenna emits a signal towards the target placed on a turntable at each rotation step. For each sequence of measurements, at least one of the following factor is changed: the position of the gun (up/down), the orientation of the turret and the depression angle between the antenna and the target.

Configuration of the experiment The antenna is placed around 5 m away from the target on an adjustable height tripod to allow measurements or depression angles between 11° and 13°. The detailed range for each sequence of measurement can be found in Appendix A. The target is on a rotating turntable and high range resolution profiles are measured every predefined angular step. A single Horizontal-Horizontal polarisation is used. To avoid any ring due to unwanted movements of the setup, a latency period is introduced after each rotation step to ensure the target is still before each measurement. The emitted signal is a stepped frequency waveform spanning a bandwidth of 5 GHz between 13 GHz and 18 GHz, with 4001 frequency points. The signal is generated and acquired using an Anritsu Vector Network Analyzer (VNA). A piece of Radiation Absorbent Ma-

Fig. 2.9: Experimental setup. Some RAM is placed in front of the turntable to limit the unwanted returns from the turntable.

terial (RAM) is laid in front of the turntable (as shown in Fig. 2.9) in order to prevent unwanted multipath effects from the turntable.

Targets The presented database includes 3 classes. Their image representation of the target depends on the targets signature, i.e., the way it backscatters the energy sent by the radar. The various classes are characterised by major signature changes. The first class is the T64 tank shown in Fig. 2.10 (b). The second class is the T72 tank shown in Fig. 2.10 (c). The third class is the BMP1 tank shown in Fig. 2.10 (a). All three targets are model targets made mainly in plastic with some metallic parts. The BMP1 model is 1.5 m long and the T64 and T72 models are 1.7 m long.

Environmental variables To avoid correlation between the training and testing set in the MGTD, the images belonging to the training and testing set are separated so that the sets of environmental conditions are different between training and test images. Each target has also training images with varied environmental conditions so that the algorithm can be more resilient. A sequence is defined as a group of images obtained from one single experiment. Images from a single sequence have thus identical environment factors

(a) BMP1

(b) T64

(c) T72

Fig. 2.10: The different target classes.

except from the target orientation. Environmental details on the sequences chosen for the training and testing sets are in Sections 2.6.2 and 2.6.2. A detailed description of all the sequences created can be found in Appendix A.

Orientation The MGTD dataset consists of target images taken every 5° starting from 0° for which the radar faces the front of the target. The training and testing sets are formed using independent image sequences collected under different environmental conditions.

Depression Angle The depression angles used to generate the dataset are for training from 21.8° to 23.4° and for testing from 17.5° to 20.3°. The depression angle is changed by adjusting the height of the antenna mount relative to the distance between the antenna and the target. When the depression angle is changed, the reflecting surfaces of the target have virtually a new orientation which will impact the way the signal is backscattered. A change in the depression angle thus affects the target signature.

Configuration changes The configuration change is defined as the displacement of an element of the target. In practice, it is the turret or gun direction change that is considered as a configuration change. All orientations of the turret against its central position are included in the following sets of angles for the training: {-90; -45; 45; 90} and for the testing: {-30; 0; 30}. The gun had only two discrete positions which are up and down.

Lab environment Not all the sequences were taken at the lab at the same period. It is not possible to remove entirely the background from the images as it was not possible to take measurements of the background alone as the targets were too heavy to be moved easily. Thus, the surrounding objects in the background can have an incidence on the resulting image. To make sure to limit correlation relative to the background, the sequences chosen for the training and testing set are not taken at the same time period and have thus a different laboratory background. All the data were taken over 3 different time periods in time labelled 1, 2 and 3 in Section 2.6.2.

Experiment parameters

The choice of the bandwidth determines the range resolution of the image. With a bandwidth of 5 GHz, the range resolution of 3.0 cm is obtained through Eq. (2.17).

$$\Delta r = \frac{c}{2B} = 3.0\,\text{cm}, \tag{2.17}$$

where c is the speed of light and B is the bandwidth of the signal.

An integration angle of 20° seemed realistic as it represents a 350 m synthetic antenna for an altitude of 1 km. To have a similar resolution in the cross-range given this integration angle, the start frequency chosen is 13 GHz. Consequently, the cross-range resolution is of 3.3 cm using Eq. (2.18).

$$\Delta x = \frac{c}{2\theta_a f_{min}} = 3.3\,\text{cm}, \tag{2.18}$$

where Δx is the cross-range resolution, c is the speed of light, θ_a is the full integration angle and f_{min} is the start frequency of the signal.

This resolution is equivalent to a 17 cm resolution on a real-size tank (9.53 m T72) which is a value achievable with existing airborne SAR. For comparison, the resolution of the MSTAR is of 30 cm.

Knowing that the model-tanks described in Section 2.6.1 are maximum 1.7 m long, an estimation of 3.3 m with $\Delta\theta = 0.2°$ for the size of the total rotating scene seems adequate. Indeed, a bigger step angle of 0.4° leads to a 1.7 m rotating scene which is too close to the maximum target size. The maximum range size of the scene is calculated.

$$W_r = \frac{c}{2\Delta f} = 120\,\text{m}, \tag{2.19}$$

where W_r is the maximum range size of the scene, c is the speed of light, Δf is the frequency step.

The maximum cross-range size of the scene is calculated and must satisfy the same criteria as the range size of the scene.

$$W_x = \frac{c}{2\Delta\theta f_{min}} = 3.3\,\text{m}, \tag{2.20}$$

where W_x is the maximum cross-range size of the scene, c is the speed of light, $\Delta\theta$ is the step angle and f_{min} is the start frequency of the signal.

Reflecting the equipment constraints and the previous estimation, a rotation step of 0.2° is chosen to allow a 3.3 m maximum scene size as seen in Eq. (2.20), which is large enough to contain the tank model.

Image generation

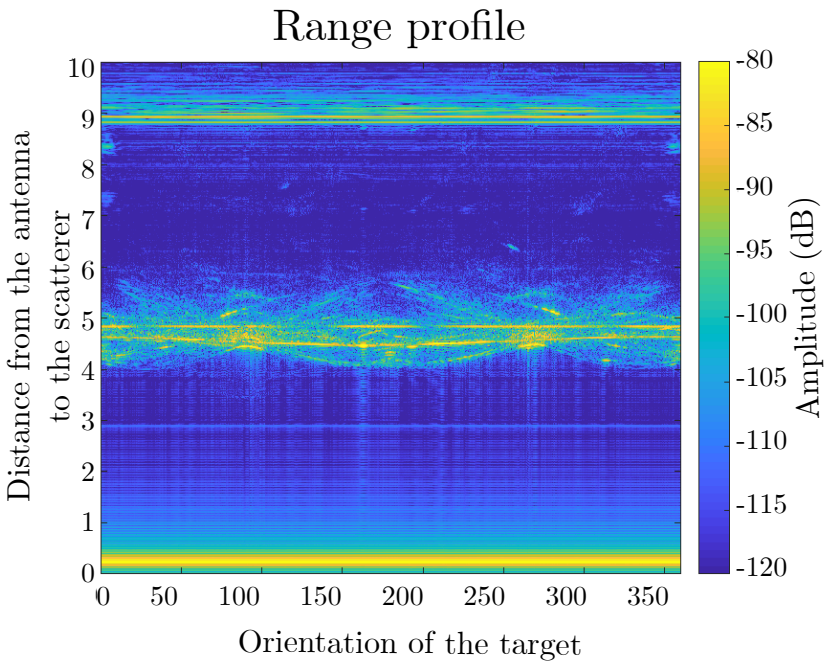


Fig. 2.11: Amplitude for each orientation of the target per range cell.

The target range profiles in Fig. 2.11 are obtained by taking the IFFT of the VNA output. For each target aspect angle, the backscattered signal amplitude is plotted for every range cell to obtain the sinogram of the main target scatterers. The range cells that contain the target are thus selected. Results show that the strongest echoes from the target are found between 4.25 m and 6.25 m.

Artificial background removal

As taking measures of the background alone was not possible due to the targets heavy weight and the duration of the acquisition, the background is present and affects the range

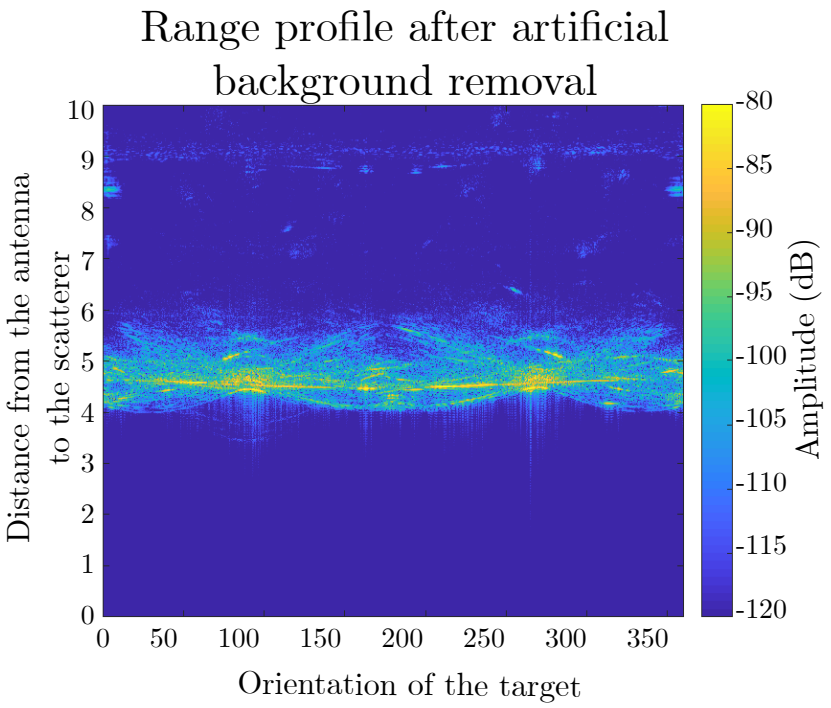


Fig. 2.12: Effect of the artificial removal of the background on the amplitude.

profile of the target. To improve the target image, an approximation of the background is done by calculating a sliding average over 8° of the whole range profile. The removal of the sliding average to the initial range profile makes possible to have an image that is less corrupted by the background . The effect of the sliding average removal on the range profile can be seen in Fig. 2.12.

Offset of the orientation starting angle The target is not always perfectly aligned with the radar at the start of the measurements. The angle correction needed is expressed as α in Fig. 2.13 and is listed in the Appendix A. This can be compensated at the image formation stage in which the measurements used for creating the image are shifted to simulate having the target at 0° from the start.

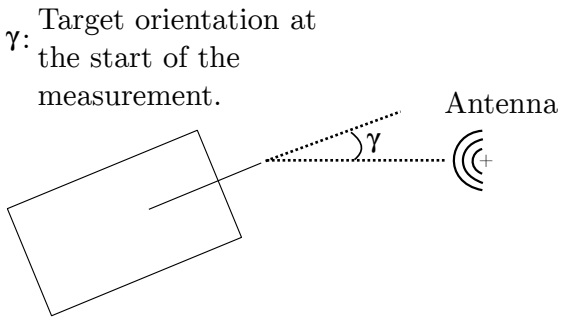
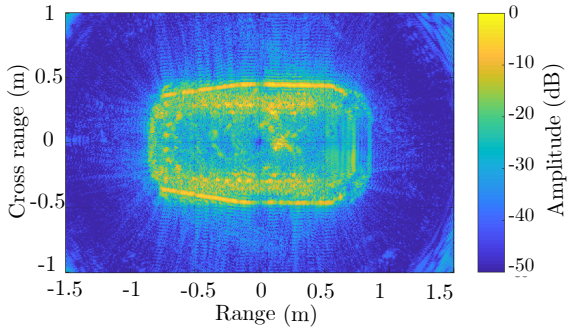
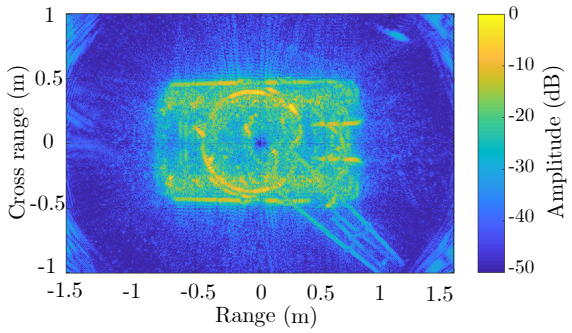


Fig. 2.13: Orientation correction for the misalignment of the target at the start of measurement.

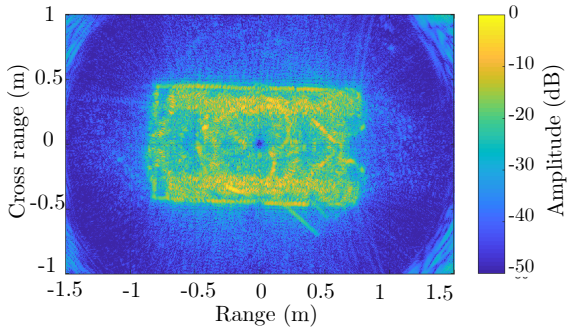
Targets ISAR representation To illustrate the signature changes between classes, the full 360° SAR images from each target generated with the backprojection algorithm are shown in Fig. 2.14. It can be seen that the T64 is more similar to the T72 than the BMP1. This is seen in particular in the turret, front and sides of the hull areas.



(a) BMP1



(b) T64

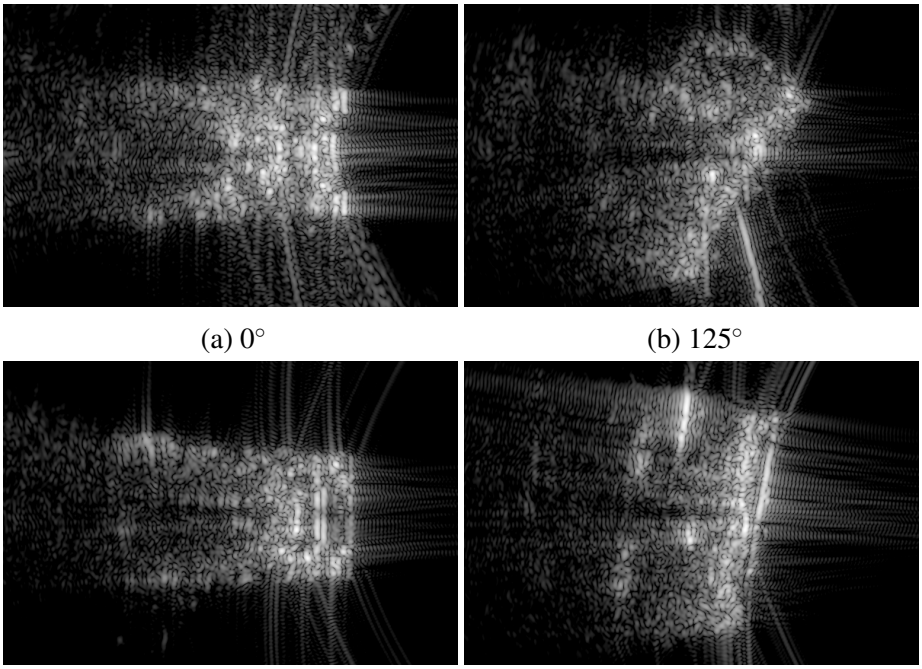


(c) T72

Fig. 2.14: ISAR images of the different target classes with a 360° integration angle.

Influence of the environmental variables on the ISAR images Various variables influence the representation of the target in the ISAR image. In this section, the influence of the orientation, depression angle and configuration of the target are shown. The ISAR images shown are images with an amplitude restricted to the [-50;0] dB range.

Orientation One of the limitations of SAR images is that small changes in the orientation of the target can result in vastly different signatures [26]. Fig. 2.15 shows different images from one unique sequence. The target is the T64 and apart from the orientation,



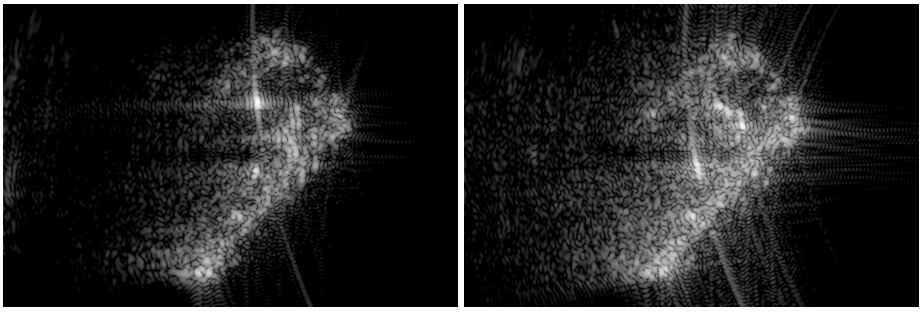
(a) 0° (b) 125°

(c) 180° (d) 280°

Fig. 2.15: Impact of the target's orientation on the SAR images.

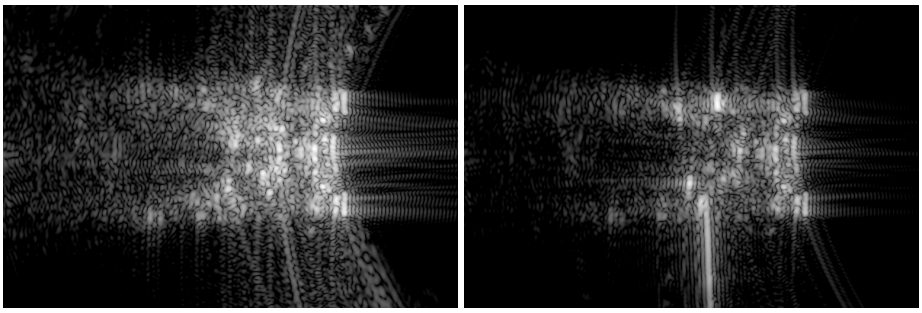
all the environmental parameters are constant. For each aspect angle, only the scatterers visible to the radar with a high enough amplitude will be included in the SAR image. Fig. 2.15 (b) shows the ISAR image of the T64 target relative to an aspect angle of 125° obtained by processing the echo signals from 115° to 135° aspect angle. Results show that the gun is only visible around 125° when the gun is nearly perpendicular to the radar and disappears in Figs.2.15 (a), 2.15 (c) and 2.15 (d) even though all the environmental conditions remained the same.

Depression angle The representation of scatterers in the produced ISAR image can be different with various depression angle. In the most extreme cases, it can be seen at a certain depression angle and disappear at another. Fig. 2.16 shows the ISAR images of the T64 with the same orientation (135°) and the same target configuration (turret at $-45°$ and gun up) at different depression angles. The location of most of the brightest scatterers is changed with the various depression angles as the reflection direction of the scatterers changes and can stop being in the direction of the radar.



(a) 19° (b) 23°

Fig. 2.16: Impact of the depression angle on the SAR images.



(a) −45° (b) 90°

Fig. 2.17: Impact of the target's configuration on the SAR images. The turret orientation is varied while the other parameters remained the same.

Target configuration changes Fig. 2.17 images from the T64 at the same aspect angle but with a different turret position. Results show that the signature changes as a function of the position of turret, and this should be taken into account when ATR is performed.

2.6.2 Setting up the database for SAR ATR

Image sequences description

The total dataset consists of 24 different sequences, of which 12 are used for training and 12 for the testing. These are made out of 3 different targets with different depression angles and configurations. Each sequence is composed of 72 images of the target obtained with a 20° integration angle every 5° to cover 360° of possible orientations for a total of 864 $(72 * 12)$ images. Each sequence represents thus one unique target with different

aspect angle but the same depression angle and target configuration. There are 1728 images in total in this dataset. Tables 2.9 and 2.10 summarise the training and testing set separation. It can be noted that the targets have different turret angles, different laboratory backgrounds and different radar heights and ranges (depression angle of 21.8° to 23.4° for training and 17.5° to 20.3° for testing) between the training and testing in order to have the maximal variety in terms of environmental variables and the most challenging dataset to classify. Details on all sequences can be found in Appendix A.

Sequence nb	Target	Radar height	Radar range	Turret Orientation	Gun Orientation	Laboratory Background
49	BMP1	1.72 m	4.45 m	45°	down	3
50	BMP1	1.72 m	4.45 m	−45°	up	3
51	BMP1	1.72 m	4.45 m	−90°	down	3
52	BMP1	1.72 m	4.45 m	90°	up	3
53	T72	1.72 m	4.39 m	−90°	down	3
54	T72	1.72 m	4.39 m	90°	up	3
55	T72	1.72 m	4.39 m	45°	down	3
56	T72	1.72 m	4.39 m	−45°	up	3
63	T64	1.72 m	4.33 m	−90°	down	3
64	T64	1.72 m	4.33 m	−45°	up	3
65	T64	1.72 m	4.33 m	90°	up	3
66	T64	1.72 m	4.33 m	45°	down	3

Table 2.9: Sequences of images used for training.

Separation between the training and testing sets

Extended operating conditions have been used to produce a diverse dataset in terms of target types, orientation, configuration and depression angle. This variety of conditions will be used to minimise the degree of correlation between the training set and the testing set. The sequences used for training and testing must be acquired differently to introduce some environmental variability to test ATR method against realistic changes in SAR data. The variation between training and testing are summarised in Table 2.11.

Sequence nb	Target	Radar height	Radar range	Turret Orientation	Gun Orientation	Laboratory Background
27	BMP1	1.54 m	4.65 m	-30°	down	2
28	BMP1	1.54 m	4.65 m	30°	up	2
29	BMP1	1.63 m	4.7 m	-30°	up	2
30	BMP1	1.63 m	4.7 m	-45°	down	2
21	T72	1.54 m	5.08 m	30°	up	1
22	T72	1.54 m	5.08 m	0°	down	1
23	T72	1.63 m	5.09 m	-30°	down	1
24	T72	1.63 m	5.09 m	0°	up	1
9	T64	1.54 m	5.10 m	30°	up	1
10	T64	1.54 m	5.10 m	0°	down	1
15	T64	1.63 m	5.12 m	30°	up	1
16	T64	1.63 m	5.12 m	0°	down	1

Table 2.10: Sequences of images used for testing.

Variable	Training	Testing
Depression angle	21.8°-23.4°	17.5°-20.3°
Laboratory background	{3}	{1;2}
Turret orientation	{-90;-45;45;90}	{-30;0;30}, One testing sequence has a -45° turret orientation.

Table 2.11: Environmental variable differences between the training and testing set.

2.6.3 Nearest neighbour classification

Nearest neighbour classification method

The same NN method as in Section 2.5.1 is applied to the MGTD. The same parameters as for the MSTAR are chosen. However, as the target occupies a larger portion of the image than in MSTAR images, the size of the mask is increased from 34×34 to 44×44 so that the target is effectively hidden.

Results of the NN method on full SAR images and partially masked SAR images

The classification scores obtained seems to have less correlation than the MSTAR SOC dataset. It is closer to that of the EOC datasets. However, this is just an indication and is

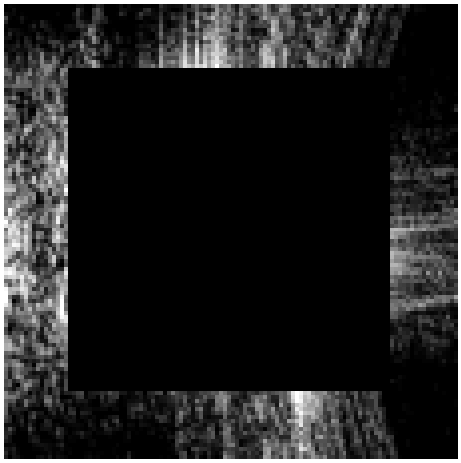


Fig. 2.18: SAR image with the target masked.

Full image	71,80%
Target hidden image	56,50%
Random classification	33,33%

Table 2.12: Classification results using the NN method on full SAR images and SAR images with a hidden target on the MGTD with targets in a fixed orientation, so that he target is hidden by the black area at all times (facing the right of the image).

not an objective way to measure correlation as many other parameters are different from the MSTAR database to the MGTD and could interfere with those scores. The number of target is different and the measurement conditions are extremely different as the MSTAR dataset is composed of SAR images taken from a field with a plane whereas the MGTD is composed of ISAR images acquired in a laboratory.

2.7 Guidelines for the performance quantification of a SAR ATR method on the MGTD

2.7.1 Correct classification rate

The performance evaluation only deals with target recognition. The recognition step consists in correctly labelling targets as defined in 2.6.1. Detection is out of the scope as the images are already focused on the targets in both the MSTAR dataset and the MGTD.

ATR methods performance are measured using the probability of correct classification P_{cc}. This is the ratio between the number of correctly labelled targets over the total number of targets. If a rejection class is used when the ATR is not able to label the target, the probability of false positive P_{fp} is also analysed. This represents the number of wrongly labelled targets over the total number of target tested. If the target is labelled as the rejection class, it is neither correctly, nor wrongly classified. The presence of a rejection class allows a reduction of false positives. These scores should be given when using the dataset separation presented in Section 2.6.2.

2.8 Conclusion

In this chapter, the MSTAR datasets are presented and the background correlation that is present between the training and testing sets is quantified. A new dataset of ISAR images is presented that can facilitate the evaluation and comparison of SAR ATR algorithms. The choice of parameters and the process leading to the acquisition of the images are explained. The operating conditions changed throughout the whole dataset to introduce variability in the dataset are also described. For each target type, different orientations, configurations and depression angles are used for each sequence as well as different laboratory background between the training and testing set. Guidelines are also provided to test fairly SAR ATR algorithms. A suggestion of separation between the training and testing set is given. Standard indicators are recommended for a baseline evaluation that should help for future comparisons of ATR methods.

Chapter 3

SAR image classification theory

Contents

The approach to the SAR ATR problem in the following work is essentially centred on methods issued from the computer vision and machine learning domains. In this Chapter, the principle of classification is explained with background on feature-based and deep learning classification methods. Some background theory is also given on the Hough transform for line detection that will be used to determine the target orientation.

3.1 Classification

Classification or target recognition in computer vision consists in the task of assigning a label to the corresponding object present in an image. In this thesis, the focus is on the classification task of targets in SAR database. For the currently used SAR ATR databases, a loose localisation of the target has already been done by a pre screener [19]. The target

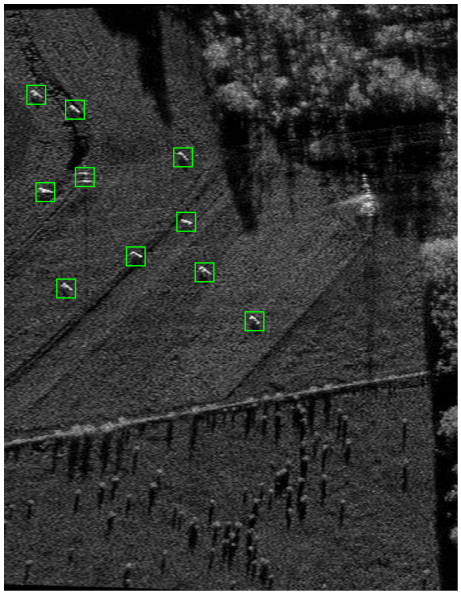


Fig. 3.1: Full SAR image prior any analysis with the targets highlighted [12].

has a probability of detection $\mathcal{P}_D$ to be noticed. The original SAR image seen in Fig. 3.1 is thus reduced to several images centred around a single target detected by the pre-screener. A successful classification is defined when the correct label is associated with the target. The target is classified with a certain probability of correct classification $\mathcal{P}_{CC}$ by the evaluated method. It is assumed that the detection step was carried out successfully for the classification to happen. Otherwise, if a target is not detected, it cannot be fed to the classifier. A false positive, or empty image, could be handled by some classifiers if they are trained for this particular case, but that is not taken into account for the $\mathcal{P}_{CC}$ computation. Only the first most probable guess is taken into account on the contrary to other databases having much more possible classes and thus assuming the object correctly classified if its class is in the top 5 label suggestions. In addition to the probability of correct classification, a confusion matrix is provided that is able to clarify which targets are the most easily confused with another target for the tested classification algorithm. This matrix is a summary of the assigned class regarding to the true class of the target classified. In case of a perfect score, the confusion matrix will be a diagonal matrix as all assigned classes correspond to the true classes.

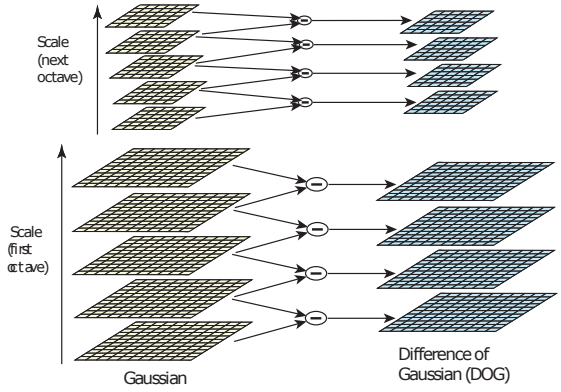
3.1.1 Feature classification

Features description

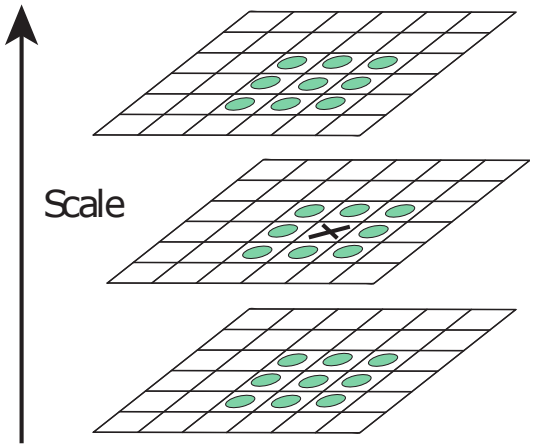
Features are used to characterise points of interests. They are specific to a location and each has a specific way to describe the surroundings of this particular location. By comparing features, it is possible to know if two areas look similar in different images. In this section, specific features will be described in detail. Gradient based features computes intensity gradients to describe the area of interest: The Scale Invariant Feature Transform (SIFT) [27] and Speeded Up Robust Features (SURF) [28] are gradient based features. Binary features that compare intensities between pairs of points are also presented: Oriented FAST and Rotated BRIEF (ORB) [29], Fast REtina Keypoint (FREAK) [30] and Binary Robust Invariant Scalable Keypoints (BRISK) [31].

Gradient based descriptors

SIFT Scale Invariant Feature Transform (SIFT) is a feature which is both scale and rotation invariant [27, 32]. The scale adaptability is achieved through analysing features of the image at various scale as shown in Fig. 3.2 (a). Each octave contains the image scaled at a unique size. SIFT uses an approximation of the Laplacian of Gaussians (LoG) which is sensitive to intensity gradients but computationally costly because it requires differentiating. The approximation consists in Differences of Gaussians (DoG) which only require the application of a Gaussian filter and a subtraction of the most blurred image to the least blurred image. Inside an octave corresponding to a unique image size, the same image is blurred using a Gaussian filter with an increasing standard deviation σ. In each octave, slow and fast change of gradients can be evaluated using the DoG on images more or less blurred. The various octaves serve to detect features of various size with the smallest images, detecting the biggest features. The researched keypoints are the ones located on extrema of the DoG image obtained even though not all extrema are kept after investigation. The final keypoint descriptor consists in the description of



(a) DoG computation [27].



(b) Keypoint neighbours [27].

Fig. 3.2: Comparison of the segmentation with and without the evolution of the GMM background model [27].

the keypoint's neighbourhood as seen in Fig. 3.2 (b) represented by a vector. Once the keypoint located, the neighbourhood of the keypoint is divided in 16 sub-areas of size 4×4. Each of these zones are described by concatenating a description of the gradient of the sub-area in 8 directions.

SIFT also achieves rotation invariance. It starts by determining the orientation of the gradient found by the DoG. For each keypoint, an histogram containing bins of 10° registers the intensity of the gradient in the keypoint neighbourhood for each bin direction. Once the histogram acquired, the vector descriptor is shifted to begin its description in the direction of the steepest gradients.

where $G(x, y, \sigma)$ is a Gaussian kernel, s represent the scale space, k is a multiplicative

Algorithm 1: Detection of SIFT features

for *n=1 to 4* **do**
  Computation of the image in the n^{th} octave: resize the original image by $\frac{1}{n}$.
  for *s=1 to 5* **do**
    Exploration of the scale space with the convolution of a Gaussian kernel with the image from the n^{th} octave: $L(x,y,k\sigma) = G(x,y,\sigma) * I(x,y)$
  end
  for *s=2 to 5* **do**
    Computation of the difference of Gaussians: $D(x,y,\sigma) = L(x,y,k\sigma) - L(x,y,\sigma)$
    Detection of the stable keypoints in the image at different scales with the localisation of local extrema by comparing points to their 8 current neighbours and 9 neighbours in the previous and next scale as in Fig. 3.2 (b). The point is selected if all neighbours have either a bigger or smaller value.
    Evaluation of the detected points and removal of the points with a too low contrast. The scale-space function D is expanded using a Taylor expansion to the order 2. The position of the real extremum $\hat{X} = (x,\hat{y},\sigma)$ is obtained by setting to 0 the derivative of D. Points that do not respect the inequality $\mid D(\hat{X}) \mid > 0.03$ are removed.
    Points poorly localised along an edge are removed. To do so, the Hessian matrix H of D is computed at the location of the keypoint. Points that do not respect the inequality $\frac{\mathrm{Tr}(H)^2}{\det(H)} < \frac{(r+1)^2}{r}$ are removed.
  end
end

constant to go from one scale s to the next with a value of $2^{\frac{1}{s}}$, r is a real number set to 10 in the original paper.

Algorithm 2: Description of SIFT features

for *each remaining detected keypoint* **do**
- The keypoint is located at (x,y). The smoothed image with the closest scale as during detection will be used for the rest of the algorithm.
- The gradient magnitude m and orientation θ are computed for sample points within a region of the keypoint.
  $m(x,y) = \sqrt{(L(x+1,y)-L(x-1,y))^2 + (L(x,y+1)-L(x,y-1))^2}$ and
  $\theta(x,y) = \arctan \frac{L(x,y+1)-L(x,y-1)}{(L(x+1,y)-L(x-1,y)}$
- Add the various gradients in an orientation histogram with 36 bins (for 360°). The weight of each gradient contribution in the histogram depends on the gradient magnitude and a Gaussian circular window with a standard deviation of $1.5k\sigma$.
- The highest peak in the histogram and all peaks over 80% of the highest peak are all selected to be separate final keypoints. Their orientations are retained.

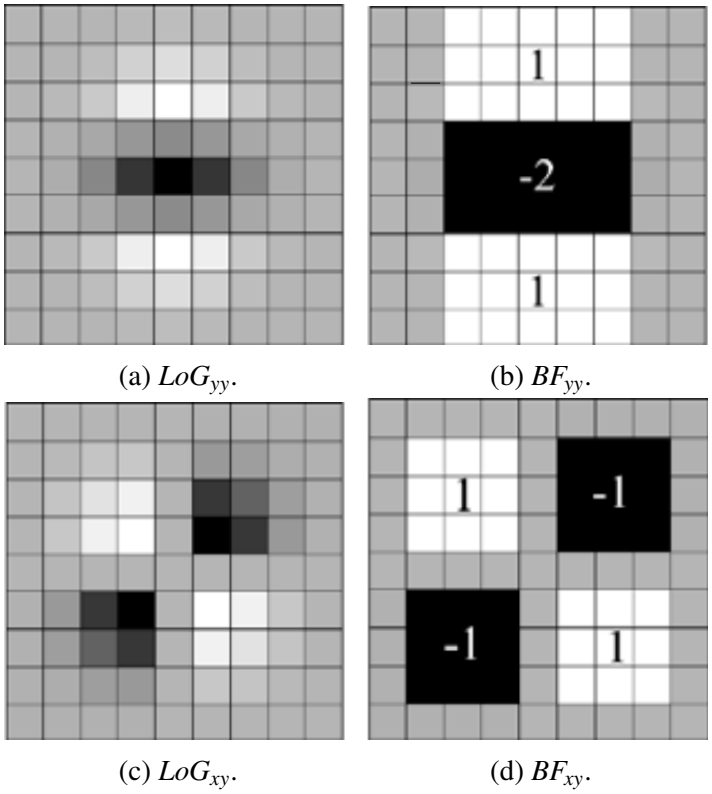
end

for *each final oriented keypoint* **do**
- Rotate the descriptor's coordinates and the gradient orientations relatively to the main orientation of the keypoint.
- The gradients magnitudes and orientations of sample points surrounding the keypoint are accumulated in several histograms, with only 8 bins this time. Gradients are still weighed using magnitude and the Gaussian window. Each histogram corresponds to a part of the surroundings relative to the keypoint.
- Concatenate the histograms in a 4×4 descriptor matrix.
- Use trilinear interpolation to smooth the descriptor.
- Reindex the matrix in a vector.
- Threshold high contributing gradients and normalise the vector to compensate for brightness changes.

end

SURF The SURF feature is based on the same principle as SIFT but simplifies and accelerates the process greatly at the expense of a loss in precision [28]. The DoG from SIFT are replaced by simpler Box Filters (BF). BF consist in a linear combination of sums of the intensity of pixels in diverse areas. One example is the Fig. 3.3 (b) in which the sum of intensities in the black area in the original image is subtracted two times to the sum of intensities in the white areas. BF have the advantage of being quickly computed with integral images. Integral images are images in which the intensity of each pixel consists in the sum of all pixels in the original image that have a smaller abscissa or ordinate. BF consist then in a linear combination of few intensities of two intensity maps. The LoG

equivalent is shown in Fig. 3.3 (a) and requires a much more costly differentiation. The DoG is less computationally expensive than the LoG but still require the application of a Gaussian filter on the image. By using BF, points of interest having strong gradients are quickly found and can be further investigated.



(a) LoG_{yy}. (b) BF_{yy}.

(c) LoG_{xy}. (d) BF_{xy}.

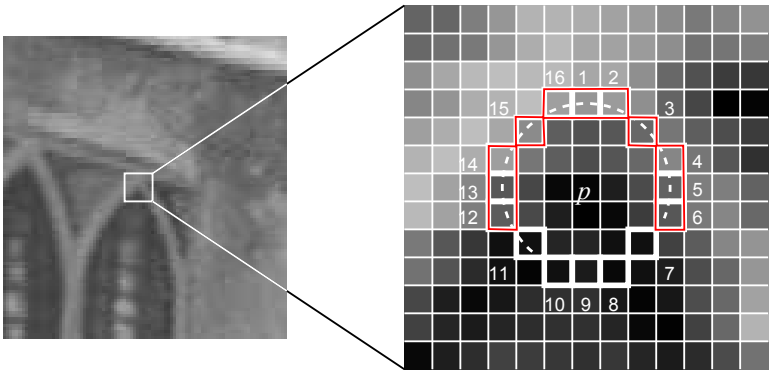
Fig. 3.3: Comparison of the LoG and BF complexity in the computation of the SURF descriptor [28].

The orientation of the feature is computed by combining the strength of responses of the keypoint's neighbourhood to both a vertical and a horizontal wavelet. The vector describing the keypoints founds consist in the response of several regions around the keypoint to the wavelets. This vector is shifted according to the orientation previously found to be rotation invariant.

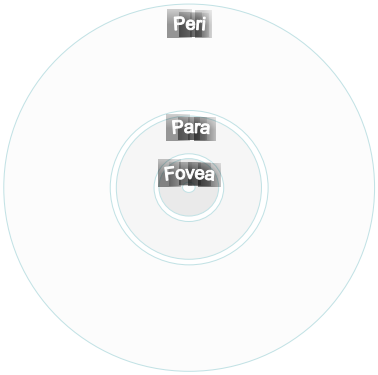
Binary descriptors

In order to be even faster, binary descriptors that do not use gradients but only the comparison of intensities of several pixels around a keypoint have been created. The result of

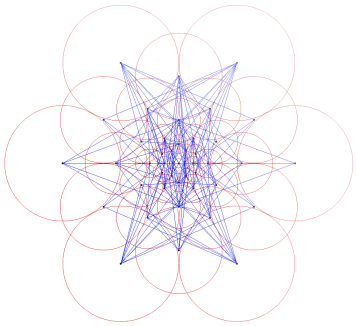
these comparisons gives a binary vector. Each binary number is the result of comparisons between two pixel intensities. These binary vectors can be compared one with each other quickly using the Hamming distance.



(a) FAST corner detection [33].



(b) FREAK retina pattern [30].



(c) Comparison of pixel pairs.

Fig. 3.4: FREAK Principle [30].

FREAK FREAK as SIFT use images from different octaves, to be scale invariant [30]. The detection in FREAK is a more complex version of the detection in Features from Accelerated Segment Test (FAST) [33]. In FAST, the points of interest are corners defined by a certain amount of consecutive surrounding pixels brighter than the centre pixel as in Fig. 3.4 (a). The FAST was improved in Adaptive and Generic Accelerated Segment Test (AGAST). The corner detection does not have to be trained to fit some specific images as the number of consecutive pixels are image dependant [34]. AGAST consists in a decision tree to find the points that qualify as corners. AGAST is adapted in FREAK to a multi-scale detection using octaves as SIFT did in Section 3.1.1. Once the keypoint

located, pairs of pixels around the keypoints are chosen relatively to a pattern based on the human retina as shown in Fig. 3.4 (b). The intensities of the pairs or pixels are compared two-by-two as shown in Fig. 3.4 (c). 512 pairs are compared and result in a binary vector dependant on which pixel was the brightest in each comparison. The choice of the pairs results from a learning phase which goal was to maximise the variance between the points compared.

As for the previous descriptors, the orientation is compensated by shifting the descriptor after determining the strongest orientation of the feature. The orientation is computed using the local gradients over long pairs specially selected in the pool of all pairs. The innovation in FREAK is in the retina pattern and the imitation of the saccadic search during the matching phase of the features. The saccadic search is the eye movements that happen when the eye compares two objects to see if they are the same. The eye moves a lot during the process in order to compare the objects not in its entirety but bits by bits. The adaptation of the saccadic search in this algorithm replaces the more classical feature matching seen in Section 15. The algorithm consists in a cascade search in which the first 16 bytes of the FREAK descriptors are compared. According to the result of this first comparison, the rest of the descriptors are compared or the match is directly rejected. This saves time during the comparison as if the beginnings of the descriptors are not similar, the rest of the descriptor will not be compared.

BRISK Binary Robust Invariant Scalable Keypoints (BRISK) use a different pattern than FREAK as shown in Fig. 3.5[31]. The detection is however the same as for FREAK. BRISK features are close to FREAK features except in the choice of the pairs compared that are shorter and on a different pattern. The feature matching is standard and does not use saccadic search.

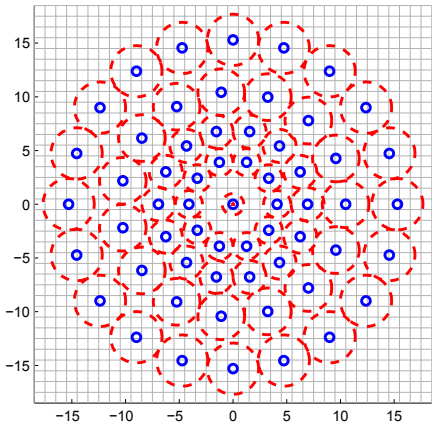


Fig. 3.5: BRISK pattern [31].

Algorithm 3: Detection of BRISK features

for *n=1 to 4* **do**
  Computation of the image c_n specific to the n^{th} octave. Resize the original image by $\frac{1}{n}$.
  for *s=0 to 3* **do**
    Down sampling of the image c_n in an image d_s by sub-sampling it by a factor $2^s \cdot 1.5$
    Apply the FAST 9-16 detector to the image of scale s. In a circle of 16 points around the studied point, at least 9 contiguous points must have an intensity higher than $I+t$ or lower than $I-t$ with I the intensity of the studied point and t a selected threshold. The maximum threshold for which this is true is retained as the FAST score.
    Verify that the 8 neighbours of the point have a lower intensity.
  end
end
for *each detected keypoint* **do**
  Verify that the neighbouring points of the detected keypoint in the current, higher and lower scale have all a lower intensity than the keypoint.
  The 3 FAST scores obtained on the 3 scales studied are considered continuous in the scale-space. These scores are refined across scale by fitting a 2D quadratic function on the FAST scores. A 1D parabola is then fitted to the quadratic function with its maximum giving the true scale of the keypoint.
  The former coordinates of the image d_s are interpolated to be the closest to the sub-sampling corresponding to the true scale of the keypoint.
end

Algorithm 4: Description of BRISK features

for *each keypoint* **do**
  for *each point* $\mathbf{p_i}$*, in the pattern in Fig. 3.5 surrounding the keypoint* **do**
    Apply Gaussian smoothing at a radius σ_i corresponding to distance from the red dashes to the sampled point at the centre of the blue point.
    The intensity of the sampled point is then $I(\mathbf{p_i}, \sigma_i)$.
  end
  for *each long pair of points* $\mathbf{p_i}$, $\mathbf{p_j} \in L$ **do**
    Estimate the local gradient $\mathbf{g}(\mathbf{p_i}, \mathbf{p_j}) = (\mathbf{p_j} - \mathbf{p_i}) \cdot \frac{I(\mathbf{p_i}, \sigma_i) - I(\mathbf{p_j}, \sigma_j)}{\|\mathbf{p_j} - \mathbf{p_i}\|^2}$
  end
  Estimate the overall pattern direction of the studied keypoint:

$$\mathbf{g} = \begin{pmatrix} g_x \\ g_y \end{pmatrix} = \frac{1}{L} \cdot \sum_{\mathbf{p_i}, \mathbf{p_j} \in L} \mathbf{g}(\mathbf{p_i}, \mathbf{p_j})$$

  Rotate the sampling pattern by $\alpha = \arctan 2(g_y, g_x)$
  for *each short pair of points* $\mathbf{p_i}$, $\mathbf{p_j} \in S$ **do**
    Perform an intensity comparison: $b = \begin{cases} 1 & \text{if } I(\mathbf{p_j^\alpha}, \sigma_j) > I(\mathbf{p_i^\alpha}, \sigma_i) \\ 0 & \text{otherwise.} \end{cases}$
  end
  Concatenate the result of all comparisons.
end

ORB The keypoint detection in the Oriented FAST and Rotated BRIEF (ORB) keypoints is an improved FAST, with the introduction of an orientation detection in addition to the keypoint detection. The direction of the keypoint is determined by computing the centroid of the area around the keypoint. The descriptor consists in a rotated Binary Robust Independent Elementary Features (BRIEF). BRIEF results of the training of decision trees that compares the brightness between pairs of pixels. The principle is similar to the other binary descriptors but the choice of pairs results from training. The direction in which the BRIEF descriptor is calculated is the direction found by the improved FAST detection.

Description of a standard feature classification method based on matching

In the previous sections, the detection and the description of keypoints for various model are explained. The resulting keypoint consists of a location in an image (x,y) and a descriptor summed up in a vector. The objective during feature matching is to be able to see what keypoints are similar enough so that they could describe the same part of the object in different images. Classification is determined according to the result of feature matching. A basic workflow for feature-based classification an object can be seen in Fig. 3.6.

Once the keypoints are computed, the first step is to match the keypoints found in the studied image with some previously computed keypoints from training images. The

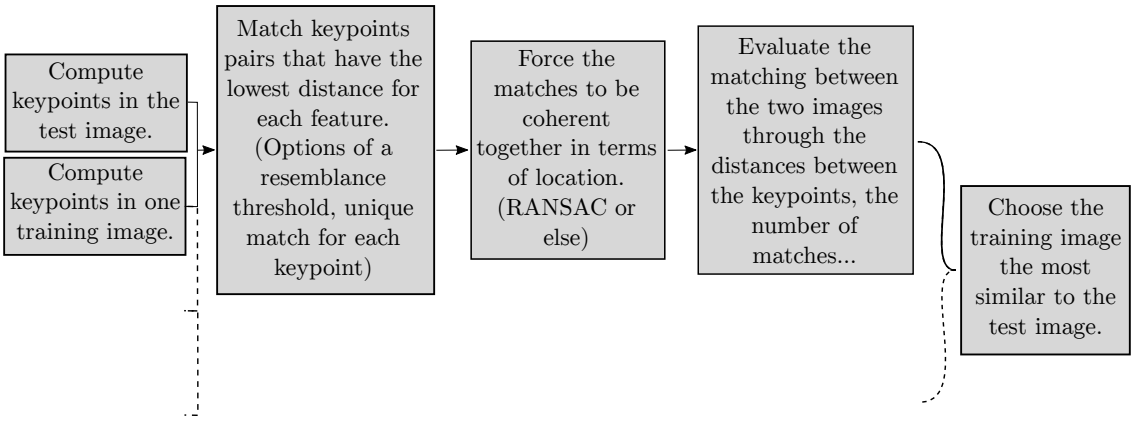


Fig. 3.6: Standard pipeline of classification based on feature matching.

bruteforce method consists in calculating the distance between each keypoint descriptor of the test image with all the keypoint descriptors in the training image. The keypoint studied in the test image is matched is with the keypoint of the training image that result in the lowest distance. Several conditions can be added to prevent wrong matches. The matches can be forced to be unique and the training keypoints can be associated with only one keypoint of the test image. The quality of the match can also be forced to be of a high standard as in Lowe method in which the distance for validating a match should be at most a defined fraction of the distance of a second possible match [27]. The second step is not mandatory but helps having better results. It consists in using Random Sample Consensus (RANSAC) which forces the matches to be coherent with the target movement between two images [35]. To find the most probable homography, RANSAC compares the number of outliers for the possible homographies issued from the feature matching between the keypoints location in the studied image and their matched keypoints in the training images. Outliers that are not in line with the homography are removed from the match list. The resulting match between the testing and training image can then be evaluated. Several criteria can be used such as the number of keypoints matched or the average distance between the testing and training keypoints. The target in the test image is then associated with the class of the training image deemed as best overall match.

Feature matching is not the only way to use features for classification. More advanced classification tools can be used such as machine learning methods such as Support Vector Machine (SVM). In any case, if a technique is based on a certain feature, the quality the target description by this feature is essential to avoid misclassification. In Section 4.4.1 a comparison is proposed between the various features presented in this section applied to SAR ATR.

3.1.2 Supervised learning

Problem formulation

The classification problem with specific labels falls under the category of supervised learning problems and can be summed up to the need to learn the function f mapping the input image space $\mathcal{X}$ to the label space $\mathcal{Y}$ such as $f : \mathcal{X} \to \mathcal{Y}$.

Objective and loss function Classification datasets provide several examples corresponding to the expected mapping $\{(x_i, y_i) \subset \mathcal{X} \times \mathcal{Y}, \forall i \in [\![1, n]\!]\}$. These n examples should be independent. Instead of trying to construct directly f, supervised methods use the provided examples to make the mapping function f learn the $\mathcal{X} \to \mathcal{Y}$ mapping. The goal is to find the mapping function f^* among the hypothesis space $\mathcal{F}$ making the least mistakes. The effectiveness of the mapping function is quantified with an objective function. The objective function is constituted by expected losses of f over the training samples, as it cannot be measured over the whole $\mathcal{X}$ space. The loss for each example i is expressed according to the difference between the true output y_i and the estimated output $f(x_i)$ such as $\mathcal{L}(f(x_i), y_i)$. Thus, the search of the best suited mapping function f^* can be expressed as an optimisation problem as in Eq. (3.1). In this case, it is assumed that a higher loss meant more errors, hence the argmin.

$$f^* = \underset{f \in \mathcal{F}}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}(f(x_i), y_i) \tag{3.1}$$

In the case of classification deep neural network, the most common loss function is the cross-entropy loss function. The result of the deep neural network for each input is, using a soft-max layer, a $1 \times k$ vector representing the probability of the input belonging to each class, k being the number of possible classes. The objective being to have a probability of 1 for the correct class and 0 for all other classes. The cross-entropy function is expressed

as in Eq. (3.2) for a sample x_i of class c. The y_i output element is defined as the reference $1 \times k$ vector of class probability defined as $y_i(j) = \delta_{j,c}, \forall j \in [\![1,k]\!]$. $\widehat{y_i}$ is the estimation of y_i by f.

$$\mathcal{L}(\hat{y}_i, y_i) = \sum_{j=1}^{k} -y_i(j) \log(\widehat{y_i(j)}) = -\log(\widehat{y_i(c)}) \tag{3.2}$$

Overfitting and regularisation However, the function f^* in Eq. (3.1) could very well map correctly only the training samples and get all other input of the $\mathcal{X}$ space wrong whereas another f function would have made more mistakes on the training samples but less in the overall $\mathcal{X}$ space. In that case, the function f^* would still be chosen even though it clearly overfitted to the training samples because the result of its objective function would be lower. For target recognition, overfitting would mean that the algorithm can achieve high scores in the training set but is not able to generalise the learned information and performs a lot worse on the testing set. Overfitting can be seen for example when a training set is not diverse enough. In order for the mapping function chosen to be better generalised, the objective function is altered with the addition of a regularisation term $R(f)$ independent from the samples to prioritise simpler functions, regardless of their performances to prevent overfitting. Indeed, it will then be harder for the model to adapt to the specifics of the training samples over generic $\mathcal{X}$ samples. The new objective function can be expressed as in Eq. (3.3).

$$f^* = \underset{f \in \mathcal{F}}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}(f(x_i), y_i) \quad + R(f) \tag{3.3}$$

In the case of deep neural network, the regularisation is often a weight decay which

is a shrinkage method. The goal of this method is to penalise the networks with big weights. If the network representing f has m layers with weights, the regularisation term can be expressed as in Eq. (3.4) with λ chosen by the user. The weights vectors for each concerned layer is expressed as $W_i, \forall i \in [\![1, m]\!]$.

$$R(f) = \frac{\lambda}{2} \sum_{i=1}^{m} W_i^\intercal W_i = \frac{\lambda}{2} \sum_{i=1}^{m} \|W_i\|_2^2 \tag{3.4}$$

Optimisation problem Once the structure of the classifier fixed, the search for its optimal version in function of its parameters can be expressed in Eq. (3.5). The vector of parameters to optimise Ψ gathers all the alterable parameters of the fixed function f. That corresponds to the weights and biases in the case of a neural network.

$$\begin{cases} \Psi^* = \operatorname{argmin} g(\Psi) \\ g(\Psi) = \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}(f_\Psi(x_i), y_i) \ \ + R(f_\Psi) \end{cases} \tag{3.5}$$

Organised search for the best vector of parameters Ψ : the Stochastic Gradient Descent with Momentum (SGDM) The values of the objective function are unknown for all possible parameters as parameters are numerous (especially in the case of deep learning methods, for which the number of parameters can reach millions or even billions) and the needed values of the objective function have to be computed for each case. A random search of the parameters is thus very unlikely to provide a good solution.

There are several optimisation methods to try to approach the best values of Ψ but the focus will be on the most commonly used method for deep learning for images which is the SGDM. In order to search better for an optimal solution, the functions investigated g

are restricted to only the differentiable functions. This allows the study of the objective function trends according to changes in the vector Ψ.

The principle of the Stochastic Gradient Descent (SGD) is to firstly compute an approximation of the gradient of the objective function in function of the parameter vector Ψ. Using this estimated derivative, it is possible to update Ψ in order to go in the direction of a minimisation of the objective function. The computation of the approximated objective function differenciation using backpropagation is explained in the next section. The update of the internal parameters of the network is done after a certain number of images have been through the network. This number of images is called the batch size. This is done usually several times until all images present in the training set have been through the network once. This is called an epoch of training. The process is repeated for a number of epochs until either the network is deemed satisfactory or a chosen number of maximum epochs has been reached.

The SGD is based on the Taylor expansion of the objective function as seen in Eq. (3.6) with h being a small variation added to the parameter vector Ψ.

$$g(\Psi + h) = \sum_{n=0}^{\infty} h^n \frac{g^{(n)}(\Psi)}{n!} \tag{3.6}$$

$$g(\Psi + h) \approx g(\Psi) + h \cdot g'(\Psi) \tag{3.7}$$

The goal is to choose h to update Ψ so that the value of the objective function is lowered. The search direction h is chosen to be equal to $-\lambda.g'(\Psi)$. The reason of the negative sign is that the objective function has to be minimised. $g'(\Psi)$ gives the correct update direction for each parameter composing the vector Ψ. λ is a constant called the learning rate and should be carefully chosen. If lambda is too big, the optimisation will not be able to converge and could even diverge. However, if it is too small, the convergence could take very long and the optimisation could even fall in a local minimum. The updated

objective function using the SGD can thus be written as in Eq. (3.8).

$$h = -\lambda \cdot g'(\Psi) \tag{3.8}$$

$$g(\Psi + h) \approx g(\Psi) - \lambda g'(\Psi)^{\intercal} \cdot g'(\Psi) \tag{3.9}$$

An improvement on this algorithm is to include a momentum term mimicking Newton's second law $F = ma$ where F is the sum of forces applied to the object, m is the object weight and a is its acceleration. The change of speed represented by the acceleration is not only dependant on the forces exerted but also the object mass. The higher the mass, the lower will be the acceleration and the least the velocity of the object will be affected. In the case of the SGDM, the position of the object is the parameter vector Ψ and the position update h, is changed so that it takes into account the previous search direction. This converges faster in practice. An intermediary variable v is introduced that keeps a decaying history of the previous search directions. The influence of the earlier search directions is less influential than the recent one. The new parameter update h can be expressed as in Eq. (3.10). v is initialised at 0 and updated at each iteration.

$$v = \alpha v + g'(\Psi) \qquad \alpha \in [0, 1[\tag{3.10}$$

$$h = -\lambda v \tag{3.11}$$

The whole SGDM method is summed up in the following algorithm:

Backpropagation The update of the weights of the neural network with the SGDM method requires an estimation of the differentiation of the objective function against the inputs of the neural network. The backpropagation is based on the differentiation chain

Algorithm 5: SGDM

Initialise g with Ψ_0;
Initialise the system's update speed $v = 0$;
Choose λ, α;
for *N iterations* **do**
    Choose m samples from the training data;
    Compute the objective function's result $g(\Psi)$;
    Compute the gradient $g'(\Psi)$ with backpropagation;
    Compute the new velocity $v := \alpha v + g'(\Psi)$;
    Compute the parameter step $h = -\lambda v$;
    Update the parameter vector $\Psi := \Psi + h$;
end

rule. Each layer has a function for the forward pass. A function for the backward pass is then defined as the differentiation of the forward pass function. Using the backward passes, the result of the objective function is cascaded backwards the network, computing gradually the contribution of each input. This is done using each differentiable terms issued from the chain rule and the activation terms computed during the forward pass.

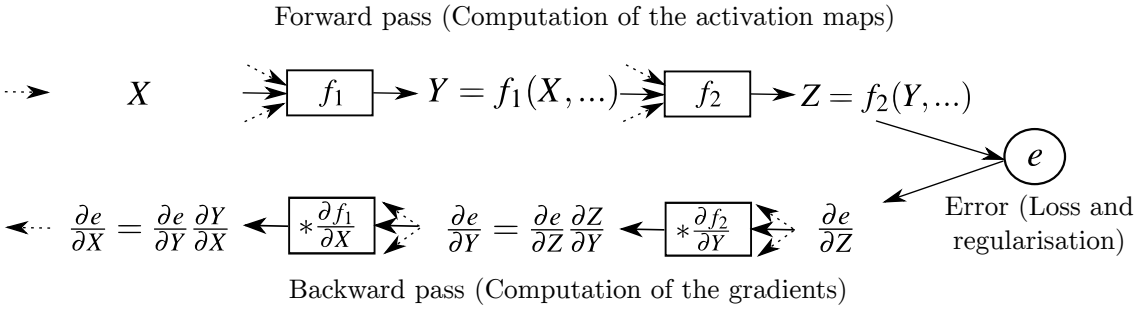


Fig. 3.7: Backpropagation principle using the chain rule.

The neural network is composed of several layers, corresponding to functions that are applied on after the other on the input. In the forward pass, the original input is transformed by the layers' functions to activations, the intermediary stages, until the last layer. The final result of the neural network is compared to the groundtruth and an error is computed. This error is the result of the objective function detailed in the previous section with a loss and regularisation term. The role of the backpropagation is to determine which parameters of the neural network had the biggest influence on the error so that these parameters can be updated and the result of the objective function reduced. In practice,

the objective function is partially differentiated according to each parameter of the neural network. X, Y, Z are defined in Fig. 3.7 as some activations of the neural network with an illustration of the forward and backward pass.

The computation of these activations requires to know the activations of the previous layer, the parameters of the neural network involved and the function applied in this layer. A common activation a can be written as in Eq. (3.12) using activations a_j from the previous layer, and the appropriate neural network's weights $\{w_j, j \in [1,n]\}$.

$$a = \sum_{j=1}^{n} w_j \cdot a_j \tag{3.12}$$

Using the chain rule, it is possible to differentiate the objective function against the activations investigated as in Fig. 3.7 and get the appropriate $\frac{\partial e}{\partial a}$. Using once more the chain rule, the individual influence of each weight w_j to the result of the objective function e is determined in Eq. (3.13) from the previously obtained partial differentiation of the error against the activation.

$$\frac{\partial e}{\partial w_j} = \frac{\partial e}{\partial a} \cdot \frac{\partial a}{\partial w_j} \tag{3.13}$$

$$\frac{\partial e}{\partial w_j} = \frac{\partial e}{\partial a} \cdot a_j \tag{3.14}$$

If the interest is broaden from a specific weight to the global problem of the influence of all parameters, it is the Jacobian matrix that is calculated for each activation against each parameter to get all the intermediary partial differentiations ($\frac{\partial Z}{\partial Y}$ or $\frac{\partial Y}{\partial X}$ represented in Fig. 3.7). The partial differentiation of the objective function against the neural network parameters can then be calculated using the element wise Hadamard product.

Neural network

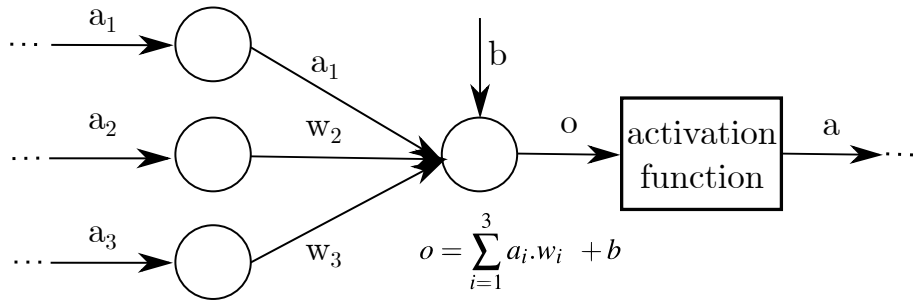


Fig. 3.8: Computation of the activation resulting from several inputs of one neuron.

The mapping function f investigated through the optimisation of the objective function in the previous section was not detailed. In this thesis it will be a Convolutional Neural Network (CNN), which is a specific case of a neural network structure. The increase of the computational capacities of the computers made possible the training of neural networks using the computationally expensive backpropagation and SGDM method. Neural networks have been efficient in several field as they are able to model data non linearly and in a complex way. They have been inspired by the biological neurons. The artificial version of the neuron is shown in Fig. 3.8. Several inputs are combined to some parameters specific to the network. The result is then passed through an activation function. The resulting activation is then passed to the concerned next neurons. The activation function introduces the capability of the neural network to produce non-linear models. Several activations functions have been proposed such as the tanh $a = \tanh(o)$, the sigmoid function $a = \frac{1}{1+e^{-o}}$ and the Rectified Linear Unit (ReLu) $a = \max(0, o)$. The complete neural network seen in Fig. 3.9 is composed of layers of neurons stacked on top and besides each other. The intermediary neuron layers are the hidden layers. A simple linear model can be deduced with a neural network with no hidden layers, and thus no activation function. The more complex the function to model, the deeper will be the neural network by stacking additional hidden layers. Some other solutions that do not rely solely on the stacking of layers exist but will not be investigated throughout this thesis.

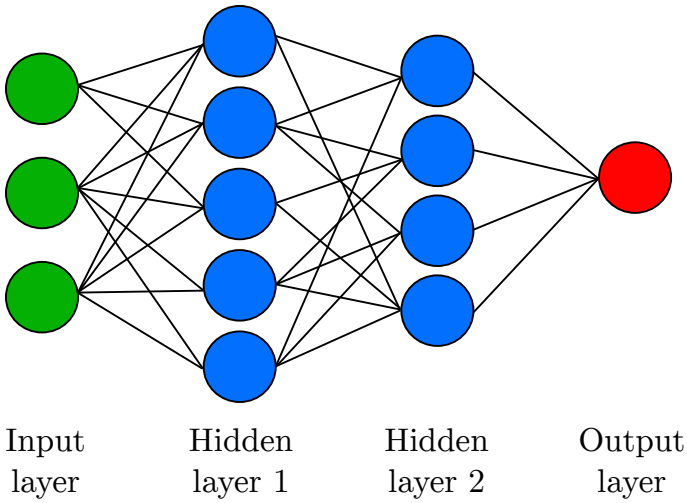


Fig. 3.9: A neural network is composed of layers of neurons where the more complex interpretation of the input is done in the hidden layers.

Convolutional Neural Networks (CNNs)

The particularity of CNNs is their ability to interpret localisation information from data [36]. When the input is a matrix, the CNN does not treat each pixel independently but the pixels that are close to each other are combined, resulting in an image interpretation taking into account the spatiality of the potential objects.

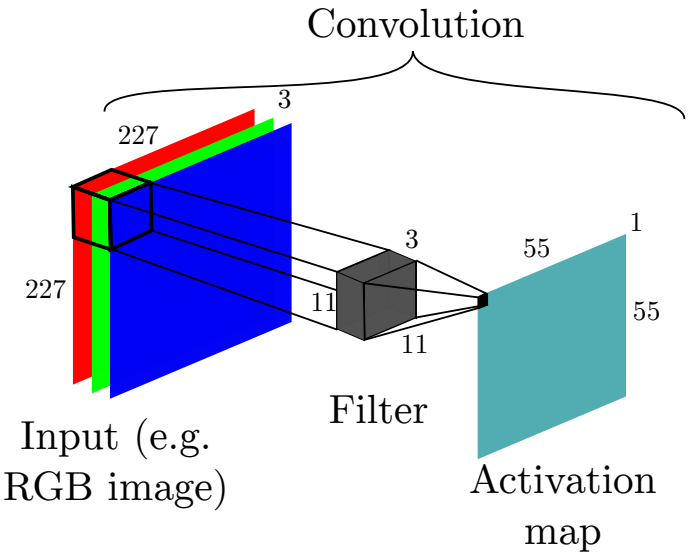


Fig. 3.10: Example of the first convolution in AlexNet.

Instead of applying the weights independently to each pixel of the input as for a standard neural network, a filter is created and is slided over and convolved with the whole

image as seen in Fig. 3.10 for the first convolution in the AlexNet network. The convolution consists in an element-wise multiplication of the intensities in the area of the image studied followed by their summation. The filter is then slided to study the neighbouring area in the image. This is done until the full input image is convolved by the filter. In the specific case represented in Fig. 3.10, the input image is $227 \times 227 \times 3$. There are in fact 96 filters (only one is represented) of size $11 \times 11 \times 3$. The results of the convolution of the 96 filters with the input image are 96 activation maps of size 55×55. Each filter will be sensitive to different type of inputs such as a gradient of colour in a certain direction but are not always interpretable. The size of the resulting activity map can be computed as in Eq. (3.15). In this equation, W_O represents the width of the output, i.e. activation map, W_I represents the width of the input and W_F is the width of the filter. P is the amount of padding done on the input image and S is the stride, i.e. the number of pixels skipped when the filter is slided.

$$W_O = \frac{W_I - W_F + 2P}{S} + 1 \tag{3.15}$$

The filters contain all the parameters Ψ that will be optimised during the training. The number of weights each filter contains is $W_F \times W_F \times D_F$ with D_F the depth of the filter which is the same as the depth of the input. It contains also 1 bias. For a full layer, the total number of parameters is obtained by multiplying the number of parameters for one filter by the total number of filters, e.g. 96 in the first layer of the AlexNet.

Other functions are present in a typical CNN structure and will be discussed in the following sections.

Pooling Pooling layers are layers without any parameters to optimise. Areas of each activation map, e.g. 2×2 square area from a specific channel, are searched for their maximum or mean. This 2×2 area is then down sampled to a unique number representing its maximum. This is a way to down sample the activation maps and hence reduce the

probability of over-fitting the data. With the provided example size, the final size of the activation map is cut by 4.

Activation - ReLU The ReLU is defined as in Eq. (3.16). This is a good alternative for the tanh and sigmoid function that suffers from the vanishing gradient problem. This problem happens when the function does not differentiate enough between different high values as it follows an asymptote. Thus, when computed, the gradient appears very low and the parameters are not updated as much as they should or even stay constant. The ReLu does not have this problem for positive activations.

$$f(x) = \begin{cases} 0 & x < 0 \\ x & x \geq 0 \end{cases} \tag{3.16}$$

Fully connected Fully connected layers are the same as in standard neural networks. They link all the activations, or cells of the activation maps to the next layer. They are computed with an element wise multiplication of the intensities with the weights and the addition of a bias. These layers are found near the end of the CNN. This is equivalent to having filters of the same size as their input.

Softmax - classification layer The softmax classifier constitutes the last layer of the CNN. For a CNN dedicated to classification the fully connected layer just before the Softmax layer gives scores under the form of a vector of the size of the number of classes. The softmax layer introduces the probability of the target X of belonging to each class Y_i by computing Eq. (3.17). $a_j, \{j \in [\![1,n]\!]\}$ representing the activation corresponding to the class j out of the n possible classes.

$$P(X \in Y_i) = \frac{e^{a_i}}{\sum_{j=1}^{n} e^{a_j}} \tag{3.17}$$

Dropout Dropout [37] is a layer that limits the over-fitting of the network. It prevents a certain percentage randomly chosen of the activations, e.g. 50%, to go through the next layer by setting them to 0. That prevents the network to rely on the same filters providing the same information and thus counter over-fitting.

Overall structure The CNN structure is composed of the functions presented stacked together. A common structure is represented in Fig. 3.11. It can be seen that the group of layers composed of the convolutional, activation and pooling layers are at the core of the CNN. To model more complex information, this core can be repeated to achieve deeper networks.

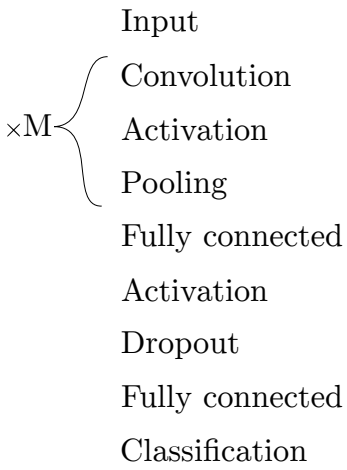
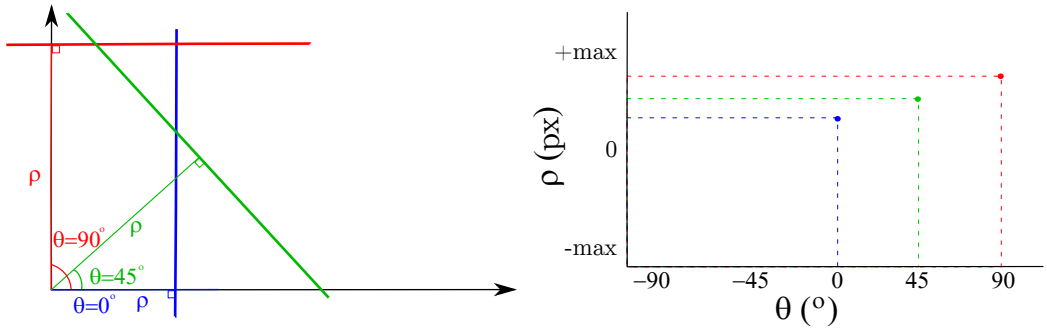


Fig. 3.11: Typical struture of a CNN with the group of 3 layers repeated multiple times to create a deeper network.

3.2 Target orientation determination

3.2.1 Hough transform



(a) Parameters describing the investigated lines.

(b) Resulting 2D matrix of the Hough transform.

Fig. 3.12: Principle of the Hough transform.

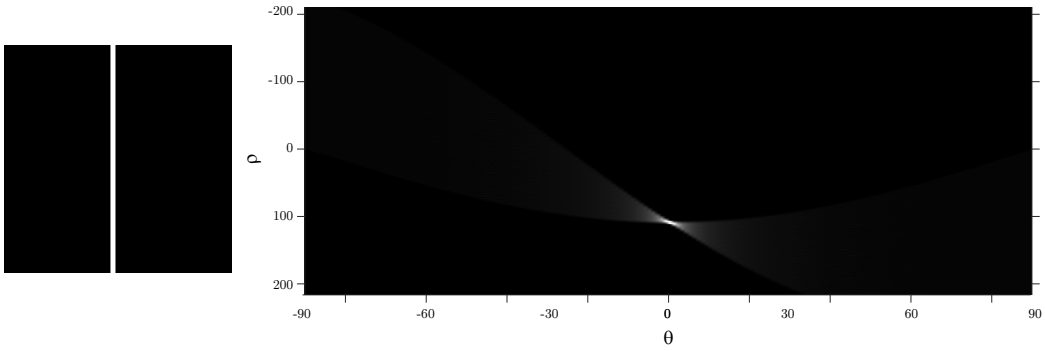
The Hough transform is an interesting tool to detect simple shapes and particularly straight lines. It is applied traditionally to a binary image. Each line in an image can be represented using the two parameters ρ and θ as seen in Fig. 3.12 (a). These parameters characterise the unique vector starting from the origin and perpendicular to the line. This vector can be expressed as:

$$\rho \cdot \mathbf{u}_\rho = \begin{pmatrix} \rho\cos(\theta)\cdot\mathbf{u_x} \\ \rho\sin(\theta)\cdot\mathbf{u_y} \end{pmatrix} \tag{3.18}$$

The line corresponds to the point with the coordinates (ρ, θ) in the 2D matrix resulting from the Hough transform in Fig. 3.12 (b). The value of this point is the accumulated intensities of all points confirming the existence of the line in the original image. Thus, the higher intensity of the point in the ρ, θ matrix, the more probable that a visible line exists in the original image on the corresponding line in the original image. More details on the computation of the accumulator 2D matrix is given in Algo.6.

A real example of the accumulator computed with the Hough transform on a single

bold line can be seen in Fig. 3.13.



(a) Original binary image.

(b) Accumulator representing lines in a 2D parameter space according to the orientation and the distance to the origin of the line.

Fig. 3.13: Real simple example of the Hough transform.

Algorithm 6: Hough transform

1 List all white points in the binary image I of size $n \times m$ such as Fig. 3.13 (a).

These points' coordinates are defined as:

$[x,y] \in \mathbb{N}^2$, $0 < x < n$, $0 < y < m$, $I(x,y) = 1$

2 Initialise the accumulator representing all lines to a null 2D matrix H:

$H(\theta,\rho) = 0$, $\forall\theta \in [-90,90]$, $\forall\rho \in [-D,D]$. D is chosen according to the resolution wanted.

3 **for** *each white point* (x_p,y_p) **do**

4 **for** θ *from* -90 *to* 90 **do**

5 Compute the corresponding ρ of the line with a θ orientation passing through the studied point: $\rho = \mid x_p\cos(\theta) + y_p\sin(\theta) \mid$

6 Cast a vote for the line (θ,ρ) by increasing its value in the accumulator: $H(\theta,\rho)++$;

7 **end**

8 **end**

9 Find the local maximums in the 2D accumulator $H(\theta,\rho)$ such as Fig. 3.13 (b);

Chapter 4

Feature-based classification

Contents

4.1 Summary

In this chapter, an ATR method that consists of a machine learning segmentation followed by a feature-based classification is proposed. The use of segmentation results in

a reduction of clutter correlation and computational time. GMMs were already used to segment images in the visual field and are here adapted to work with single channel SAR images. Indeed, segmenting SAR images can be challenging because of the blurry edges and the high speckle. The GMMs are used to create a model of the background present in the SAR image. This model evolves with time to include new distributions representing variations of the background and removes obsolete distributions at the same time. As the segmentation is seen in this case as a first step towards classification, the recall rate is deemed the most important score to evaluate. A high recall rate of 88%, higher than for the popular threshold method, was obtained. After the SAR image segmentation, the target goes through the feature-based classification process. The choice of feature is made after comparing the performance between different descriptors with a special emphasis on binary features, such as BRISK. The features computed in the tested image are matched with features found in training images with a target in a comparable orientation. The matches retained are geometrically coherent with a unique homography between the tested and trained target obtained with RANSAC. The resulting class for the tested target is the class of the training target that had the most matches. Results show the proposed method achieves a 93.40% probability of correct classification when the MSTAR SOC dataset with 3 targets is tested. Results also show this approach to be less sensitive to clutter than methods employing gradient features, such as SIFT or SURF, and template methods.

The presented feature-based classification is evaluated on two different datasets. Indeed, recent studies have reported very high ATR rates of SAR images based on the MSTAR database and one of the limitations of the MSTAR database is known correlation between clutter contribution in the training and testing sets. The method is thus evaluated on both the standard MSTAR SOC dataset for 3 targets and a proposed different partition for 3 targets with less correlation described in Section 2.4.

4.2 Introduction

In this chapter, feature-based classification methods that have been studied extensively in the visual band are investigated. The advantage of feature methods over template methods such as the one presented in Section 2.5.1 is an improvement of the computational and memory load because targets are represented only by a group of features and not the full image[38–40]. Feature methods also benefit from being totally humanely crafted over the better performing deep learning algorithms. There are thus no black boxes and it is possible to understand and eventually correct mistakes more easily. In order to focus the search for targets features, a segmentation of the image is carried out prior to classification.

The purpose of segmentation is to give meaning to an image and facilitate further analysis. It is challenging to segment SAR images, as there are no sharp edges to delimit the target or the shadow from the background. Segmentation methods have been extensively studied in the visible band, as well as in other domains with poorer resolution such as X-ray images or SAR images [41–44]. For SAR images, the presence of noise with a high standard deviation makes the choice of a simple segmentation method by fixed thresholds prone to errors as in some cases parts of the target remain undetected, or some background is falsely detected as a target. GMMs (Gaussian Mixture Models) have been successfully employed for segmentation in the visible domain and for sea-ice satellite SAR images classification in C-Band [45, 46]. GMMs enable a finer segmentation as, rather than using an intensity threshold, a description of the background is stored. The algorithm proposed in this chapter is such that the GMMs describing the background adapt automatically overtime along with the background evolution.

The segmentation unburdens the classification task from a significant amount of background processing that can lead to a large computational time and result in feature mismatches. Features are used to characterise specific areas of the target once they are isolated from the image. Examples of such feature methods are Principal Component Analysis (PCA) [38], Zernike moments [47] and Scale Invariant Feature Transform (SIFT) descriptors [48]. Nevertheless, some of these features present some disadvantages too.

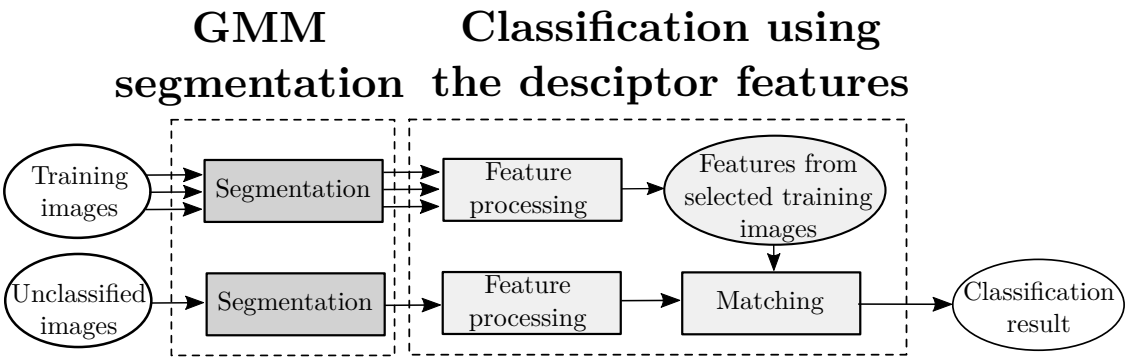


Fig. 4.1: Overview of the proposed feature classification with GMM segmentation and classification with binary features.

The PCA method may result in features that describe the high variable SAR speckle at the expense of valuable information about the target. SIFT performs particularly well in the visible band [49, 50] but has not been tested on the standard 3-target MSTAR dataset [48]. Feature crafting [27–31] has been extensively investigated in the visible band and could be applied to the SAR domain. Feature classification relies primarily on an accurate description of the target keypoints. However, as SAR images have poorer resolution than optical images, the detailed descriptors that perform the best in the visual band, such as SURF and SIFT, could be more affected by the SAR speckle than less detailed descriptors, such as binary features. Some methods achieving feature-based classification have been investigated in the SAR domain but their descriptive capability for SAR images have not been compared [38–40].

An overview of the whole method can be seen in Fig. 4.1.

4.3 Segmentation

4.3.1 Segmentation baseline: SARBake

Evaluating and comparing segmentation methods remains a true research challenge as there is no publicly available official groundtruth data. A manual segmentation method was proposed as a segmentation reference. It includes manual segmentation by an analyst followed by a quality control check by a supervisor [51]. However, the result of

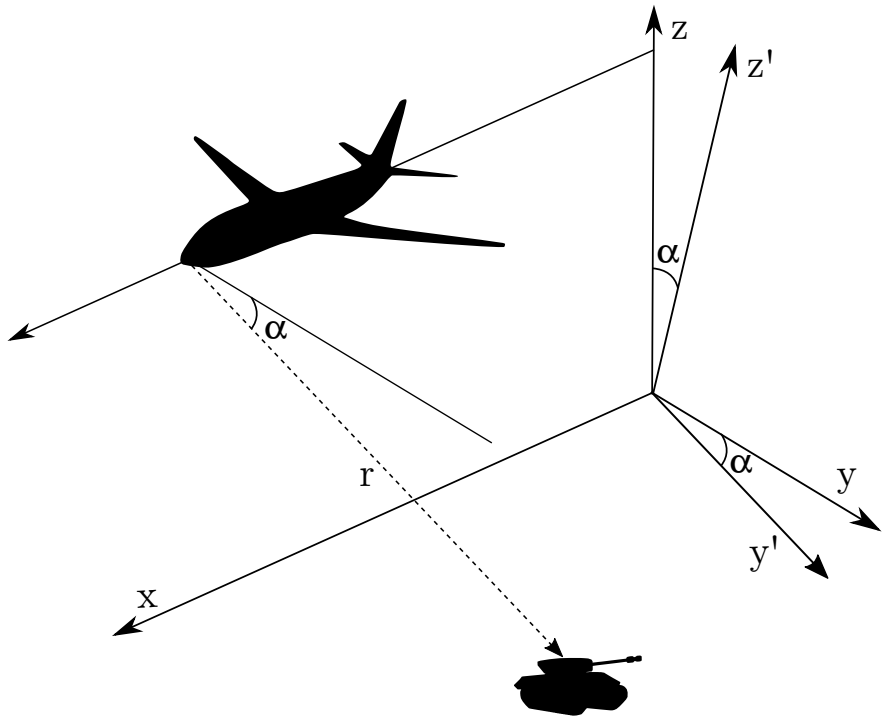


Fig. 4.2: SAR setup geometry with r the distance between surface of the model and the antenna and α the depression angle. The antenna goes at a speed $\vec{v}$ along the x-axis.

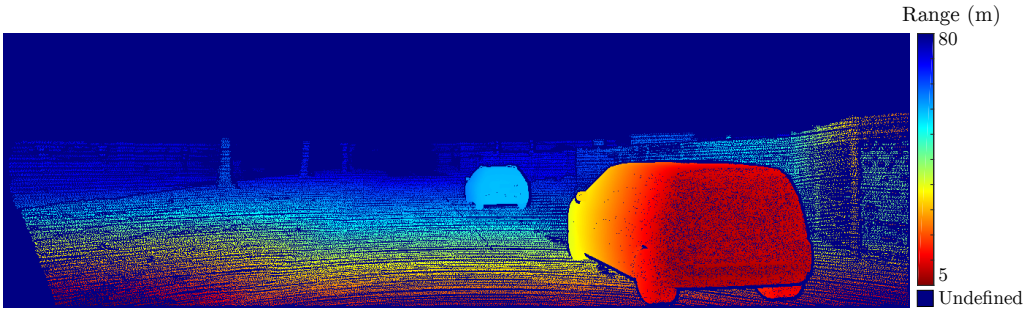
this segmentation is not publicly available and reproducing it would be labour intensive. An alternative baseline, called SARBake, is based on an orthographic projection of a Computer-Aided Design (CAD) target model according to the direction of the radar illumination [44]. In this method, far-field conditions are assumed, CAD representations of the target are assumed identical to the targets in the MSTAR and any multi-path effects are discarded.

For each image in the MSTAR database, the orientation of the target and the depression angle at which the image was taken are known. The geometry of the scene is reproduced using a CAD model of the target as shown in Fig. 4.2 so that the location of the target reflecting points are known. So that one axis can represent the range, a change of coordinates for which the y and z are rotated by an angle α is applied.

$$(x,y,z) \longrightarrow (x,r,\frac{z}{\cos(\alpha)}) \tag{4.1}$$

Following the change of coordinates, both the plane representing the ground and the CAD model of the target are rendered into a depth map viewed from the airplane. To that effect, all the rays illuminating the target are assumed parallel and the intensities of the resulting 2D image account for the target distance r from the antenna. An example of a depth map in a different context is given in Fig. 4.3. This example is acquired from a car in traffic using a Light Detection And Ranging (LiDAR). The ground and target are treated separately and each one has its own depth map that will be used to deduce the position of the shadow. The method could be improved when the profile of the ground is known instead of being assumed to be planar. The depth map is a 2D image. The depth map abscissa represents the x-axis and is thus aligned with the antenna movement in Fig. 4.2. The ordinates of the depth map represents a scaled z-axis $\left(\frac{z}{\cos(\alpha)}\right)$, proportional to the airplane height from the ground and target.

(a) Photo of an urban scene.



(b) Groundtruth depth map associated with the urban scene.

Fig. 4.3: Example of a depth map in the Kitti dataset representing vehicles and scene flow [52, 53].

Using a depth map for the target and a depth map for the ground, the location of the target, shadow and background is determined. As shown in Fig. 4.4, for each column, representing one step for a platform moving at $\vec{v}$ along the x-axis, the shortest r represents

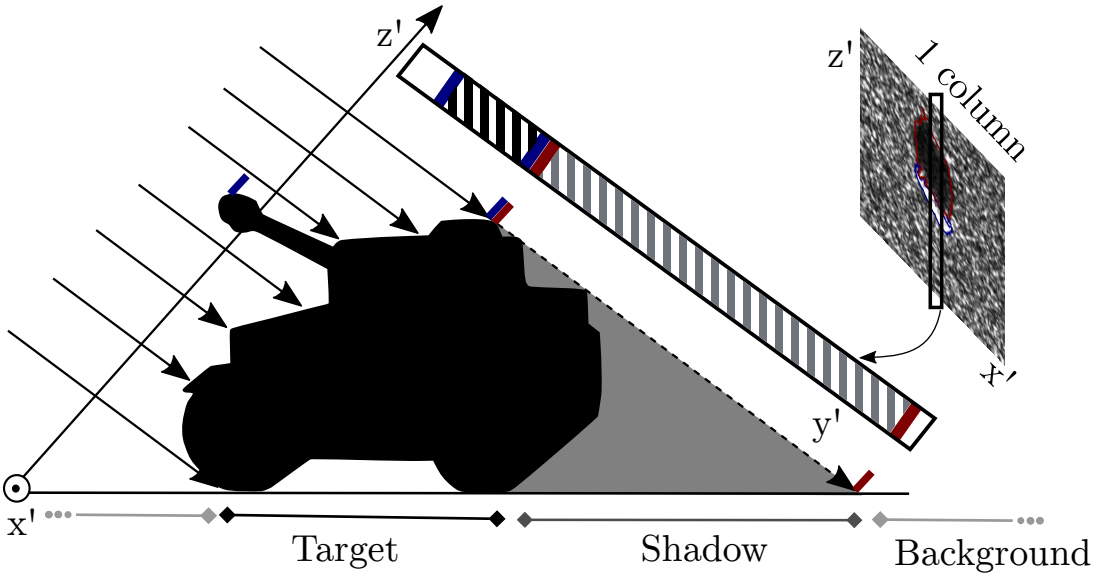


Fig. 4.4: Target, shadow, background area according to the SAR setup geometry.

the beginning of the target. The zones without any points belonging to the target or the ground belong to the shadow. The rest is designated as background. The depth map showing the set of distances from the antenna to the target or ground is transformed so that the intensity reflects the zone described with 0 for the background, 1 for the target and 2 for the shadow.

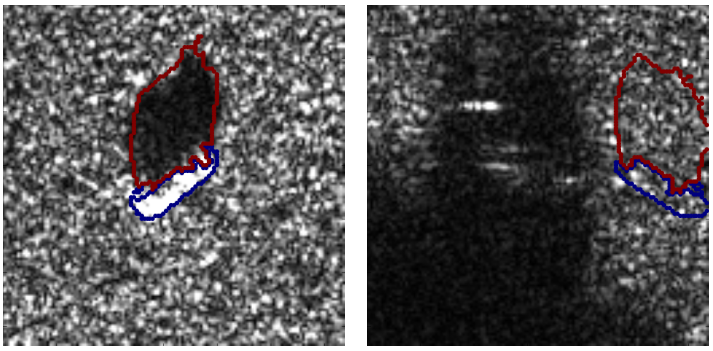
The location of the target in a SAR image is unknown and therefore it needs to be estimated to correctly superimpose the resulting segmentation on the SAR image. The precise location of the segmented image over the SAR image is given by the 2D cross-correlation between the two images defined as:

$$C(k,l) = \sum_{x=1}^{m}\sum_{y=1}^{n} I_1(x,y).I_2(x-k,y-l), \begin{cases} k \leq \mid m-1 \mid \\ l \leq \mid n-1 \mid \end{cases} \tag{4.2}$$

where $I_x \in \mathbb{R}(m \times n)$ is one of the images to be cross correlated. I_x is zero padded so that $I_x(x-k,y-l)$ is always defined. Thus when k or l is respectively larger than x or y, $I_x(x-k,y-l) = 0$.

The highest value of the cross-correlation gives the proper localisation of the segmented image over the SAR image.

The SARBake baseline for segmentation has been applied to all the images in the MSTAR SOC 10 targets described in Section 2.4.2. However, it could not be done for the EOC as the exact CAD models for the other target variants were not found. SARBake results are often well localised as seen in Fig. 4.5 (a) with only a few instances where the cross-correlation was affected by strong changes in the background as seen in Fig. 4.5 (b).



(a) Correct result of the SARBake segmentation.

(b) Result of the SARBake with strong background changes.

Fig. 4.5: Results of the SARBake segmentation on images from the MSTAR SOC 10 as seen in [54].

4.3.2 Segmentation reference method: Threshold method

In order to compare the proposed GMM segmentation method, the threshold method proposed in [55] is also implemented. In this method, to have similar images across sequences, the intensities of each image pixels are normalised between 0 and 1. A histogram equalisation is also applied.

The objective of histogram equalisation is to obtain the histogram the closest to a histogram with equiprobable intensities. The new intensities after histogram equalisation of the image are computed following this transformation.

The result of the normalisation and the histogram equalisation can be seen in Fig. 4.6.

The image is then processed through a mean filter before a fixed threshold found experimentally is applied as seen in Fig. 4.7 (a). A second threshold equal to the median

$$I'(x,y) = \frac{cdf_I(I(x,y))}{cdf_I(1)} \tag{4.3}$$

where I is the image to be transformed with intensities ranging from 0 to 1,
I' is the image resulting from the histogram equalisation,
cdf_I is the cumulative distribution of the image I.

of the intensities of the pixels remaining after the first segmentation is applied. This second threshold gives the final result of the segmentation as seen in Fig. 4.7 (b).

The full threshold segmentation can be expressed with the following pseudo-code.

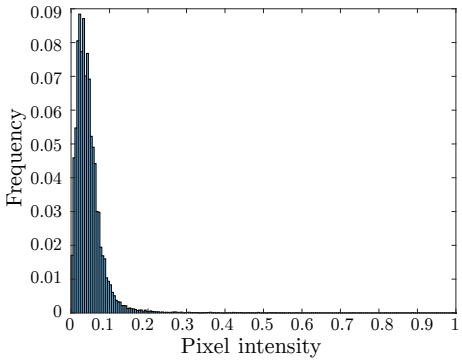
Algorithm 7: Threshold segmentation

1 Normalise the intensities.
2 Do a histogram equalisation.
3 Apply a mean filter.
4 Apply the threshold found experimentally.
5 Find the median of the remaining pixels.
6 Use this median as a second threshold.

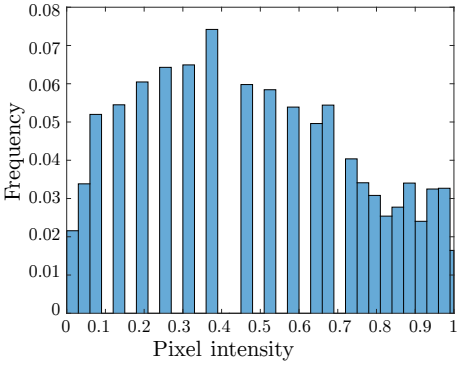
4.3.3 Segmentation with GMMs

Existing segmentation methods isolate the target in SAR images [47, 48]. After some pre-processing, segmentation is usually achieved by relying on thresholds. Some methods enhance the precision by applying an adapted threshold based on the contour of the previous segmentation results [56].

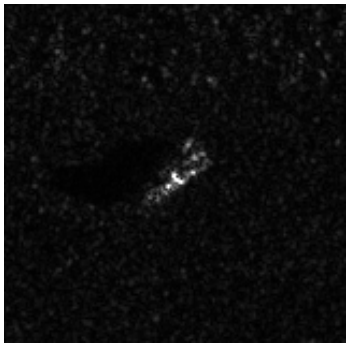
Previous methods are evaluated using scores that value above the rest the precision of the segmentation as with a Dice score [44]. In this case, the segmentation is used as a first step towards classification. The objective is to retain all possible target features while discarding most of the background. It is preferred in this case that some background is taken wrongly as a part of the target rather than missing a part of the target. The segmentation reduces the number of features to be analysed and matched as features in the background



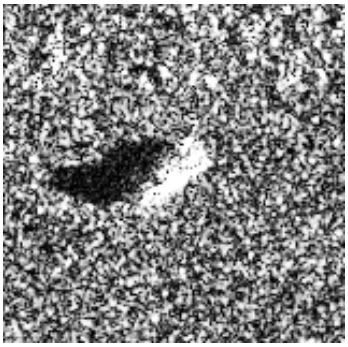
(a) Histogram of the original image.



(b) Histogram of the image after histogram equalisation.



(c) Original image.

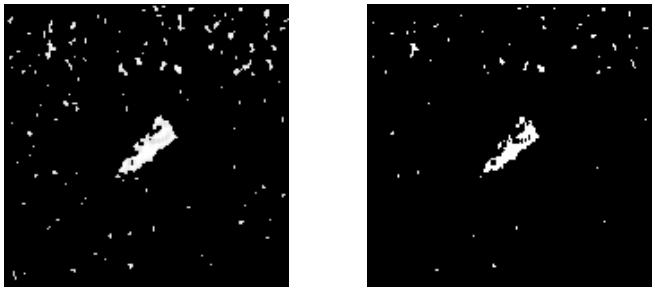


(d) Image after histogram equalisation.

Fig. 4.6: Result of histogram equalisation on the MSTAR SAR image.

will be mostly discarded. The segmentation can improve other classification methods as well. Indeed, template methods are focused on the target and have a minimal representation of clutter, and model based methods can be helped by the segmentation information stating which part of the image belongs to each class [57, 58]. The segmentation can also prevent mismatches between features of the target and the clutter as it will be suppressed through segmentation [59, 60]. The drawback is that part of multipath information could be removed during this process.

The main evaluation of our segmentation method will be through the recall rate, while providing the other classical rates, to assess the retention of the target area. However, the area surrounding the target can be misclassified as a target zone because of the recall rate focus at the expense of precision. This is balanced by the additional information on the target the multipath in the background can give.



(a) First rough segmentation from a fixed threshold.

(b) Second segmentation with the median of the remaining pixels.

Fig. 4.7: Result of the threshold segmentation.

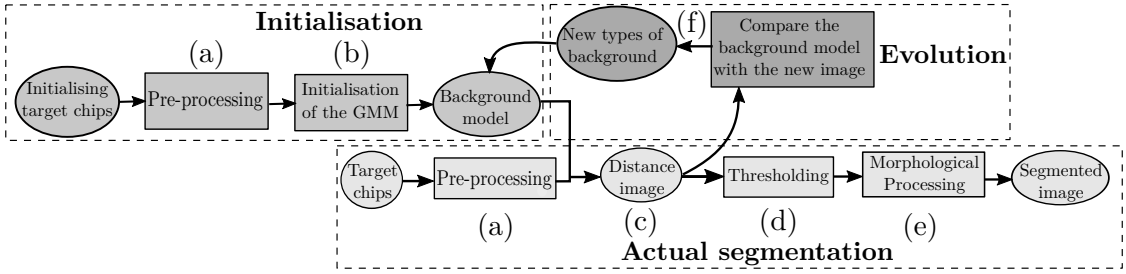


Fig. 4.8: Pipeline of the evolutive GMM segmentation proposed.

GMM method

The presented segmentation method relies on the GMM machine learning technique [8]. The major steps of the method can be seen in Fig. 4.8. Firstly, the initialisation creates an initial model of the background. To initialise the background GMM model, a few target images are chosen and processed to extract a GMM characterisation of the background. Then, the actual segmentation is carried out by comparing the background model to the image under test. This gives a distance image representing the area likelihood to belong to the background. The final image is obtained after thresholding the distance image and some further morphological processing. The remaining areas are the ones the least likely to be part of the background, namely the target in the foreground. The algorithm can work without evolution, however adding a learning phase makes it more accurate as the background is not the same throughout each sequence. The background model integrates new types of background as well as discarding the GMMs that no longer represent the current background along the sequence of images.

Choice of the initialising target images The first step consists in choosing the images that will be used to initialise the background. Few images from the beginning of the sequence are chosen rather than along each sequence. Indeed, It can be noticed that because of the evolution of the background, the background model leads to better segmentation at first if the initialising images are taken from a limited time period at the sequence beginning. The variety of GMMs fitting the background at each stage is achieved along the sequence by making the model evolve.

Data pre-processing Different pre-processing methods are tested in order to obtain the highest recall rates. The objective of the pre-processing (see Fig. 4.8 (a)), is to reduce the noise without blurring the contour of the target. The GMM segmentation is applied after the pre-processing stages. The pre-processing steps are chosen empirically. It led to the most accurate segmentation on challenging images for threshold segmentation with strong background changes. Two bilateral filters and a median filter is the combination that is retained. This pre-processing was compared with other pre-processing methods such as mean filtering, contrast stretching or histogram equalisation.

Adaptation of the GMM to the single channel SAR image The GMM is a distribution model obtained as a linear combination of Gaussian distribution functions to model data that could be subdivided in different subsets. It has already been used to segment visual images [45]. In our case, the GMMs represent the probability distribution of different areas of the images and follows Eq. (4.4). The data publicly available for SAR ATR are taken in similar backgrounds, thus the background is modelled instead of the target as it has less variance and is more predictable in the databases investigated. The proposed method is tested in the case of sequences of images taken one after the other, in which the background slowly evolves.

$$\begin{cases} \mathcal{B}(\theta_1,...,\theta_n) &= \{GMM_1(\theta_1),...,GMM_n(\theta_n)\} \\ GMM_i(\theta_i) &= \sum_{j=1}^{K} \phi_j * \mathcal{N}_j \end{cases} \tag{4.4}$$

where $\mathcal{B}$ is the background model, GMM_i is the i_{th} GMM composing the background model. Each GMM is the contribution of K Gaussian distributions with each $\mathcal{N}_j$ Gaussian distribution having a weight ϕ_j. The objective is to model the intensity probability distribution θ_i.

Images in the coloured visual spectrum have usually three channels (red, green and blue). Multi-polarised images could be an equivalent for SAR images but different polarisations were not available in the MSTAR dataset. The image is separated in several different patches. The intensities in each patch are assumed to belong to the same kind of background and will be modelled by a GMM. In practice, these local patches are each a 10×10 square group of pixels. As a group, the intensities balance the lack of information due to the single channel and limit the impact of the noise. The initialisation of the background model (see Fig. 4.8 (b)) begins by estimating the GMM parameters using a K-Mean algorithm. This algorithm is quick and gives a first idea of the clustering of the data in different Gaussian functions. The expectation-maximisation (EM) algorithm is used to provide an accurate estimation of the GMM parameters. As the background occupies the most space in the images, the GMMs related to the background are the most frequent ones. The most similar GMMs are grouped together to determine their prevalence. The similarity is established using the Kullback-Leibler (K-L) divergence based on a Gaussian approximation [61] written in Eq. (4.5).

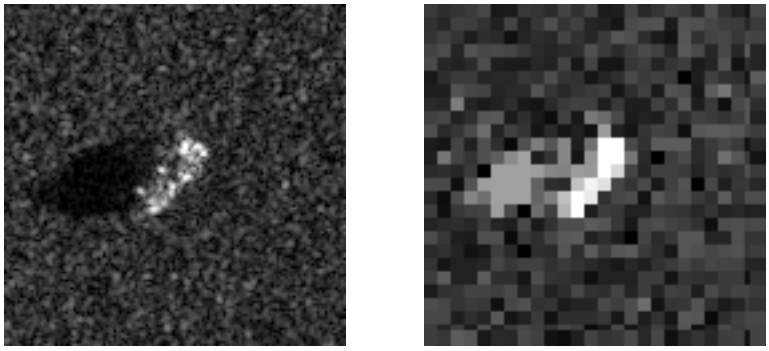
$$\begin{cases} dist(GMM_i, GMM_j) &= \min\limits_{\forall m\in S_i, \forall n\in S_j} KL(\mathcal{N}_m, \mathcal{N}_n) \\ KL(\mathcal{N}_m, \mathcal{N}_n) &= \ln\left(\frac{\sigma_n}{\sigma_m}\right) + \frac{(\mu_m-\mu_n)^2+\sigma_m^2-\sigma_n^2}{2\sigma_n} \end{cases} \tag{4.5}$$

where each GMM GMM_i is the contribution of S_i Gaussian distributions. The Gaussian distribution N_m is the m_{th} contribution to the GMM. This Gaussian distribution has a mean μ_m and a standard deviation σ_m.

If the divergence is below a threshold, the GMMs are considered similar and the

weight representing the occurrence of these GMMs is updated as the sum of the two GMMs weights. The GMMs selected to model the background are the 80% GMMs with the heaviest weight. They are likely to represent the background as one target only occupies around 2% of the image space in the MSTAR dataset.

Segmentation based on the similarity of the image to the background model. Segmentation is a two phases process. The distance image (see Fig. 4.8 (c)), is computed using the background model previously obtained and the image under test. It is then thresholded (see Fig. 4.8 (d)), to obtain the binary segmented image.



(a) Original image.

(b) KL distance image between the image background and background model.

Fig. 4.9: Distance image between the extracted GMMs and background model.

Distance image The image is divided in 10×10 square patches and GMM parameters are estimated for each patch. The distance image in Fig. 4.9 links the intensity of the pixels to the KL divergence as in Eq. (4.5) between the GMMs from the background model and the GMMs found in the image to segment. The new intensity of each patch is the minimum divergence found between the new GMM and the background model GMMs. A logarithmic filter applied to the distance image stretches the lower intensities and makes the choice of the threshold more accurate.

Thresholding The choice of the threshold (see Fig. 4.8 (d)) relies on the assumption that the target covers a small part of the image. The threshold is chosen so that only the

brightest part of the image is retained. A lower threshold could be used to select a larger potential target areas knowing that the post morphological processing would suppress the majority of false alarms. However, some false alarms could still remain while the segmentation would select more background areas.

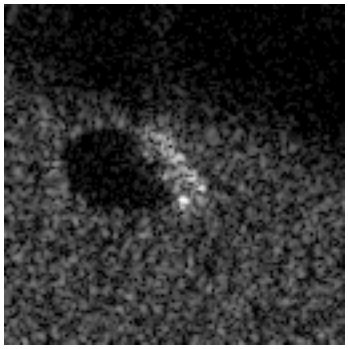
Morphological filtering Morphological filtering (see in Fig. 4.8 (e)), is essential to correct the first results of the segmentation. The target can be split up after the segmentation process in different parts and the dilation helps reconnecting them. However, the dilation makes the detected target larger and swells the misclassified background parts. The dilation is thus followed by an erosion. At this stage, the target is detected and its shape is well approximated but there are still false positives. Most are removed while areas below a specific surface are suppressed. An optional step is to add a dilation boosting the recall rate. As the target is now the only positive area, a dilation adds to the detection areas surrounding or on the target.

The standard GMM segmentation can be expressed with the following pseudo-code.

Algorithm 8: GMM segmentation

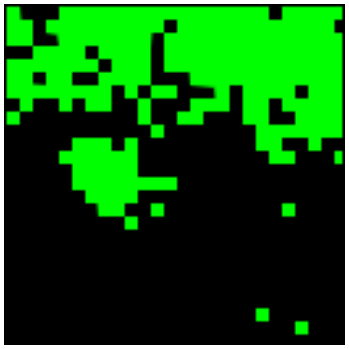
1 Pre-process the image with 2 bilateral filters and 1 median filter.
2 Compute GMMs over the image separated in 10×10 patches with a K-Mean and EM.
3 Compute which are the most present distribution using the KL distance to evaluate the GMM similarity. This provides the background model.
4 Use the KL distance to compute an image showing the distance of distributions to the background model.
5 Apply a logarithm function to the distance image.
6 Threshold the logarithmic distance image.

Evolution of the background model Along the sequence, the background can change and if the background model keeps only the original GMMs, the new background types will not be represented by the model and lead to false positives as shown in Fig. 4.10 (b).



(a) Original image.

(b) Segmentation result without evolution of the GMM background model.



(c) Location of the GMM candidates for the integration in the background model.

(d) Segmentation result with the evolution of the GMM background model.

Fig. 4.10: Comparison of the segmentation with and without the evolution of the GMM background model.

To take into account this problem, the set of background GMMs fitting the background at the beginning of the sequence evolves (see Fig. 4.8 (f)). The evolution consists in the removal of the out of date GMMs and the introduction of the GMMs fitting the new background types. The evolution step works for the images when they are following each other in a sequence.

GMM individual parameters are computed in the same way as during the initialisation. In addition, as expressed in Section 4.3.3, the weight of types of backgrounds is computed by computing the number of GMMs representing similar backgrounds together. The higher this number, or background weight, the more the modelled background is present in the image under test. Only types of GMMs representing backgrounds in the top 80% of most present distributions are kept.

The GMMs surrounding or located on the found target are removed as they can be related to the target. To do so, the location of the new GMMs in Fig. 4.10 (c) is compared to the dilated segmented image shown in Fig. 4.10 (d). The GMMs in the positive area will not be considered for background modelling. An additional criteria for a GMM to be kept or added to the background model for the next image is consistency across several images. If the GMM satisfies these criteria on 5 following images, it is included in the background model, otherwise the GMM is removed from the model. One can see the improved result of the segmentation by comparing Fig. 4.10 (b) and Fig. 4.10 (d) to the original image shown in Fig. 4.10 (a).

The GMM segmentation with an evolutive background and extra morphological filtering can be expressed with the following pseudo-code.

Algorithm 9: Evolutive GMM segmentation

1 **Initialisation**
2 Get a starting background model with Algo.8 on a set of images at the beginning of the sequence.
3 **for** *each image of the sequence* **do**
4 Compute the background model with Algo.8. Update the previous background model by adding new GMMs that are often present outside the target area in the current and the previous images. Remove the unused GMMs. Use the KL distance to compute an image showing the distance of distributions to the background model.
5 Apply a logarithm function to the distance image.
6 Threshold the logarithmic distance image.
7 Apply dilation and erosion to the segmented image (Morphological filtering).
8 **end**

Segmentation method	Precision	**Recall**	Dice Score
Technique 1	17 %	71 %	28 %
Technique 2	63 %	78 %	69 %
Technique 3	47 %	**88** %	61 %
Technique 4	62 %	55 %	58 %

Table 4.1: Segmentation results. Technique 1: Basic GMMs. Technique 2: GMMs with evolution. Technique 3: GMMs with evolution and morphological processing. Technique 4: Pre-processing and thresholding explained in Section 4.3.2[55].

Results of the various tested segmentation methods

Scores definitions Several scores can be applied to evaluate the segmentation. The precision rewards the detection of the target and is sensitive to the false detection of clutter. As expressed in Section 4.3.3, the recall rate is the score that is the most interesting as its focus is the detection of the area of interest, even if this implies the false detection of surrounding background. The Dice score shows the trade-off between those two scores. The detail of the calculation of those scores is explained thereafter.

$$\text{Precision} = \frac{\text{True positives}}{\text{True positives} + \text{False positives}} \tag{4.6}$$

$$\text{Recall} = \frac{\text{True positives}}{\text{True positives} + \text{False negatives}} \tag{4.7}$$

$$\text{Dice} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4.8}$$

Results The main objective of this segmentation is to detect the target as a whole even if the precision decreases. False negatives are more damaging for further processing and classification than false positives. Indeed, the clutter having none of the target features will be discarded at the classification stage whereas a part of target missing could mean the loss of a crucial feature. This is why the recall rate is used as the most important criterion to evaluate the performance of our segmentation method.

GMM technique A summary of the segmentation methods can be found in Algo.7, 8 and 9.

The very low rate of 17% of precision of technique 1 (Section 4.3.3 up to Section 4.3.3) in Table 4.1 shows that the basic GMMs technique is prone to false alarms. This endorses the overall change of the background throughout the sequences and that the background model should be updated accordingly. This hypothesis is confirmed looking at the 63% precision rate achieved once the evolution process is introduced as per technique 2 (Section 4.3.3 up to Section 6 but no morphological processing as in Section 4.3.3). The number of true positives increases with the recall rate. There can be more true positives because of the dilation only if the previously detected area is already on or near the target. The difference between the recalls of techniques 2 and 3 from 78% to 88% in Table 4.1 confirms the previous detection was correctly located. As can be seen in Table 4.1 from the precision and recall rate of techniques 2 and 3, the last morphological step is a trade-off between precision and recall. With the highest recall rate, technique 3 (Section 4.3.3 up to Section 6 and morphological processing as in Section 4.3.3) is favoured.

Comparison with other techniques Technique 4 (Section 4.3.2) in Table 4.1 is used as a preliminary step to several classification methods [48, 55]. This method consists of a histogram equalisation followed by a mean filter as preprocessing. A constant intensity threshold is then used to remove the pixels with a low intensity. The median of the intensity of the pixels remaining is used as a second threshold. Usually the target is well detected even if it is in several pieces. However, the background is detected as well. These results are shown by a comparable precision of the GMM technique 2 of 62% seen in Table 4.1 but a much lower recall rate of 55%.

Conclusion

The presented technique has a high recall rate of 88% satisfying the objective of a loose segmentation keeping most of the target and its features while removing the background to ease further analysis of the image. A higher recall rate was observed for this segmentation method than for other techniques that were tested. This can be interesting for feature or model based classification methods with a heavy computation load but that requires a detailed description of the target.

4.4 Classification with features

Once the segmentation is achieved, feature-based classification is carried out. In order to achieve this classification, features describing a specific part of the target have to be computed and compared to features from the training images. After the comparison, the most resemblant features are matched together and the correct target is determined. This feature-based classification is thus highly dependent of the describing ability of the chosen features. Several features used in the visual band are applied to the SAR domain independently to evaluate each in their ability to describe SAR features. A performance comparison between gradient descriptor and binary descriptor based classification method is presented. The gradient based descriptors compared are the Scale Invariant Feature Transform (SIFT) [27] and Speeded Up Robust Features (SURF) [28]. The binary descriptors compared are the ORB [29], Fast REtina Keypoint (FREAK) [30] and Binary Robust Invariant Scalable Keypoints (BRISK) [31]. Results show that binary features perform better on SAR than the gradient based features SURF and SIFT. Out of all the features compared, the BRISK based method achieved the highest score. The method is tested both on the standard MSTAR SOC 3-target dataset defined in Section 2.4.2 and on an alternative 3-target dataset with a different partition between training and testing sets defined in Section 2.4.2.

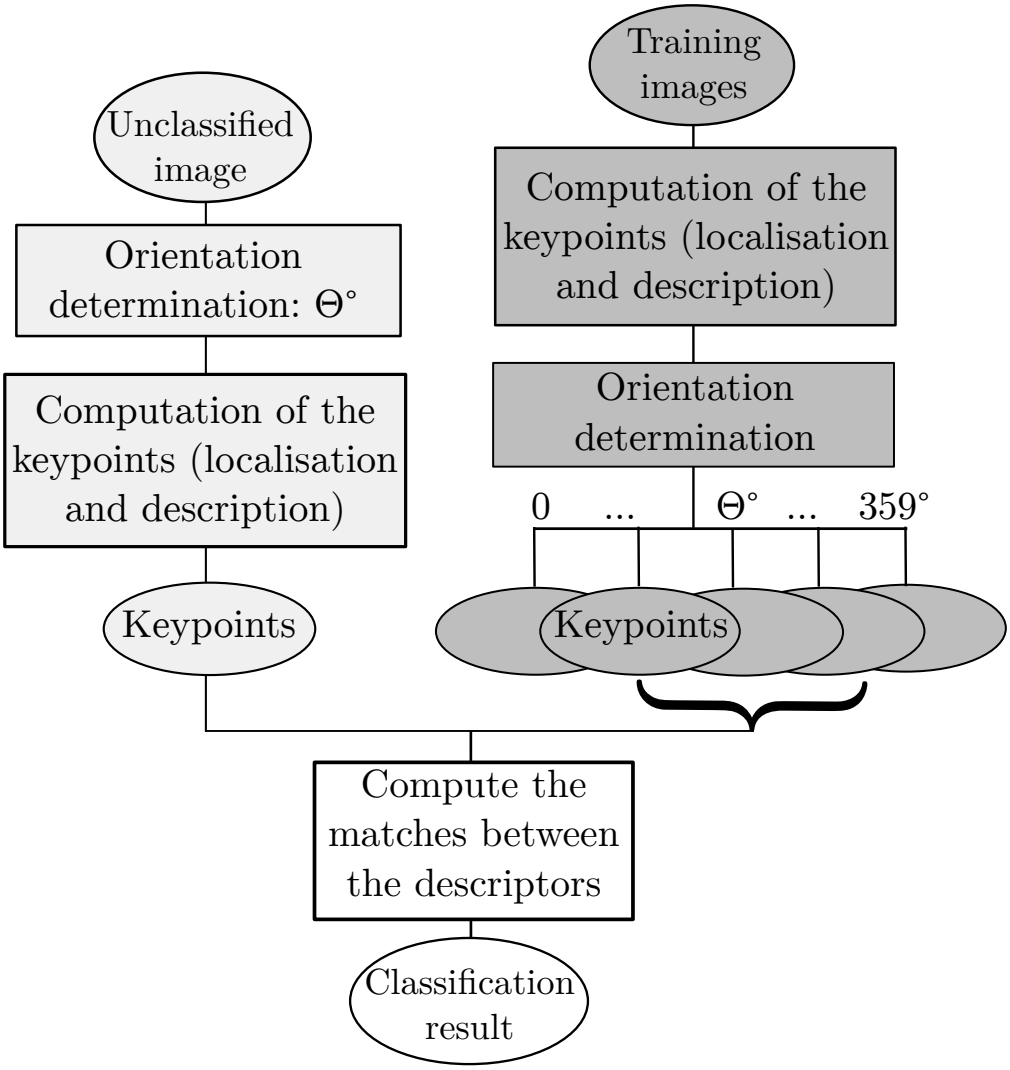


Fig. 4.11: Pipeline of the classification.

4.4.1 Methodology of the feature-based classification method proposed

The classification step associates a label to each detected target. The feature descriptors are chosen so to differentiate the best between different targets while being the most stable possible against changes of other factors such as noise or environmental changes. A detailed comparison of the various combinations of detector, descriptors and matching methods is presented in Section 4.4.1. After the segmentation step with GMMs presented in Section 4.3.3, the classification consists in matching the descriptors between the training and test images as shown in Fig. 4.11.

Features compared

Several types of features including detectors, descriptors and matching methods were tested and compared [7]. These were tuned to achieve the best classification rate and only those providing the best performance were employed in the classification task. The detector finds the location of a relevant keypoint in the image. Its surroundings are characterised in a vector by a descriptor. The keypoints are defined by their location and the vector characterising the corresponding local area. The resulting keypoints will be compared and associated with one another using a matching method. The detectors analysed in this work included the Difference of Gaussian (DoG) [27], the Box Filter (BF) [28], Harris corners [62], Adaptive and Generic Accelerated Segment Test (AGAST) [31, 34], Features from Accelerated Segment Test (FAST) [33] and fixed points on a grid laid over the target. The descriptors were selected based on previous research between those providing the highest performance in the visible domain. These include the Scale Invariant Feature Transform (SIFT) [27] and speeded up robust features (SURF) [28]. Binary features are also included such as Fast REtina Keypoint (FREAK) [30], Binary Robust Invariant Scalable Keypoints (BRISK) [31] and Oriented FAST and Rotated BRIEF (ORB) [29]. The acquisition of these descriptors is further explained in Section 3.1.1. Feature matching is based on two matching methods: Lowe's Nearest Neighbour Distance Ratio (NNDR) [27], or a brute-force method. The brute-force method used in this work retained only the best match for each feature of the tested target. The distance used to implement these was either the Hamming distance or the Euclidean distance for binary descriptors and non-binary descriptors respectively.

Determination of the orientation of the target

The target orientation is computed using a Hough transform [56, 63]. The image is segmented using the GMM models as in Section 4.3.3. Dilation and erosion operations with a small kernel are applied to the segmented binary image to smooth the edges of the target. The biggest shape is retained and the rest is considered as noise and is deleted. The

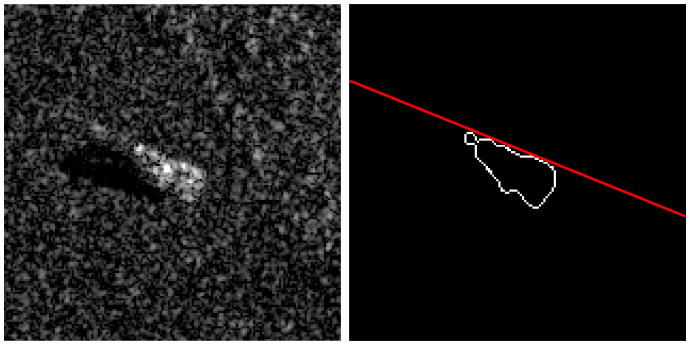


Fig. 4.12: Determination of the target orientation with Hough transform.

image of the contour is used to compute the Hough transform. Only the brightest peak in the Hough transform matrix is selected, giving the longest visible line in the image as shown in Fig. 4.12. This line is superposed to the longest edge of the target which gives the target orientation. As the final estimation of the target orientation is done by fitting a line to the target contour, the front of the target is not distinguishable from the back. Thus, the orientation is known modulo 180°.

Selection of images to model the target

The classification relies on the feature matching between images of the target under test and all training targets. In order to facilitate the matching, the target in the training image should look the most alike to the test image. The training images are chosen so that the targets they contained have a close orientation to the target to be recognised. The targets in the training images have an orientation within a range of 80° to the target to be recognised as shown in Eq. (4.9).

$$\begin{aligned}\theta_{training} \in &[\theta_{testing} - \psi; \theta_{testing} + \psi] \cup \\ &[\theta_{testing} + 180 - \psi; \theta_{testing} + 180 + \psi],\end{aligned} \tag{4.9}$$

where ψ is the orientation tolerance set to 20,
$\theta_{testing}$ is the orientation of the unclassified target,
$\theta_{training}$ is the orientation of one target from the training image set.

25 training images are chosen within the set of compatible training images to provide the features forming the target model. The value of the orientation tolerance ψ must be small enough so that the training and test images are similar and big enough to ensure that there are enough training images within this range to form the target model. The choice of ψ is justified in Section 4.4.2.

Feature description of the target

The features represent the pixel intensities over an area and must be standardised so that feature matching is effective. After the segmentation with the GMMs, the images are thus pre-processed. A median and bilateral filter are applied to reduce the noise followed by a contrast stretch aimed at thresholding the 5% highest and lowest intensities. The various descriptors are computed either with their built-in detector as in Section 4.4.1 or on equally spaced locations (intersections of a grid) on the target area found by the segmentation. The grid method performed better as the usual detectors were perturbed by the speckle and did not find enough robust features. The grid will thus be used as a detector to choose the location where the descriptors will be computed in the results. For each descriptor, the relevant parameters (octave number, pattern scale, peak and edge threshold) were tuned until the best performing combination was found. BRISK performed the best as shown in Section 3.1.1 and will be used thus in the final method.

Matching

The resulting descriptors were matched using either the Lowe's ratio method, or a brute-force method [27]. The Hamming norm was used for binary descriptors and the Euclidean norm for gradient descriptors. This brute-force matching assigns the best match, i.e. the match with the lowest distance for each point, while excluding the possibility for points to have more than one match. Both methods are used for matching as reported in Section 4.4.2. Many false matches were observed in the results. To limit the number of false matches, RANSAC was employed to mitigate the geometrical differences caused by

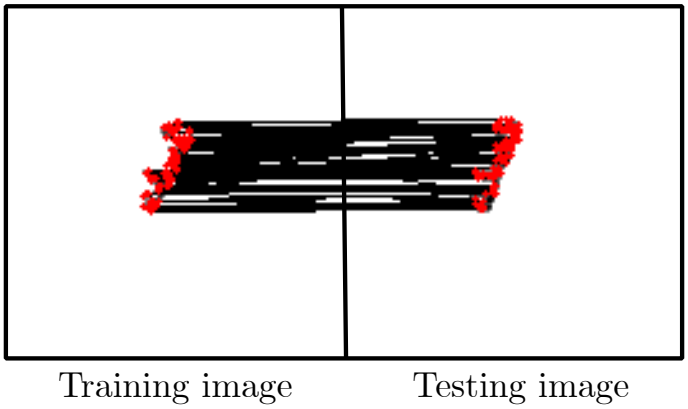
different viewing conditions and obtain results of the type of those shown in Fig. 4.13 [35]. In Fig. 4.13 (a), features found on the training target are assigned to similar features on the testing target. The only matches retained in Fig. 4.13 (b) are all the all black lines. Those matches are compatible geometrically with RANSAC and that is shown as the lines representing the matches are parallel. Indeed, if one match is at the front of the target, another match just next to it can't be on the target rear. If the targets are indeed similar, most of the matches should be retained. If most matches are lost between Fig. 4.13 (a) and Fig. 4.13 (b), it is unlikely that the target tested is of the same type as this training target. The type of the target with the greatest average number of remaining matches is assigned to the target under test.

4.4.2 Results

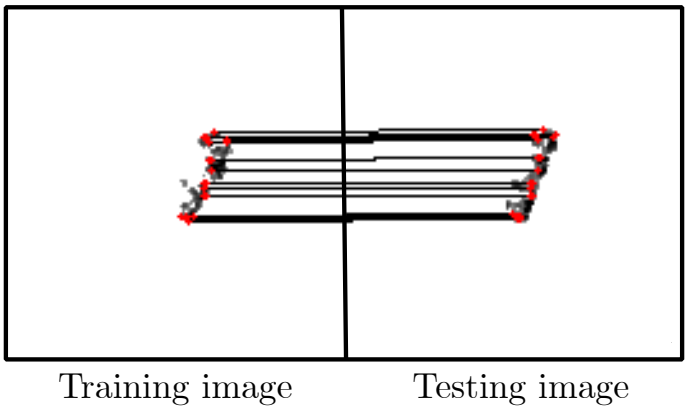
Results were obtained with an alternative partition between training and testing for 3-class targets SAR ATR problem. This partition, referred as dataset B in Section 2.4.2, was selected as it was deemed less affected by background correlation as shown in Section 4.4.2. Section 4.4.2 shows the precision of the standard Hough transform determining target orientations. It ensures that the training images picked for the target model have a similar orientation to the tested target. In Section 4.4.2, the choice of BRISK as the feature descriptor is justified by a comparative study with other descriptors. The relevance of the target model generated depending on the restriction of the orientations of the training targets and the number of computed features per target is described in 4.4.2. Lastly, the full ATR method is evaluated in Section 4.4.2.

Correlation between training and testing sets in both dataset A and proposed dataset B

To justify the use of another partition than the usual one for 3-class targets SAR ATR problem, a NN based ATR method [13, 24] is applied to dataset A and B from Section 4.4.2 with the full images and background-only images (target removed). Table 4.2



(a) Original matches.



(b) Matches after RANSAC.

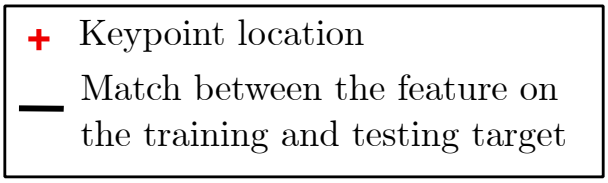


Fig. 4.13: Matches refinement with RANSAC.

shows correlation between the training and test images in dataset A as the algorithm still achieves a 90.88% rate without most of the target information. The same method only reached 57.88% in dataset B and was thus less influenced by correlation.

Orientation determination

The Hough transform is evaluated using the errors between the orientation found by the Hough transform and the supplied target azimuth. This method is however not capable of distinguishing the front from the back of the target and thus all errors are given modulo

	Original images	Background only image (target removed)
Dataset A	99.25%	90.88%
Dataset B	82.75%	57.88%

Table 4.2: Nearest neighbour recognition rates between Dataset A and B.

180°. The evaluation is made on the same dataset as in [64] with 6874 images from the 10 targets belonging to the Standard Operating Condition (SOC) at both depression angle 15° and 17°. This is the MSTAR SOC dataset for 10 targets that is described in Table 2.3. The mean error and standard deviation of the error are given as it has already been done but the Mean Absolute Error (MAE) and the Root Mean Squared Error (RMSE) are also provided as these indicators can give some additional information [56, 64, 65]. MAE and RMSE are defined in Eq. (4.10).

$$MAE = \frac{\sum_{i=1}^{n} |a - \hat{a}|}{n}, \qquad RMSE = \sqrt{\frac{\sum_{i=1}^{n} (a - \hat{a})^2}{n}}. \tag{4.10}$$

where a is the correct angle given by the MSTAR database,
$\hat{a}$ is the angle found by the Hough Transform.

The RMSE is more affected by the bigger errors because of the square applied to the error. Some high RMSE in Table 4.3 can be explained because the Hough transform could mistake the short side of the target for the longer side, adding an error close to 90°. Conditionally Gaussian Model (CGM) is more accurate but requires the target type beforehand [66]. This accuracy is comparable to the continuous wavelet transform without slant plane adjustment [64] but could be improved by taking the average angle of a cluster of lines instead of using a single line [56]. These results are shown to complete the results on the robustness of the feature classification method against change in the target orientation in the training and the tested image.

Target	Image number	Mean	σ	MAE	RMSE
2S1	573	-0.75	20.97	8.31	20.96
BMP2	1285	-0.10	13.79	5.85	13.78
BRDM	572	-0.47	24.99	11.84	24.97
BTR60	451	-0.33	9.96	4.41	9.96
BTR70	429	-0.31	12.96	5.01	12.95
D7	573	-1.32	19.27	9.25	19.30
T62	572	-2.15	13.42	6.48	13.58
T72	1273	-1.64	14.13	6.77	14.22
ZIL	573	-2.14	15.22	8.38	15.36
ZSU	573	0.73	21.57	10.60	21.57
Total	6874	-0.87	16.81	7.52	16.88

Table 4.3: Statistics of the errors in the target orientation determination.

Comparison of the possible descriptors for the target characterisation

25 images are used to build the target model with an orientation tolerance ψ of 20°. The best scores on dataset B for each combination of detectors, descriptors and matching methods are shown in Table 4.4. SIFT [27] and SURF [28] features perform well in the visible band, but binary descriptors, such as FREAK [30], BRISK [31] and ORB [29], outperformed them largely in the SAR domain on the MSTAR. BRISK performed the best as seen in Table 4.4 with a classification rate of 91.57%. FREAK and BRISK performed more than 10% better than ORB.

	Gradient descriptor		Binary descriptor		
Detector	Grid	Grid	Grid	Grid	Grid
Descriptor	SIFT	SURF	ORB	FREAK	BRISK
Matcher	NNDR	Brut. L2	Brut.H.	Brut.H.	Brut.H.
Result	56.94%	51.34%	78.93%	89.00%	91.57%

NNDR is the nearest neighbour distance ratio, brut. is the brute-force matching method used with *H.*, the hamming distance or *L2*, the Euclidean distance.

Table 4.4: Comparison of performance for gradient based and binary descriptors.

Optimal description of the target model

The construction of the target model is a trade-off between the precision of the model and the computational load to build it. The following results show the effect of varying the keypoint detection method and the amount of keypoints used on the target. In addition to that, the impact of selecting training images with a target in a similar orientation to the tested target is shown.

Evolution of the classification accuracy with the number of keypoints evaluated SAR images do not have a resolution as high as that of optical images. In addition, the detection of robust features can be difficult because of the high speckle and the small size of the working area. Indeed, the target in the MSTAR occupies on average 2-4% of the total image. The grid is not a detector per se but it enables the evaluation of descriptors by bypassing the challenging detection task in SAR. The score achieved with the AGAST detector is also given. The results are for the BRISK based classification method on dataset B using 25 training images for the target model creation and a 20° orientation tolerance.

Detector	Space between each intersection on the grid (px)				AGAST detector [31]
	1	2	5	8	
Classification rate	91.57%	89.83%	79.03%	75.74%	81.40%

Table 4.5: Impact of the detection and number of keypoints.

Optimal orientation range for selecting the training images used for matching The training and testing targets look the most alike for similar orientations. The results of the BRISK based classification are shown with 25 training images and a 20° orientation tolerance for the choice of the training targets. As shown in Table 4.6, comparing the test target with training targets with similar orientation improves the score from 79.86% to 91.57%. It shows that the feature themselves are changed, and not only rotated as

BRISK is rotation invariant. Finding the orientation with the Hough transform is not computationally intensive in this case because images were already segmented for the feature classification.

Viewing angle tolerance ψ	20°	30°	40°	No orientation selection
Classification rate	91.57%	88.18%	85.41%	79.86%

Table 4.6: Impact of orientation resemblance between the training targets and the tested target.

BRISK classification on GMM segmented and processed images

The classification rate on the background-only images represents the combined influence of the target shadow, multipath and the correlation between the training and test images backgrounds. The NN method achieves a recognition rate of 90% using the background-only images in dataset A as shown in Table 2.4.2, whereas our method scored 41.76% as shown in Table 4.7 showing a lesser influence of background correlation. Indeed, key-points detected in the background, are removed by the matching and RANSAC steps. Clutter is also largely ignored by the BRISK features because intensities comparison and not intensities themselves are registered, thus reducing the influence of most small intensity changes caused by noise.

The importance of the segmentation for the feature classification, in addition to reduce the feature computation and matching time, is shown in the score drop from the segmented target to the full original image (from 93.40% to 67.10% in dataset A and from 91.57% to 61.66% in dataset B). The segmentation avoids mismatches and makes it possible to reach much higher scores. The proposed method achieves a score of 91.57% in the proposed partition, dataset B, and 93.40% in the 3-class SAR ATR problem usual partition.

	Original images	Segmented images: Target only, background removed	Segmented images: Background only, target removed
Dataset A	67.10%	93.40%	41.76%
Dataset B	61.66%	91.57%	38.64%

Table 4.7: Classification rates achieved using the proposed method.

4.4.3 Conclusion

Target recognition in the MSTAR dataset based on binary local features has not yet been reported in the literature. Binary features, in addition to being faster than SIFT and SURF, are less influenced by clutter and give more robust features. The proposed BRISK feature-based classification ATR method achieves a 93.40% classification rate on the MSTAR SOC with 3 targets, describing better keypoints than usual visual features while still ignoring clutter. The choice of extracting features from training targets with a similar orientation to the target under test proved efficient raising the classification score from 79.86% to 91.57% on the MSTAR dataset B with 3 targets described in Section 2.4.2 but required segmentation of the image.

Chapter 5

Deep learning classification

Contents

5.1 Summary

This chapter consists in the implementation of deep learning methods in order to perform SAR ATR. The main architecture used is an AlexNet to which the number of classes in the last layer is adapted to the number of targets in the SAR dataset. Transfer learning is applied to the CNN trained initially on ImageNet in the visual domain to adapt to the various SAR dataset. The learning rate is chosen after a review of the scores reached after 5 training epochs using randomly chosen learning rates inside a reasonable range.

To compensate for the low number of training images, data augmentation is used to improve the scores obtained. A traditional translation data augmentation is included with a random X-Y translation applied to the original training data. There is no improvement in the MSTAR dataset and the result was even worsened on the MSTAR EOC 1 with an average classification delta of -0.81%. However, the CNN becomes much more robust to target translation with an average classification improvement of 28.89% compared to the CNN trained on the original data only. This does not happen in the MSTAR dataset but could happen realistically in SAR images. The translation data augmentation improves by 10% the classification score of the AlexNet on the MGTD. The translation data augmentation is thus kept for the rest of the chapter. The translation data augmentation makes the classification rate reach 97.77%, 74.72%, 92.57%, 85.46%, respectively on the MSTAR SOC 10, MSTAR EOC 1, MSTAR EOC 2, MSTAR EOC 3.

In addition to the traditional translation data augmentation method, a noise-based data augmentation specific to SAR data is proposed. The objective is to simulate a noisier acquisition. The range profiles are modified with the addition of noise following a Weibull distribution before computing the SAR images. This data augmentation is only applied to the MGTD images for which the range profiles are available. The results are presented in Section 5.3.3. The combination of the translation and noise data augmentation makes the classification rate reach 91.20% on the MGTD.

The second part of the chapter is focused on the implementation of the pose-informed architecture. The objective is to take better into account the impact on the target appear-

ance in SAR images produced by a target orientation change. The target recognition is handled in two stages in the pose-informed method. First, the orientation of the target has to be determined. The image is then analysed by the CNN specifically trained on images of targets with similar orientations as the target to be recognised. In order to determine the target orientation, a Hough transform is first applied to the contour of the target. 90° errors are prevented by comparing the amount of illumination of the target in rectangles superimposed on the target with a horizontal or vertical direction. This gives an accurate orientation between 0° and 180°. The full orientation of the target is determined using a CNN to compute the direction uncertainty of the target. The CNN is trained to distinguish the front from the back of the target. The pose-informed architecture consists in n specialised CNNs, each trained on $\frac{360}{n}$ degree range of orientation, with n between 2 and 8. A parent CNN is firstly trained using transfer learning from the visual to the SAR domain. Then, for each specialised CNN, this parent CNN is retrained using transfer learning from the full SAR database to the SAR images containing a target with orientations from the appropriate range. The issue for such architecture is the very scarce data that can be used for training. As only a small percentage of the training data is used for validation of the model, very few images are left for validation. It is thus more probable for different models to achieve an identical validation score. In order to measure fairly the results, it is proposed to provide not one unique result but a range of results achieved with the same best validation score. The complete results are given in Section 5.4.5. Apart from the MSTAR EOC 3, the proposed pose-informed model performs better than the standard CNN on all dataset tested. The classification scores are improved by 2-6%.

5.2 Introduction

Machine learning algorithms are able to use input training data provided to them to adjust their parameters in order to achieve better scores. Algorithms stemming from the Machine learning domain have already been implemented on the MSTAR dataset to perform

ATR. They have been relying on the following algorithms: Boosting algorithms such as Adaboost, are trained by adapting a combination of weak classifiers to create a strong classifier. Support Vector Machine (SVM) relies on finding the hyperplane maximising its distance with groups of data points of different classes. K-Nearest Neighbour (KNN) finds the points representing the best each class. Each point should gather the highest number of neighbouring data points from the same class. MInimum Noise and Correlation Energy (MINACE) aims at finding a class specific filter minimising the correlation between training samples of a single class. All these algorithms use either the full image as primary data or a lower dimensionality representation of the image achieved by processing the image with a function such as a 2D Fourier transform, elliptical Fourier descriptors or a PCA (Principle Component Analysis). Classification can be obtain by supplying the target images to a single algorithm such as SVM or improved SVM, boosting algorithm or MINACE filter [55, 67–69]. Classification can also be obtain by interpreting information from the training images with several algorithms such as KNN, MINACE and a SVM, or a boosting algorithm with a SVM [70, 71]. Methods relying on machine learning achieve better results than other direct classification methods, comprising of a direct comparison between training and test images as those presented in Chapter 4. All of these algorithms have however mainly only been tested on small MSTAR datasets, i.e. 3 to 8 targets, with no extensive conditions, i.e. no change of depression angle bigger than 2°, no target variants or different target configurations. The SVM method, when applied to a sparse representations of images belonging to a more complex dataset such as the MSTAR SOC 10, only achieved 80% of correct classification [72]. It seems thus that the SAR ATR problem requires more complex algorithms in order to accurately differentiate between the target classes. Fusion of features given by some of these machine learning algorithms (MINACE, KNN, SVM) achieves only slightly better results [70]. Thus, another approach is required.

The current state of the art SAR ATR methods rely heavily on CNNs issued from deep learning and also commonly used in the visual domain. Some neural networks have

been specifically developed for SAR ATR which are shallower than CNN solutions used in the visual domain as the image resolution in the visual domain is much higher than that of SAR, providing very detailed features [21, 73, 74]. CNNs can be used directly but also as a feature extractor. Complex features can be extracted from the images in the activation maps after the activation layers and the multiple stacked convolutional layers. These features can be directly used by a SVM for classification. This method provides improved results with respect to former feature-based solutions such as applying a feature dictionary on the sparse representation of the target (up to 99.5% compared to 80% on the MSTAR SOC 10) [75–77]. CNNs used directly or as a feature extractors have in any case to be trained to fit the SAR data. They can be either trained entirely on SAR data or benefit from transfer learning with a pre-training in the visual domain on ImageNet to allow a greater amount of data preceding SAR training [76, 77]. CNNs can achieve extremely high recognition rates on the MSTAR with scores reaching 99.1%, 96.12%, 98.93% and 98.60% on respectively the MSTAR SOC 10, MSTAR EOC 1, MSTAR EOC 2 and MSTAR EOC 3 or even higher on the MSTAR SOC 10 [21]. These scores were achieved with A-ConvNet in which the fully connected layers at the end of the network are replaced by convolutional layers to reduce the number of free parameters to train and try to hinder over-fitting.

Correct classification rates of deep learning methods essentially rely on their training capabilities and an essential variable of the training process is the training data. Ideally, the data should be abundant and diverse to avoid overfitting. In the ideal case, the images should also be independent, however, this is not often the case of SAR dataset due to the acquisition constraints. The image acquisition procedure is more complex in the SAR domain than in the visual domain and, as a result, overfitting becomes a key challenge in SAR ATR. Several methods are investigated in order to optimise the usage of SAR data. The network can be modified itself with the use of shallower CNNs and less parameters to train, or with the addition of dropout layers to randomly silence activations and prevent the usage of only a few features [21, 37, 73]. Training of the CNN can also be improved

to include unlabelled data as with a Contractive Auto Encoder (CAE) [78]. The network has to extract relevant features of an unlabelled image and summarise it into a compact representation. This constitutes the encoder part of the CAE. The network then has in the decoder part to recreate the original image from the compact representation. The loss function of the CAE is related to the similitude between the original image and the image created by the CAE. The encoder part has thus to learn the relevant features that could be further used for recognition purposes. After this first training, the encoder is used as the first part of the CNN with a regression softmax layer added at the end [79, 80]. Alternatively, another classification method can replace the classification layer [81]. The addition of unlabelled SAR data to the training from other database than SAR ATR specific databases could increase the classification score and the robustness of such a network.

Another unsupervised approach is to use a Generative Adversarial Network (GAN) to learn the features. The GAN method consists of two distinct entities: a generator and a discriminator. The generator creates images while the discriminator, that is a CNN, has to differentiate between real images and synthetic images created by the generator. Once the discriminator is well trained, it is able to produce what looks like realistic synthetic images. This method produces very convincing images in the visual field and has lately also been implemented in SAR [82–85]. If this method can help directly with the training of the classifier, as the discriminator needs to learn the images features to produce new images beforehand, the direct usage of the synthetic image to increase the training set could become a problem. Indeed, previous research in different domains has shown that GAN can introduce some artefacts in the image [86, 87]. These can be sometimes corrected if first identified [87].

In some instances, the system had also to be guided to represent both the target and clutter. Because of the tendency of CNNs to overfit SAR data, it is not known to what extent the produced images are realistic, as the background in training images is often similar within a whole sequence and the multi-path effects are also different from one

target to another. SAR images are more difficult to totally interpret than visual images, which makes the detection of the potential artefacts a challenging task itself. In practice, previous work shows the addition of realistic simulated images to the training set can improve the classification scores [88]. Residual networks have also been used to refine images generated without artificial intelligence but using point scatterer models on 3D target models [89].

In this chapter, data augmentation solutions which might provide improved performance in conjunction with other method creating artificial images are investigated. Data augmentation has the disadvantage of producing images that can be less varied than the more computationally expensive simulated images obtained with 3D models. Data augmentation can produce however more realistic images compared to artificial images generated with GAN as these could be corrupted with artefacts [86, 87]. Images with targets with new aspect angles respectively to the aspect angles present in the training set have been simulated by averaging images of targets in close orientations [20]. However, this successfully works only for very close orientations because the scattering properties of target components change with the aspect angle. For example, the barrel is only discernible when perpendicular to the incoming signal.

Data augmentation solutions have already been implemented in SAR [77, 90, 91]. Several types of data augmentation are implemented in this chapter. A translation data augmentation is implemented and the robustness of the resulting trained CNN is evaluated against translation on the MSTAR EOCs dataset. A data augmentation simulating realistic SAR noise amplitude to add to the range profile and affecting thus the produced final image is also proposed. This is done by adding noise following a Weibull distribution to the High Range Resolution Profiles before SAR processing. This shows an improvement in the classification scores on the MGTD.

Improvement in classification performances can also be achieved by providing extra information to a classifier. Some methods rely on information fusion using additional features on top of the learnt deep learning features, such as a GMM representation of the

SAR image or some texture information [92, 93]. Other methods introduce time in the SAR ATR model by using several images from each SAR image sequence, rather than performing classification on a single image and assume that all images are from the same target. Classification of a group of images (2 to 4 images) is carried out using a multiview deep learning network or a Long Short-Term Memory (LSTM) architecture [20, 94, 95]. The LSTM is a recurrent neural network that has several inputs. The processing is such that the information given by the first images is retained and used to process later images. With this architecture, a score of 99.90% can be achieved on the MSTAR SOC 10 database [94]. However, these results were obtained with group of images with quite different target orientations and that is an unrealistic scenario for a straight and short flight. These scores cannot be directly compared to traditional classification methods which have less information and are based only on a single image of the target.

In Section 2.6.1, it has been shown that the target orientation has a strong influence on the target appearance [19, 96]. An improvement in feature-based classification performances has been shown when a precise target orientation is included in the algorithm in Section 4.4.2. In this chapter, performances are assessed with the inclusion of information about the target orientation. Improvement in classification performances has already been shown in previous work by including orientation knowledge. For example, the orientation was included previously in the loss function for the training of a CNN. In addition to the main objective to optimise in the loss function regarding the target class, a secondary objective is added for the CNN consisting in determining the correct target orientation [97]. Chapter 6 shows that the orientation of the target is learned to a certain extent anyway by the CNN, even without including it specifically in the CNN's loss. In order to take into account the orientation differently, it is proposed to train CNNs that will be specialised in the recognition of targets in a specific range of orientation. This system, called the pose-informed method, assigns the image to be recognised to a CNN trained specifically on images with a target orientation in a similar range once the target orientation has been determined in the test image. It can be imagined that this method could be extended so

that each CNN is trained to handle specific environmental conditions as these can affect deeply the SAR image appearance. The determination of the target orientation required for the pose-informed method is achieved over 360°. 360° orientation determination has been rarely done before [98, 99]. Most methods give an approximation of the orientation modulo 180° [56] or ranges of possible orientation [100]. Some methods are also based on prior class information to obtain a precise target orientation whilst this is not a requirement of the orientation determination proposed in this thesis [101, 102]. An alteration to the traditional Hough transform method is proposed that reduces the number of 90° errors by studying the direction of the target intensity, giving a 180° angle information. Training is then required in order to determine the final orientation of the target with a CNN that determines the front from the back of the target. The proposed method improves the precision of orientation determination compared to former methods.

In order to evaluate the pose-informed method, a fairer result representation is proposed for classification results on small datasets. Indeed, in a small training dataset and with an even smaller set of images used for validation of the model, it is possible that several models reach the same best validation score. However, their results on the testing set can vary substantially. For example, the random initialisation made on the last layers of the CNNs can impact the performance. With this in mind, the results of several trained model achieving the same high validation score are presented. The worst and best results on the testing set are given for the networks achieving the highest score on the validation set.

Firstly, this chapter introduces a classical AlexNet trained using transfer learning on the MSTAR database and MGTD to have a comparison baseline. Then, the benefit of a classical translation data augmentation as well as an innovative Weibull noise based data augmentation is shown. This is followed by the presentation of the orientation determination combining a Hough transform, simple checking for 90° error and a CNN. The pose-informed model is then investigated and it is shown to provide better results than the standard CNN on 4 of the 5 datasets it is tested on.

5.3 Deep learning approach with classical architecture

5.3.1 CNN used

The CNN used as a baseline for comparison, in this thesis is an AlexNet [103]. The network architecture list of layers is presented in Table. 5.1 and the role of each layer has been presented in more detail in Chapter 3. This architecture was chosen as it has been adapted with success for a variety of other applications already such as human pose estimation, video classification or semantic segmentation [104–106]. The architecture of AlexNet is straightforward compared to more recent networks, such as ResNet, GoogleNet [107, 108]. The number of weights is lower than for deeper models such as VGGNet [109]. A simple architecture was selected for ease of implementation and to benefit from the availability of pre-trained models. Other pre-trained models have more recently been added in Matlab version R2019a. The effectiveness of the architecture on its own is not the prime focus of this study as the main goal is to investigate if performances of a chosen network can be improved using the pose-informed solution proposed and evaluated in the end of this chapter. Thus the AlexNet is deemed a good candidate with its relative simplicity and contained number of weight.

5.3.2 Training

In order to reduce training time and compensate for the low number of images available compared to usual deep learning training needs, a transfer learning from a pre-trained AlexNet on the ImageNet [110] to the appropriate SAR or ISAR database presented in Chapter 2. The training method used is the stochastic gradient descent with momentum method described in Chapter 3. The network is firstly trained on a visual database and it, and especially the later layers, is retrained on the SAR data.

Layer	Type	Channel number	Activation size
1	Input	3	227×227
2	Convolution	96	11×11×3
3	ReLu	-	-
4	MaxPooling	-	3×3
5	Convolution	256	5×5×96
6	ReLu	-	-
7	Maxpooling	-	3×3
8	Convolution	384	3×3×256
9	ReLu	-	-
10	Convolution	384	3×3×384
11	ReLu	-	-
12	Convolution	256	3×3×384
13	ReLu	-	-
14	Maxpooling	-	3×3
15	Fully connected	9216	4096
16	ReLu	-	-
17	Dropout 34%	-	-
18	Fully connected	4096	4096
19	ReLu	-	-
20	Dropout 30%	-	-
21	Fully connected	4096	Number of classes (3, 4, 10)
22	Softmax	-	-
23	Output class	-	-

Table 5.1: Layers of the AlexNet inspired CNN for image classification.

Parameter	Value
Learning rate (up to layer 9)	$1 \cdot \lambda_0$
Learning rate (after layer 9)	$6 \cdot \lambda_0$
Learning rate (last layer)	$12 \cdot \lambda_0$
Initial learning rate λ_0	MSTAR: 8.10^{-5}, MGTD: $1.2.10^{-5}$
Epochs number	75
Learning rate dropping rate	0.75
Number of periods before dropping the learning rate	7
Batch size	15

Table 5.2: AlexNet training parameters.

Choice of the parameters

The majority of the parameters were first initialised with probable values and then refined with a human-guided search. As the learning rate is one of the most sensitive training

parameters for a CNN, it was extensively researched with a study of the classification rate achieved after 5 epochs in a grid search. If the learning rate is too low, the CNN is not able to learn the correct weights but if the learning rate is too high, the weights cannot settle in the minimum and the loss can increase. A decaying learning rate is chosen so that the learning rate diminishes as the loss becomes closer to the optimum. The learning rate is different for the various layers of the network. Indeed, for transfer learning, the training is mainly focused on the deepest layers. As the network is from a different modality, the lower layers still needed light training. Layers higher than layer 9 have a higher learning rate with the highest learning rate being for the last layer. The learning rate λ_0 from which stem the learning rates in each layer is researched extensively using a random grid search [111]. The classification score on the training, validation and testing set are drawn after 5 epochs according to the changing learning rate to choose the most appropriate learning rate. An epoch consists of a period of training during which all the training images have been through the network once as seen in Section 3.1.2. These scores are represented in a graph using a logarithmic scale as the learning rate varied greatly between several cases.

The initialisation of the weights in the untrained last fully connected layer of the network is a Gaussian with a 0 mean and a 0.01 standard deviation. A Xavier initialisation was also tested without improving the classification score [112].

The values of the training parameters are summed up in Table 5.2.

Investigation of the learning rate The learning rate λ_0 for the AlexNet for the MSTAR database is chosen to maximise the accuracy of the validation set obtained after 5 epochs as shown in Fig. 5.1. Similar plots for the EOCs introduced in Section 2.4.2 are represented in Figs. B.1 to B.3 can be found in Appendix B. The retained value is $\lambda_0 = 8.10^{-5}$. The accuracy drops for smaller values as it would require too many epochs before reaching the optimum. The weights could also settle in a local minimum. For higher values, the updates of the weights can become divergent, unable to stabilise for an optimal loss.

The same process is done on the MGTD introduced in Section 2.6. The learning rate

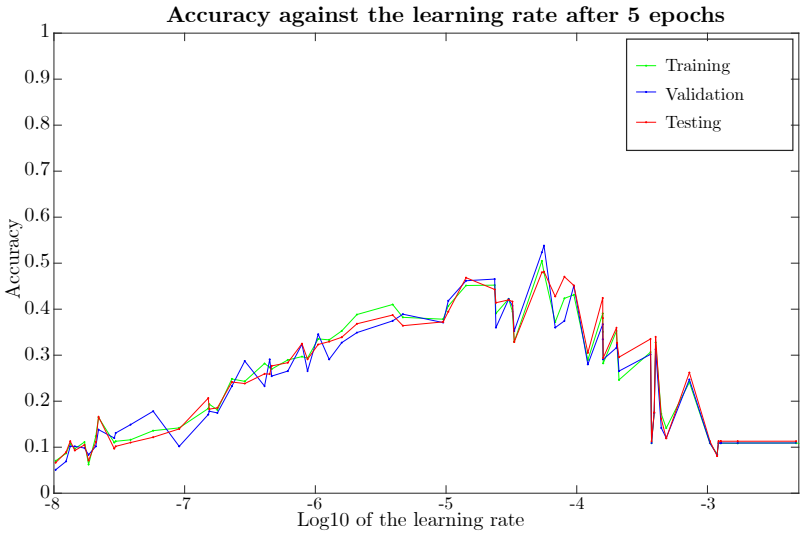


Fig. 5.1: Learning rate study for the SOC 10 MSTAR dataset.

is chosen so that the classification performance is the highest on the validation set after 5 epochs as shown in Fig.5.2. The retained learning rate is $\lambda_0 = 1.2.10^{-5}$.

In addition, it can be seen by looking at the score difference between the validation set and the training set in Figs. 5.1 and 5.2 that the MGTD has a more challenging testing set than the MSTAR SOC 10 dataset. The EOC 1 dataset has also a drop between the validation and testing set but to a lesser extent in Fig.B.1. Figs. B.2 and B.3 are less interpretable as the results in the testing set seem to be less representative of the results in the training and validation sets.

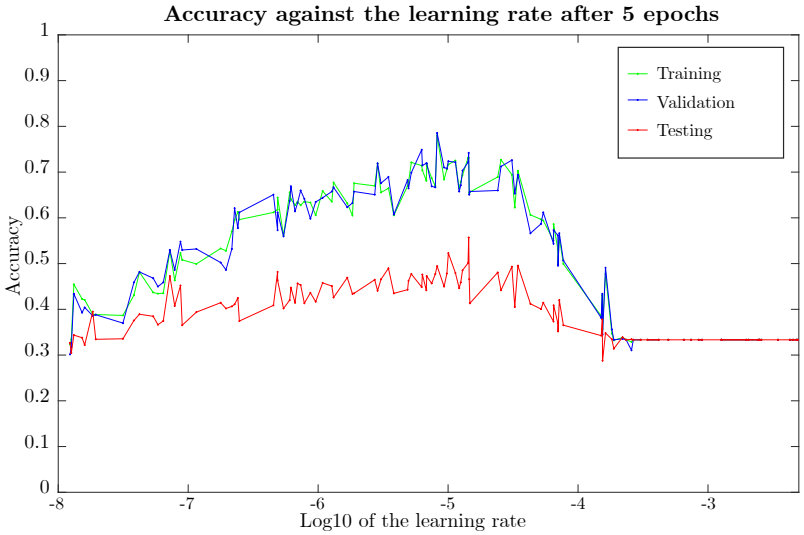


Fig. 5.2: Learning rate study for the MGTD.

Transfer learning

Once the AlexNet was trained on ImageNet, the last fully connected layer (layer 21) is replaced to bring down the 1000 output classes to the number of classes in the studied dataset [103, 110]. Using the parameters shown in Table 5.2, the CNN is retrained and a smoothed evolution of the loss can be seen in Fig.5.3. Similar losses can be seen for the other datasets.

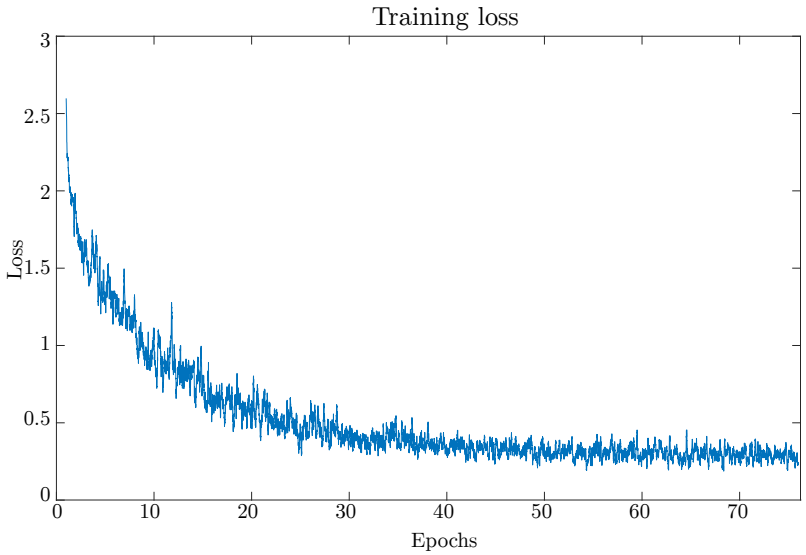


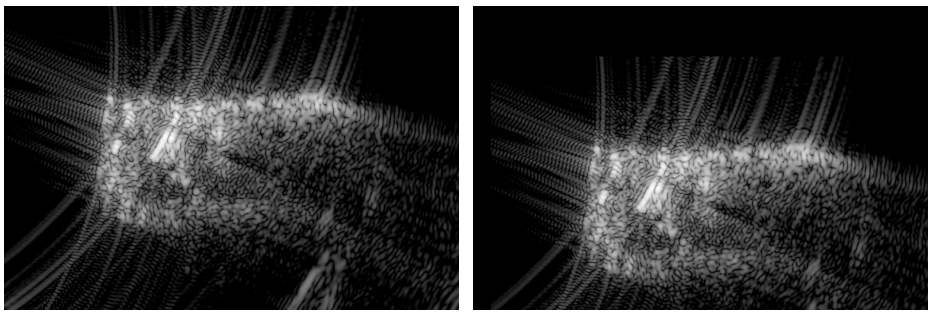
Fig. 5.3: Evolution of the training loss during SAR training on the MSTAR SOC 10 dataset with a turning target.

Data augmentation

Deep learning performance is significantly dependant on the number and variety of the training images. An extensive dataset with a more diverse representation of the object will reduce the chances of overfitting to the data and provide better classification rates [113]. As SAR images require more time and means to be obtained, there are far fewer images and datasets in the SAR domain than in the visual domain. This problem can be partially tackled with data augmentation which consists in adding new images to the training set created by deforming original images from the training set and thus provides more training examples. This approach is commonly used in the visual domain and little investigated in the SAR domain. The data augmentation solutions investigated in SAR include

mainly classical data augmentations used in the visual domain such as cropping images, translating the image so that the target appear at a different location, or rotating targets so their orientation change [77, 90, 91, 114, 115]. Some data augmentations are also specific to the SAR domain and provide an improvement of performance. For example, the colour jitter approach consisting in adding Gaussian noise to an image in the visual domain, is transposed to SAR with the addition of multiplicative noise following an exponential distribution to the image [91]. Another method consists in rescaling the images to distort the target and simulate variations in depression angle and target configuration [116].

This section investigates a classical data augmentation but also proposes a method that simulates close to realistic noise during the acquisition of the signal to provide additional noisier images than original SAR images. This is achieved by adding noise amplitude following a Weibull distribution to the range profiles before image formation. The Weibull distribution has been shown to be a good fit to model the multiplicative speckle noise in the MSTAR images [57]. Similar parameters are chosen. The range profiles used show the amplitude of the signal and not the complex IQ form. The resulting images with noise supplement the training set and thus may improve the robustness of the trained deep learning architecture.



(a) Original image of a T72. (b) Translated image.

Fig. 5.4: Classical data augmentation in the visual domain.

Classical data augmentation Translation has been used as a classical data augmentation method. The input images are firstly resized to 227×227 pixels and repeated on 3 channels as it is the required input size of the Alexnet. Each image is then translated

of a vector $[x,y] \in \mathbb{R}^2, [x,y] \in \big[[-100;100],[-100;100]\big]$. In practice, this means that the target is always entirely in the image even if the target was not exactly centred to begin with. The areas of the images that are not assigned to a value are zero-padded to retain the original size of the image as can be seen in Fig.5.4. The translated images are added to the original training set.

Range profile data augmentation on the MGTD

Range profile data augmentation requirements The proposed method requires access to the range profiles used to generate the image which are not given in the MSTAR dataset, which only provides the amplitude of the image samples. Because it remained impossible to retrieve the complete range profile, the proposed noise based data augmentation is only applied to the MGTD.

Noise based data augmentation In addition to the traditional data augmentation, a SAR specific data augmentation is proposed. This technique relies on the artificial addition of noise to the radar returns before processing the ISAR images to replicate a noisier acquisition [9]. The objective of choosing realistic noise for data augmentation is to make the deep learning method less sensitive to perturbations inherent to the SAR data such as speckle noise. The noise that will be added to the signal requires thus to follow a distribution close to that of the speckle found in SAR images. Various probability distributions have been studied to represent the amplitude of speckle noise such as the Rayleigh distribution, K-distribution and Weibull distribution [57, 117–119]. In this chapter, a Weibull distribution with a shape parameter of ~ 2 close to a Rayleigh distribution to provide a representation of SAR speckle is selected [57, 120]. The associated probability density

function of the Weibull distribution is shown in Eq. (5.1).

$$\begin{cases} p(x,\beta,\eta) & = U(x)\cdot\frac{\beta}{\eta}\cdot\left(\frac{x}{\eta}\right)^{\beta-1}\cdot e^{-(\frac{x}{\eta})^{\beta}} \\ \beta \approx 2 & \\ \eta > 0 & \end{cases} \tag{5.1}$$

where U is the step function, β is the shape parameter, and η is the scale parameter. If $\beta = 2$, the distribution is a Rayleigh.

Data augmentation type	Parameters			
	Shape β	Scale η	Weighting a	Weighting b
Noise 1	2.15	0.75	8e-8	1.01
Noise 2	2.15	0.75	1e-7	1.007
Noise 3	2.15	0.75	5e-7	1.01
Noise 4	2.15	0.75	7e-1	1.008

Parameters are chosen empirically.

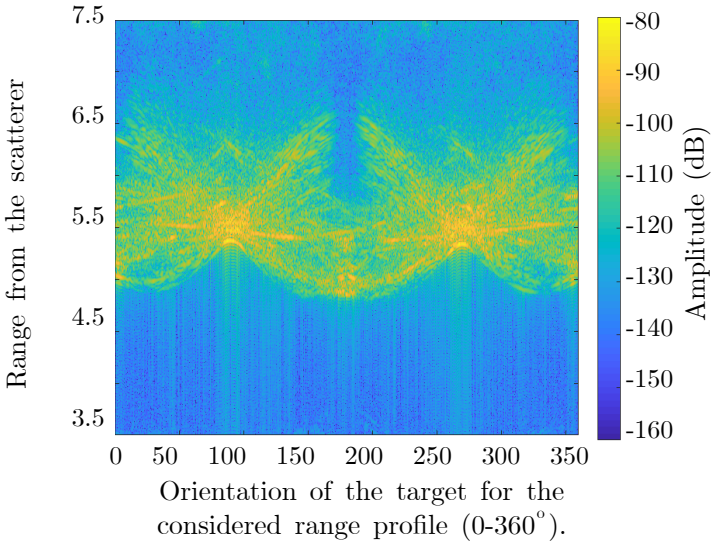
Table 5.3: Weibull based noise parameters

The range profiles of the target are obtained with an inverse Fourier transform of the Vector Network Analyzer (VNA) returned frequency signal. The noise is then added to the range profiles. Multiplicative and additive noise following a Weibull distribution are applied to each range profile for each orientation of the target as in Eq. (5.2).

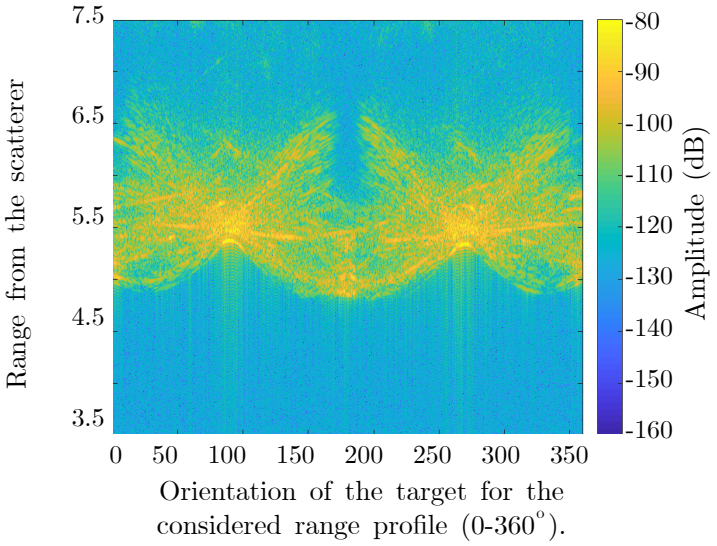
$$\mid s'(m,\tau_n) \mid = a\cdot X(m) + (1 + b\cdot Y(m)) \mid s(m,\tau_n) \mid \tag{5.2}$$

where s' and s are respectively the modified and original range profile, m is the range bin, τ_n is the n^{th} pulse, $X(m)$ and $Y(m)$ are random variables following a Weibull distribution, a and b are weighting constants. All the parameters used to create the 4 additional sets of images are summarised in Table 5.3.

An example of a modified range profile with noise in Fig.5.5 (b) with the lowest additive Signal to Noise Ratio ($SNR_{Additive}$) (Noise 4 in Table 5.4) can be compared



(a) Original range profile.



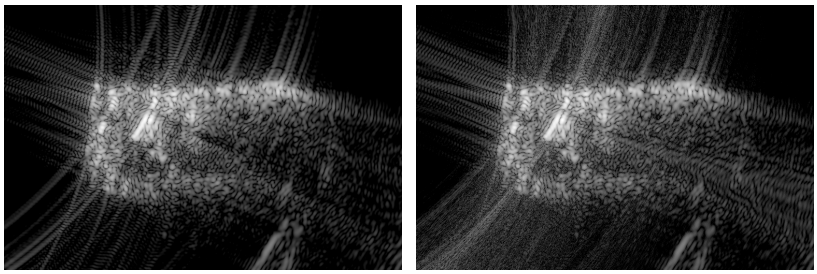
(b) Range profile after the noise addition.

Fig. 5.5: Effect on the range profile of the Weibull based data augmentation on the MGTD. SNR details associated with noise 4 in Table 5.4

to the clean original range profile in Fig.5.5 (a). The computation of $SNR_{Additive}$ and $SNR_{Multiplicative}$ is detailed in Section 5.3.2. Once the range profile is updated, the standard backprojection algorithm is resumed to create the target images [14].

The parameters chosen for the Weibull distributions of the noise are summed up in Table 5.3 and change the impact of the noise on the SAR image. The choice for the parameters of the noise distribution is made to transform the original image while still keeping the target image interpretable. The effect of the noise data augmentation on

the final target image can be seen in Fig.5.6 in which both the original image and the image issued from data augmentation with the noise associated with the lowest additive $SNR_{Additive}$ (Noise 4 in Table 5.3) are shown.



(a) Original target image. (b) Target image after the noise addition.

Fig. 5.6: Effect on the target image of the Weibull based data augmentation.

Impact measurement of the artificial noise on the original images In order to evaluate the noise already present in the original data and compare the different noise based data augmentations, the SNR is calculated in each case. The SNR of the unmodified data is estimated on the range profiles. The strongest peak in the range profile represents the power of the signal while the average range profile value between 3.5m and 4.3m where no target is present is used to determine the mean noise power. The signal and noise power issued from the range profile gives a SNR of 57 dB for the original signal.

The SNR relative to the artificial noise does not account for the noise already present in the original signal. Each noise power is calculated independently for the multiplicative and the additional noise following Eq. (5.3).

$$[h]\begin{cases} P_{Signal} = \frac{1}{m}\sum_{m=3.5}^{7.5}|s(m,\tau_n)|^2 \\ P_{Additive} = \frac{1}{m}\sum_{m=3.5}^{7.5}(a.X(m))^2 \\ P_{Multiplicative} = \frac{1}{m}\sum_{m=3.5}^{7.5}|b\cdot Y(m)\cdot s(m,\tau_n)|^2 \\ SNR_{Additive}(dB) = 10\cdot\log(\frac{P_{Signal}}{P_{Additive}}) \\ SNR_{Multiplicative}(dB) = 10\cdot\log(\frac{P_{Signal}}{P_{Multiplicative}}) \end{cases} \tag{5.3}$$

where P_{Signal}, $P_{Additive}$, $P_{Multiplicative}$ are respectively the signal, additive noise and multiplicative noise power, m is the range bin, τ_n is the n^{th} pulse, a and b are weighting constants, and X and Y are Weibull distributions. $SNR_{Additive}$ and $SNR_{Multiplicative}$ refers to the SNR related to respectively the additive and multiplicative noise.

Table 5.4: SNR for the additive and multiplicative noise augmentations relative to the original signal.

Data augmentation type	Additive noise $SNR_{Additive}$ (dB)	Multiplicative noise $SNR_{Multiplicative}$ (dB)
Noise 1	18	2
Noise 2	16	2
Noise 3	2	2
Noise 4	-1	2

The resulting SNR for the various parameter combinations are given in Table 5.4. Typical SNR for target detection in SAR ranges from 10 dB to 20 dB [121–123]. The SNR in images in the MSTAR is in this range with an average value of 16 dB for all images in the MSTAR SOC 10. This SNR value was calculated with the signal power being the maximum intensity on the target while the average intensity in a wide zone in the background further from the target was taken as the noise power. Some chosen SNR values for the data augmentation are in this range and there are also lower values so that the image classification is more challenging. Indeed, lower classification scores have been reported for a SNR below 5dB [60].

Impact of data augmentation on the classification results

Data augmentation on the MSTAR datasets The influence of the classical translation data augmentation is not clear on the MSTAR datasets. The results achieved on the various MSTAR datasets for the CNN trained on the plain training set are compared to the results achieved with the CNN trained with additional images created with translation data augmentation as seen in Table 5.5. The results are close but different than the

results obtained in Section 5.4.5 because of the random initialisation used and the parameters tuning that was less refined when these results were obtained than in Table. 5.2, and achieved slightly lower scores. The translation data augmentation improves by 1.07% for the MSTAR SOC 10 and by 5.36% for the MSTAR EOC 2, however, the classification scores lowers by 8.77% or by 0.14% respectively on the MSTAR EOC 1 and EOC 3. If the improvement due to robustness is understandable, the deterioration of the score shows that it could also appear beneficial to let the network overfit the data, showing the resemblance of the testing and training set. After some investigation, targets were found to be slightly better centred in the MSTAR SOC 10 and EOC 2 than in the MSTAR EOC 1 and EOC 3 in the testing set.

To show the importance of robustness, the scores of a network trained on plain data are compared to the scores of the network trained with additional translated data (the same presented in Table 5.5 on an altered testing set). The target is moved by a random distance of -6 m to 6 m, equivalent in the image to a -20 to 20 pixels translation in both x and y directions. The influence of overfitting is shown in Table 5.5. The results show a very sharp drop in the accuracy of the networks trained without translation data augmentation apart in the MSTAR EOC 1 dataset. This drop in accuracy goes from 5% in the best case up to 43% in the worst case. The networks trained without the translation data augmentation rely thus heavily on the precision of the target detection.

For the cases studied in this thesis, the added translation data augmentation during training provides more robust results for real applications. It indeed improves the results over the majority of the MSTAR dataset, MSTAR SOC 10, EOC 2 and EOC 3, with the exception of the MSTAR EOC 1 dataset. It is thus kept for further investigation in the rest of the thesis. These results also show that choosing a network solely on one classification score could be harmful, especially in the SAR domain for which the datasets are small and less representative of the possible wide variety of images acquired in different scenarios. Indeed, images in SAR testing sets are less diverse than in visual testing dataset because of the acquisition difficulty. In the case of ImageNet, the 40 million images can be from

Dataset	Testing data	Classification score of the CNN trained on plain training	Classification score of the CNN trained with translation data augmentation
MSTAR SOC 10	Plain data	96.70%	97.77%
MSTAR SOC 10	Data with a target randomly translated	53.86%	97.07%
MSTAR EOC 1	Plain data	83.49%	74.72%
MSTAR EOC 1	Data with a target randomly translated	78.02%	74.63%
MSTAR EOC 2	Plain data	87.95%	92.57%
MSTAR EOC 2	Data with a target randomly translated	51.81%	92.34%
MSTAR EOC 3	Plain data	85.61%	85.46%
MSTAR EOC 3	Data with a target randomly translated	51.73%	86.94%

Table 5.5: Robustness test of the trained AlexNet against the translation of the target in the testing set.

various sources as they are directly drawn from internet searches using chosen keywords, from different locations, from different angles [110]... On the other hand, SAR and ISAR dataset were taken with a single system for each dataset with less environmental changes. Moreover, they are composed of images in sequences that are not independent as the are taken over a short period of time in a scene with little change. Thus, robustness is not totally taken into account even in the more complicated MSTAR EOCs dataset. Robustness of the CNNs is not a standard test on the MSTAR dataset even if algorithms perform less when confronted to occlusion, noise or translation [91, 92, 124, 125]. Standard robustness testing would thus be interesting for SAR ATR methods.

Results of the range profile data augmentation on the MGTD The classification scores benefit greatly from both the translation and noise based data augmentation as seen in Table 5.6. This sharp increase (between 10% and 15%) shows that data augmentation can tackle the low number of training images and their lack of diversity which weakens the CNN training. In the MGTD with a rotating target, there is little target translation, or overlay. Thus, using data augmentation dedicated to these changes has a limited impact.

However, all SAR images suffer from noise and speckle to a certain extent. Noise is thus a compelling variable to take into account during training. The simulation of speckle for data augmentation resulted in more resilient CNNs and improved by 5% the total classification score over the CNN trained with translation data augmentation alone.

Training set	Classification rate
No data augmentation	76,85%
Training set with translation data augmentation	86,34%
Training set with translation and Weibull noise data augmentation	91,20%

Table 5.6: Influence of the data augmentation to the CNN classification rate on the MGTD

Further work related to data augmentation Further work with the Weibull noise based data augmentation could lead to new denoising methods. Denoising SAR images is currently achieved with filters relying on statistical assumptions on speckle [126, 127]. Denoising has now been approached in other domains such as pre-processing for image recognition or voice recognition with decoders [128, 129]. The decoders are networks that are trained using purposefully noisy data in input to reconstruct the original clear data. The noisy inputs are created using real or synthetic noise.

Once trained, they are able to denoise data with real noise as long as this noise is similar to the simulated noise during training. This process could also be applied to SAR images using the images created with the Weibull noise data augmentation as it is a realistic noise for SAR data. In this work, the noise is added to the amplitude of the range profile which makes it less realistic, but a good first approach. The result could be a decoder neural network able to denoise SAR images. This system could be then compared to more standard filters.

5.3.3 Classification results for the classical CNN architecture

The AlexNet is trained with the parameters described in Section 5.3.2. The networks trained on MSTAR datasets benefit from translation data augmentation while the net-

Image set	Classification rate
Training set	99.71%
Validation set	99.27%
Testing set	97.77%

Table 5.7: Classification scores on the MSTAR SOC 10 for the AlexNet.

	2S1	BMP	BRDM	BTR60	BTR70	D7	T62	T72	ZIL	ZSU	Score (%)
2S1	255	0	0	0	0	0	19	0	0	0	93.07
BMP	0	185	1	0	0	0	0	9	0	0	94.87
BRDM	1	3	267	0	0	0	0	3	0	0	97.44
BTR60	0	0	1	192	2	0	0	0	0	0	98.46
BTR70	0	0	2	1	193	0	0	0	0	0	98.47
D7	0	0	0	0	0	273	1	0	0	0	99.64
T62	0	0	0	0	0	3	269	0	1	0	98.53
T72	0	1	0	2	0	0	0	193	0	0	98.47
ZIL	0	0	0	0	0	0	0	0	274	0	100
ZSU	0	0	0	0	0	4	0	0	0	270	98.54

Table 5.8: Confusion matrix on the MSTAR SOC 10 for the AlexNet.

Image set	Classification rate
Training set	100%
Validation set	100%
Testing set	74.72%

Table 5.9: Classification scores on the MSTAR EOC 1 for the AlexNet.

	2S1	BRDM	T72	ZSU	Score
2S1	288	0	0	0	100%
BRDM	0	285	0	2	99.30%
T72	195	12	50	31	17.36%
ZSU	51	0	0	237	82.29%

Table 5.10: Confusion matrix on the MSTAR EOC 1 for the AlexNet.

Image set	Classification rate
Training set	99.8%
Validation set	100%
Testing set	92.32%

Table 5.11: Classification scores on the MSTAR EOC 2 for the AlexNet.

	BMP	T72	BRDM	BTR70	Score
BMP	680	102	17	58	79.35%
T72	62	2624	11	15	96.76%

Table 5.12: Confusion matrix on the MSTAR EOC 2 for the AlexNet.

Image set	Classification rate
Training set	99.55%
Validation set	98.99%
Testing set	85.46%

Table 5.13: Classification scores on the MSTAR EOC 3 for the AlexNet.

	T72	BMP	BRDM	BTR70	Score
T72	2316	322	51	21	85.46%

Table 5.14: Confusion matrix on the MSTAR EOC 3 for the AlexNet.

work trained on the MGTD benefits from both the translation data augmentation and the Weibull noise data augmentation presented in Section 5.3.2. Each of these networks are trained using 90% of the training sets presented in Chapter 2, while the remaining 10% is used for validation to evaluate the various networks and choose those performing best. The network with the highest score on the validation set is retained and the final classification score given in the tables is the classification rate achieved on the testing set. A sum up of the various scores attained is given in Tables 5.7, 5.9, 5.11, 5.13 and 5.15 for each dataset. The details of the classification rates achieved on the testing set can be seen in the respective confusion matrices shown in Tables 5.8, 5.10, 5.12, 5.14 and 5.16. The AlexNet network achieves only 97.77% on the MSTAR SOC 10 dataset, which is lower than the state-of-the art algorithms achieving scores over 99% [21, 92, 95, 97, 116]. However, it still achieves scores comparable with simpler CNNs architectures on unprocessed images [73, 130] and even compares to some network using multiple views of the same target to achieve classification on the MSTAR SOC 10 [20]. The performance on the EOCs is poorer than what has been reported with scores reaching over 94%, 95%, 98% respectively for the EOC 1, EOC 2 and EOC 3 [20, 21]. However, the scores reported on the EOC datasets were obtained with more complex architectures or used several images

for classification. This could be due to the network having a less effective training on the smaller set of image (4 sequences compared to 10 sequences for the SOC alternative dataset.). Because the validation set is made of images from the same sequence as the training images, the overfitting of the network to the training data with a deterioration of the accuracy on validation was not significant but still had an impact on the scores related to the testing set. Some networks with lower validation scores achieved higher scores on the testing set than the CNNs with the highest scores on validation, which supports the above hypothesis. They are though not retained, because there is no mean to discern them before testing.

There is not yet any literature on the MGTD. A simple NN method was tested in Section 2.6.3 with a classification score of 71.80%. Using a CNN improves this score significantly as seen in Table 5.15.

Image set	Classification rate
Training set	100%
Validation set	100%
Testing set	81.20%

Table 5.15: Classification scores on the MGTD for the AlexNet.

	BMP1	T64	T72	Score
BMP1	259	10	23	88.69%
T64	14	257	21	88.01%
T72	10	14	268	91.78%

Table 5.16: Confusion matrix on the MGTD for the AlexNet.

5.4 Deep learning approach with pose informed architecture

Sensitivity of SAR images to environmental changes is a key challenge for SAR ATR algorithms. Changes of the acquisition scenario can modify the SAR image substantially

and lead to a drop of the classification rate [21, 26, 131]. It is therefore important to adapt the ATR solution so that changes in viewing conditions are taken into account. One option is to optimise, during training, the loss caused by target orientation in addition to the loss related to the target class, so that the CNN is forced to account for the influence of the target aspect angle [97]. Alternatively, artificial images that are particularly prone to misclassification by the CNN can be introduced to improve its robustness against mistakes [116]. This approach is sufficient for few variables, but a high number of variables cannot be addressed effectively in this manner. Our approach relies on the determination of the viewing conditions before choosing the most adequate trained CNN for target classification. The technique presented here focuses on the orientation of the target but could be extended to more diverse conditions as long as enough examples with similar conditions are present and referenced in the training set. Several other parameters, e.g. the depression angle or layover [26] could be chosen over the orientation for the neural network to focus on. However, it has already been seen that the orientation has a great impact on the classification rate in Chapter 4. The orientation is well distributed and its value is known in the various dataset. Thus, orientation is chosen as the main influence of the pose-informed architecture.

5.4.1 Pose informed architecture

In this chapter, the pose-informed deep learning architecture summarised in Fig.5.7 is proposed to handle target classification [10]. The first step consists in the determination of the target orientation using a Hough transform and a CNN, as described in Section.5.4.2. Once the orientation CNN is found, the appropriate pose-informed CNN is used to determine the target class, as explained in Section 5.4.3. This pose-informed CNN is specifically trained on an orientation range that includes the predicted orientation. There are n pose-informed CNNs, each trained on $\frac{360}{n}$ degree range of orientation, with n between 2 and 8.

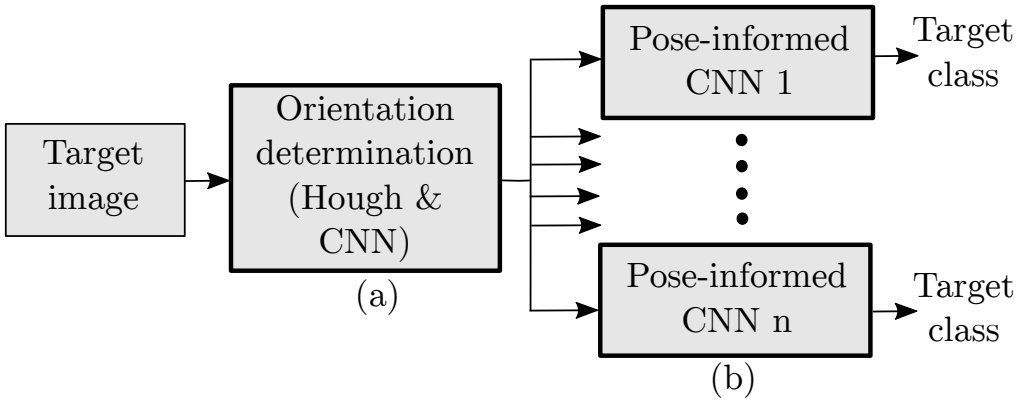


Fig. 5.7: Overview of the pose-informed architecture.

5.4.2 Processing of the image set prior the pose informed deep learning method

Determination of the orientation of the target

The determination of the target orientation is the first-step of the pose-informed method. To determine the orientation modulo 180° of the target, several methods can be used. Statistical methods are precise, however they require training which introduces additional randomness to the process, for example the initialisation of an expectation-maximisation algorithm [99, 101, 102]. Moreover, the values of the estimated distributions are target specific. This approach requires the target classification task to be carried out before the orientation determination. For these reasons, a direct pose estimator was selected, at the expense of a small precision loss. Methods not relying on the evaluation of a statistical distribution have already been investigated, for example by estimating a bounding box or a Hough transform considering several poses [56]. Here, a CNN is used to differentiate the front from the back of the target.

An alternative solution could be a CNN that handles completely the orientation determination. A CNN on the AlexNet model was created with a regression layer to that end, but without success. As the discontinuity at 0° and 360° complicated the loss, another CNN was tested with two outputs representing the cosine and sine of the orientation angle so that the loss could be continuous. This solution did not give good results either. It

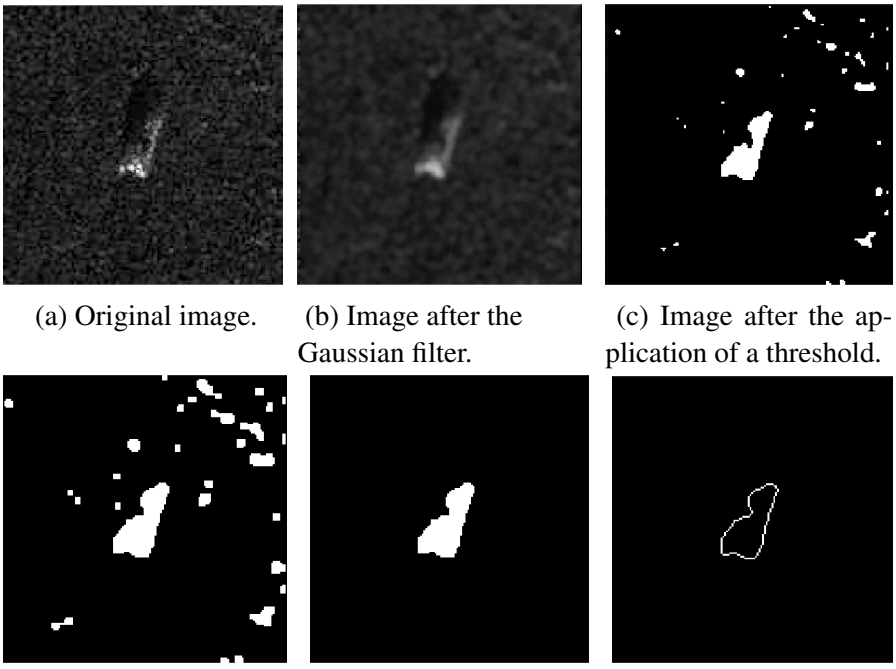
could be because each output independently could be associated with two target orientations. Some deep learning methods have though some success retrieving the orientation of text in images in the visible domain [132]. The proposed CNN gives an orientation modulo 180°. This methods relies however on the determination of an encapsulating box on the zone of interest that works well for text but has been shown to be a rather poor SAR target orientation determination (standard deviation of 14.02° against 8.12° for the Hough transform in [56]).

The orientation is found in two steps. Firstly, the orientation is determined modulo 180° with a Hough transform by relying on the rectangular shape of the target. Once the image is rotated with the determined angle, the image contains a horizontal target. This rotated image is fed to the CNN which determines the direction of the target by recognising the front from the back of the target. With these two steps, the full 360° orientation of the target can be determined. The image of the target can then be analysed by the appropriate pose-informed CNN in Figure 5.7 (b).

As the CNN uses the full images and not the segmented image, it is not worth investing in a computer expensive method such as the GMMs (Section 4.4.1) if the precision is only slightly degraded by using a simpler segmentation method. In this section, the pose estimation is investigated using a simpler segmentation method and improving the orientation determined with a Hough transform by using prior knowledge that targets have a rectangular shape.

Method for pose estimation

Simple segmentation The objective of the segmentation for pose estimation consists in extracting a precise contour of the target so that the target orientation can be accurately estimated. In SAR images, one or two edges of the target are usually well defined. Once detected with the Hough transform, the longest straight edge sets the target orientation. The segmentation process starts by applying a Gaussian filter to smooth the picture and obtain a simpler target shape to segment. The impact of the Gaussian filter



(a) Original image. (b) Image after the Gaussian filter. (c) Image after the application of a threshold.

(d) Image after morphological filtering and hole filling. (e) Selection of the biggest blob representing the potential target. (f) Contour of the potential target.

Fig. 5.8: Contour acquisition of the target via segmentation.

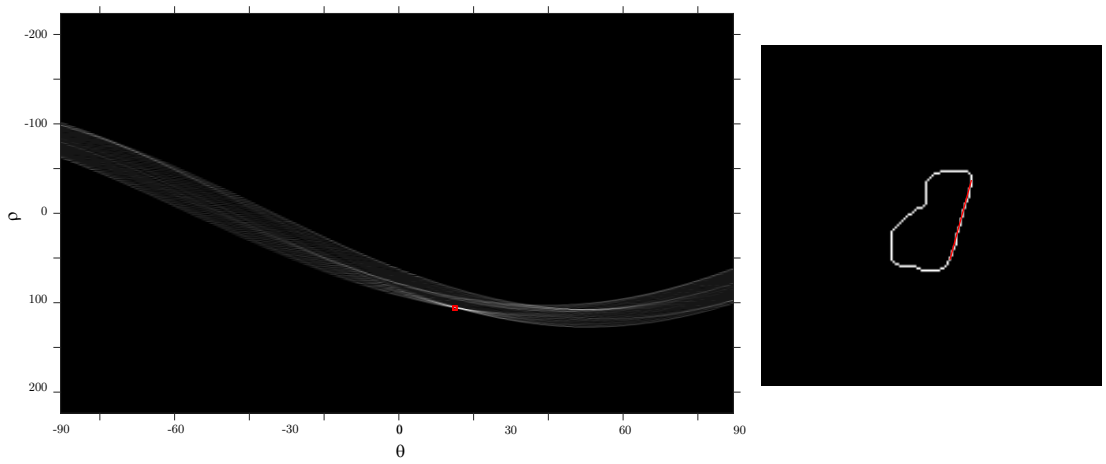
on a sample original image (Fig.5.8 (a)) can be seen in Fig.5.8 (b). After that, the image is binarised using a threshold. This threshold is chosen to keep only 65% of the brightest pixels by computing the intensity cumulative distribution in images from the MSTAR database. This is moved up to 88% in the MGTD as the target occupies more space. These percentages were determined empirically and are dependant on the size of the target regarding its total SAR image area occupancy. The resulting binary image can be seen in Fig.5.8 (c). To smooth the edges of the target, morphological filtering is applied. Two steps of dilation and one of erosion give the result shown in Fig.5.8 (d). Lastly, the smaller blobs are suppressed to keep only the biggest as shown in Fig.5.8 (e) before extracting its contour as shown in Fig.5.8.

Hough transform applied to the SAR images Once the image is segmented, a Hough transform is applied to the target contour. Only one peak, the brightest of the resulting matrix is kept (Fig.5.9 (a)). It corresponds to the longest line that can be su-

where

α	is the orientation tolerance to choose the contributing peaks,
θ_0	is from the polar representation ρ_0, θ_0 of the main orientation line.
$P_i(\rho_i, \theta_i)$	is one of the considered extra peak points whose intensity is over an experimentally determined threshold T. This line has a polar representation of ρ_i, θ_i,
$\theta_{0,\parallel}, \theta_{0,\perp}$	are the orientations of the lines coherent with the orientation estimation θ_0. These lines are either on the same or parallel edge of the target or on the perpendicular edges.

perimposed on the target contour in Fig.5.9 (b). The orientation of the line is kept as the orientation of the target.

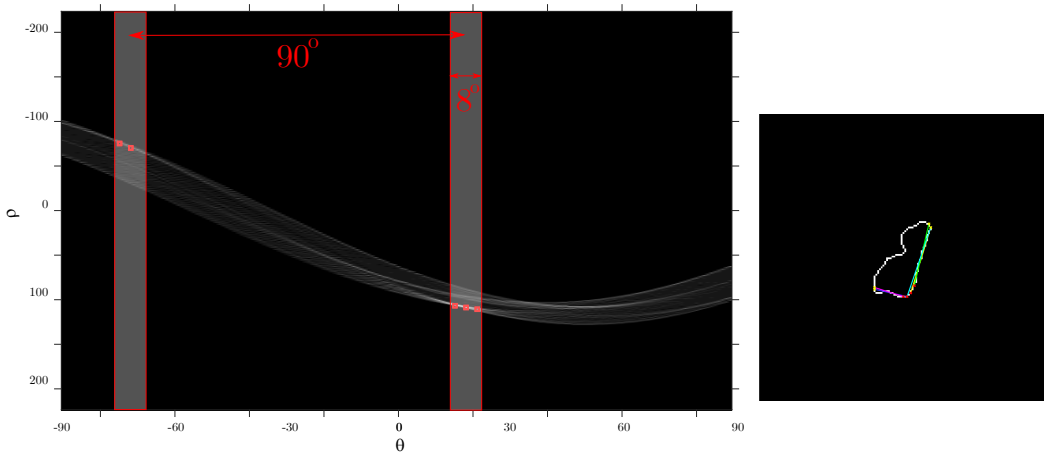


(a) Matrix resulting from the Hough transform on the contour Fig.5.8 (f) and selection of the brightest peak.

(b) Resulting longest line on the target contour.

Fig. 5.9: Direct application of the Hough transform to find the target orientation.

Averaged Hough transform The proposed averaged Hough transform takes into account the rectangular shape of the target. The Hough transform is firstly applied to the image in order to obtain the ρ, θ matrix. All peaks beyond a certain threshold are considered. This intensity threshold on the Hough transform map was obtained by experience after testing of the orientation determination algorithms. This threshold is the same for all images in all databases and is equal to $0.0005 \cdot \max(H)$, with H being the Hough transform map. Between these peaks, only those with a θ in accordance with the estimated target orientation are kept. These peaks correspond to lines on one of the 4 edges of the target. To be considered as agreeing with the line orientation of the main peak, θ has



(a) Peak selection to compute the averaged target orientation.

(b) Corresponding lines compatible with the rough orientation estimation.

Fig. 5.10: Determination of the lines compatible with a unique target orientation estimation.

to be 4° degrees from the main orientation, or perpendicular of the main orientation. A representation of the peaks considered is summarised in Fig.5.10 (a). All the possible points representing lines that are compatible with the main study point are expressed in Eq. (5.4). The lines corresponding to these peaks in the contour image can be seen in Fig.5.10 (b). The details on the choice of these peaks is expressed in Eq. (5.4).

The rough orientation estimation θ_0 is then refined, using the input of all the peaks in agreement with the rough estimation. The precise estimation is the result of the weighted

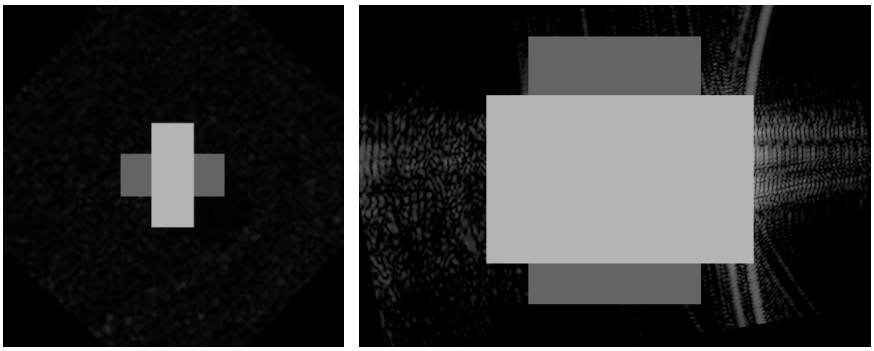
$$
\begin{aligned}
&\alpha = 4, \\
&\theta_i \in \{\theta_{0,\parallel}; \theta_{0,\perp}\} \\
&\theta_{0,\parallel} \in \begin{cases} [-90; \theta_0 - 180 + \alpha] \cup [\theta_0 - \alpha; 90], & \text{if } \theta_0 > 90 - \alpha \\ [\theta_0 - \alpha; \theta_0 + \alpha], & \text{if } \theta_0 \in [-90 + \alpha; 90 - \alpha] \\ [-90; \theta_0 + \alpha] \cup [\theta_0 + 180 - \alpha; 90], & \text{if } \theta_0 < -90 + \alpha \end{cases} \\
&\theta_{0,\perp} \in \begin{cases} [\theta_0 - 90 - \alpha; \theta_0 - 90 + \alpha], & \text{if } \theta_0 > \alpha \\ [-90; \theta_0 - 90 + \alpha] \cup [\theta_0 + 90 - \alpha; 90], & \text{if } \theta_0 \in [-\alpha; \alpha] \\ [\theta_0 + 90 - \alpha; \theta_0 + 90 + \alpha], & \text{if } \theta_0 < -\alpha \end{cases}
\end{aligned}
\tag{5.4}
$$

$$\text{for } \theta_i \in \theta_{0,\parallel}$$
$$-90 \leqslant d_i \leqslant 90, \quad d_i \equiv \theta_0 - \theta_i \pmod{180} \tag{5.5}$$
$$\text{for } \theta_i \in \theta_{0,\perp}$$
$$-90 \leqslant d_i \leqslant 90, \quad d_i \equiv \theta_0 - \theta_i - 90 \pmod{180} \tag{5.6}$$
$$\hat{\theta_0} = \theta_0 + \frac{\sum_{i=1}^{n} L_i.d_i}{\sum_{i=1}^{n} L_i} \pmod{180} \tag{5.7}$$

where

L_i	is the length of the line associated with the peak point i. This line supports the hypothesis of an estimated target orientation θ_0,
$\theta_{0,\parallel}$, $\theta_{0,\perp}$	are the orientations of the lines coherent with the orientation estimation θ_0. These lines are either on the same or parallel edge of the target or on the perpendicular edges,
d_i	is the distance between the orientation of the main orientation θ_0 and the orientation of the contributing line i,
$\hat{\theta_0}$	is the refined orientation estimate of the target deduced from all n contributing lines.

average of all the orientations of the contributing peaks according to the length of the lines they represent. The peak intensity represents the validity of the line comprising of its length and the number of pixels contributing to it. This weighed average is expressed in Eq. (5.7).



(a) Location of the integral images to compute the vertical ratio for the MSTAR dataset images.

(b) Location of the integral images to compute the vertical ratio for the MGTD images.

Fig. 5.11: Location of the integral images to compute the vertical ratio for both dataset.

$$H(I_\theta) = \frac{\sum_{i=c_1-l}^{c_1+l} \sum_{j=c_2-s}^{c_2+s} I_\theta(i,j)}{\sum_{i=c_1-s}^{c_2+s} \sum_{j=c_2-l}^{c_2+l} I_\theta(i,j)} \tag{5.8}$$

where

θ	is the estimation of the orientation of the target,
I_θ	is the result of the rotation of the original image of an angle $-\theta$,
$H(I_\theta)$	is the vertical ratio computed for the image I_θ,
c_1, c_2	are the abscissa and ordinate defining the centre of the image (once the target centred),
s, l	are half the length of respectively the short and long side of the rectangle,

Vertical ratio Because of the difficulty to segment the target with not clearly defined edges and a varying illumination, the long edges are sometimes not detected as lines. These lines can be broken in several parts or not being detected at all if the illumination is focused on one of the small edges. If the small edge is the only edge determined as a line by the Hough transform, the estimated orientation will be off by roughly 90°. In order to limit these errors, the rectangular shape of the target and the prior knowledge that the intensity of the target is higher on average than that of the noise and clutter in all databases. Once the orientation of the target is computed by the Hough transform, the image is rotated to compensate the target orientation. If the orientation is off by 90°, the target will be vertical instead of horizontal. A vertical ratio of the sum of intensities of the pixels contained respectively in two rectangles of fixed size is computed, with either a 0° or 180° direction as seen in Fig.5.11. This ratio is expressed in Eq. (5.8).

After some testing on the training sets, the cut-off value of the vertical ratio is determined to be 1.2 in the MGTD and 1.09 in the MSTAR database. If the vertical threshold is higher than that threshold, it is assumed that the estimated orientation is off by 90° and this value is added to the original orientation estimation.

Results of the pose estimation For both dataset, the absolute error determining the orientation modulo 180° is calculated. In order to better visualise the repartition of errors, the cumulative distribution of these errors is shown in Fig.5.12. For comparison, the same

scores as in Section 4.4.2 are given (i.e. the mean error, the standard deviation, the mean absolute error and the root mean square error). The direct and averaged Hough transform orientation determination are compared with the optional usage of the vertical ratio to discriminate 90° errors.

The averaged Hough transform seems not to have significant advantages with the usage of a vertical ratio. It worked better on the MGTD than on the MSTAR dataset. The orientation estimation in the rest of the chapter will be carried out using the direct Hough transform associated with the vertical ratio.

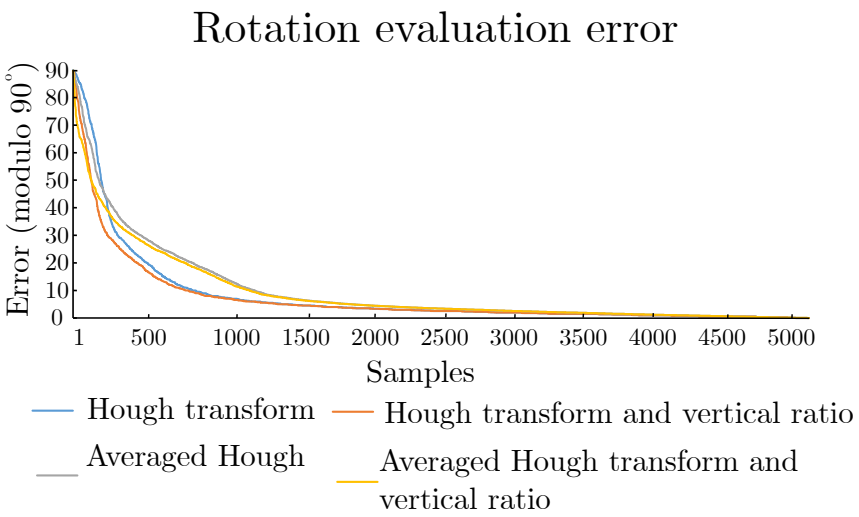


Fig. 5.12: Error distribution of the orientation estimation in the MSTAR dataset.

Pose estimation on the MSTAR dataset The distribution of the absolute error on a compilation of all testing sets in the MSTAR SOC 10, EOC 1, 2, 3 is shown in Fig.5.12.

The averaged Hough transform reduces the errors of over 45°. When strong errors occur, the long edge has not been correctly detected. For example, if a small edge of the target is detected instead of the long edge as it can happens for certain illumination configurations, this adds a 90° error. Specific illumination configuration indeed sometimes makes the longer edges less noticeable and the segmentation can result in a shape closer to a square than a rectangle. In these cases, contributing lines have a higher weight as the main line on the long edge is not very well defined and thus shorter. Averaging reduces the possible errors. For errors in the 0°-45° range, the averaging is found to actually

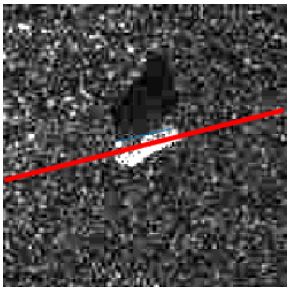


Fig. 5.13: Potential drawback of the averaged Hough transform using the wrong edge of the target.

damage the orientation estimation with a difference between both curves of a maximum of 10°. Contributing lines in the averaged Hough transform can focus on the less illuminated edge of the target, as in Fig.5.13. Because of the segmentation threshold, this edge passes slightly inside the target and is not parallel to the target orientation. This would increase the orientation error. Overall, the averaged Hough transform is only interesting for the biggest errors. Further work could be done in order for the averaged Hough transform to focus on the best illuminated edge using the shadow position. Another option could be to improve the contouring by adapting the segmentation threshold to the various areas of the target.

It can be seen indeed that the vertical ratio diminishes especially the number of bigger errors. The higher the error, the most likely the ratio will be over the set threshold as described in Section 5.4.2.

These observations are confirmed by Table 5.17 with the best scores attained for the direct Hough transform with a vertical ratio. The Hough transform achieves better results than what was already reported (7.52° against 11.7°) for its first application on the MSTAR SOC data which shows the importance of proper segmentation for this method [133]. The Hough transform could perform better if the best pose out of the several potential poses was considered as in [56]. It achieves similar results to various geometrical pose estimation methods (MAE of 6.70° against 5.91°) [133]. It performs however worse than methods based on wavelets or entropy. These methods require training beforehand [64, 101].

Method to estimate the target orientation	Mean	σ	MAE	RMSE
Direct Hough transform on the GMM segmented target	-0.87	16.81	7.52	16.88
Direct Hough transform on the threshold segmented target	4.78	16.62	7.80	17.30
Direct Hough transform on the threshold segmented target with vertical ratio	3.88	**13.81**	**6.76**	14.34
Averaged Hough transform on the threshold segmented target	4.63	17.54	9.71	18.14
Averaged Hough transform on the threshold segmented target with vertical ratio	4.26	15.56	8.94	16.13

Table 5.17: Error statistics of the errors in the target orientation determination in the MSTAR SOC 10 database.

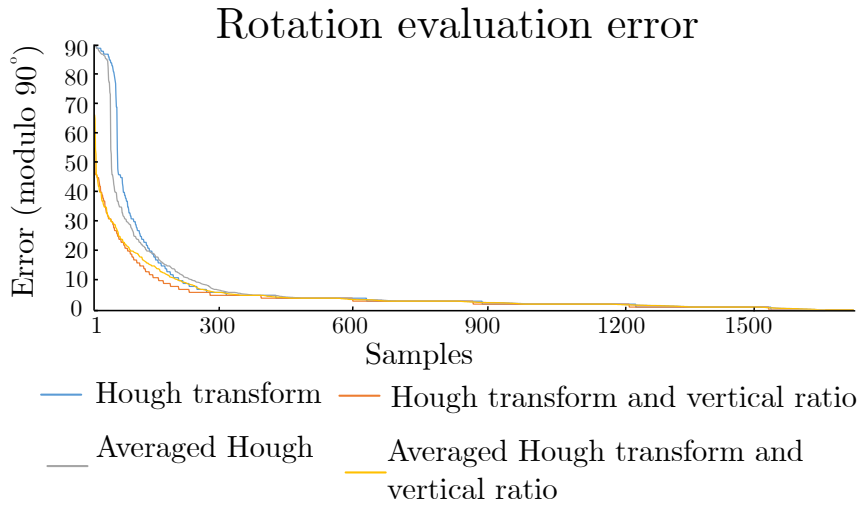
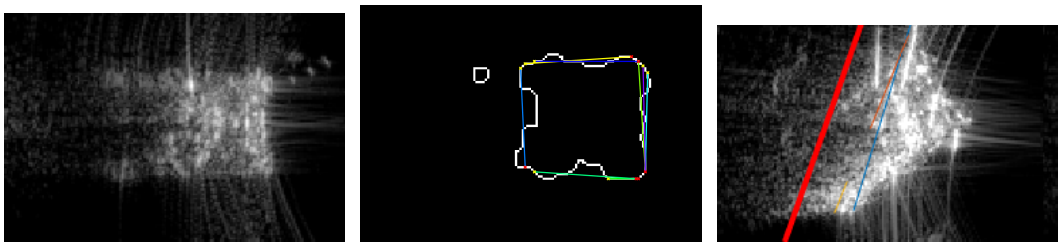


Fig. 5.14: Error distribution of the orientation estimation in the MGTD.

Pose estimation on the MGTD 90° errors occur often in the MGTD because of a segmentation error, such as in Fig.5.15, due to the illumination of the target being essentially localised on a small side of the target. Part of the target does not reach the segmentation threshold and the contour of the target resemble more a square than a rectangle. However, the vertical ratio corrects these mistakes, using the original images that take into account even the lower intensities of the target. The introduction of the vertical ratio improves more the orientation error on the MGTD than on the MSTAR data. This important error drop is reported in Fig.5.14.

The errors in the 0°-40° range are as in the MSTAR due do the difference of illumi-

nation inside the target. Some areas are not included in the segmentation because of a too low intensity. This new edge is picked up by the Hough transform as the line is longer than the real edge, more or less on the target diagonal as seen in Fig.5.15 (c). This worsen the first estimation of the target orientation.



(a) Original image leading to a potential 90° error.

(b) Corresponding segmented image and Hough transform lines.

(c) Error due to the illumination from a specific aspect angle.

Fig. 5.15: Causes of the main biggest errors.

The best performing algorithm in Table. 5.18 is the direct Hough transform with the vertical ratio. However, it should be noted that without the vertical ratio, the averaged Hough transform performs better than the direct Hough transform.

Method to estimate the target orientation	Mean	σ	MAE	RMSE
Direct Hough transform on the threshold segmented target	2.98	16.59	6.79	16.85
Direct Hough transform on the threshold segmented target with vertical ratio	2.31	**7.86**	**4.37**	8.14
Averaged Hough transform on the threshold segmented target	2.74	14.83	6.49	15.08
Averaged Hough transform on the threshold segmented target with vertical ratio	2.12	8.43	4.80	8.43

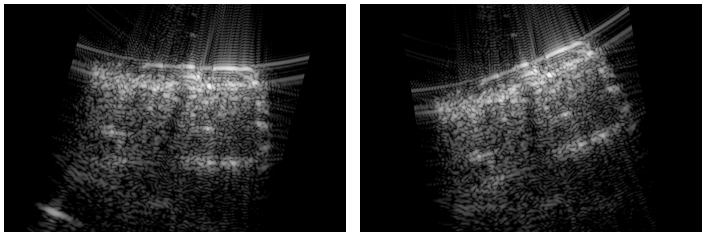
Table 5.18: Error statistics of the target orientation determination in the MGTD.

Target direction

The Hough transform gives an estimation of the orientation of the target modulo 180°. However, features are different for the front and the back of the target. Very few methods address the determination of the exact pose of the target on the full 360° [98, 99]. The pose-informed classification method requires prior knowledge on the 360° orientation as

the image will be distributed to a CNN trained on targets with similar orientation. In order to determine the direction of the target, a CNN similar to the one proposed in Section 5.3.1 is used. This CNN, given a rotated input image with a horizontal target, determines if the final target orientation should be θ or $\theta + 180$ with θ the target orientation given by the Hough transform.

CNN The CNN used for this analysis is the same AlexNet as presented in Section 5.3.1. The only difference is that the last fully connected layer provides only two classes, i.e. Front or Back of the target facing the right side of the image. The two classes are labelled 0° or 180°. The training parameters are the same as in Section 5.3.1 and, similarly, only the learning rate is adapted to each database.



(a) Image with a 180° orientation rotated by the given groundtruth angle.

(b) Image with an assumed 180° direction after image rotation with a Hough transform estimation off by 20°.

Fig. 5.16: Example of images with a 180° direction label.

Training For training, rotated images with a horizontal target are supplied to the network from the appropriate training dataset. In order to maximise the training data, two types of images are supplied:

- Images rotated with the groundtruth orientations: Each image is rotated using the groundtruth angle to produce two new images with two different target orientations consisting of a 0° or 180° direction. An example of a groundtruth 180° direction is given in Fig.5.16 (a).

- Images rotated with the orientation found with Hough transform: The image is

rotated according to the found orientation with the Hough transform. The 0° or 180° labels are assigned according to the closest label orientation to the found Hough transform orientation, as per Eq. (5.9). An example of a 180° direction label from the Hough transform estimation with an error is shown in Fig.5.16 (b).

Rotation data augmentation is also included with a random rotation between -15° and 15° of the training data in order to make the CNN robust against potential orientation estimation errors made by the Hough transform.

$$\text{Direction label} \in \begin{cases} \{0^\circ\} & \text{if } \mid \theta - \hat{\theta} \mid < 90 \text{ or } \mid\mid \theta - \hat{\theta} \mid -360 \mid < 90 \\ \{180^\circ\} & \text{else.} \end{cases} \tag{5.9}$$

where

θ	is the groundruth full orientation of the target,
$\hat{\theta}$	is the estimated orientation of the target found from the Hough transform.

Results In order to have a realistic evaluation, the CNN was evaluated on the real testing data consisting of images rotated using the Hough transform estimation and thus with potential orientation errors.

When the difference between the final orientation and the groundtruth angle was less than 90°, the CNN determined the correct target direction (Front or Back). Indeed, if the error is more than 90°, the error could have been minimised by adding 180° to the target orientation. Results are represented in Table 5.19 for the MSTAR databases and in Table 5.21 for the MGTD. In addition, metrics to evaluate the 360° orientation determination are given in Table 5.20 for the MSTAR databases and in Table 5.22 for the MGTD.

Dataset	0°	180°	Direction classification score
Training SOC 10	98.24%	98.69%	98.46%
Testing SOC 10	96.21%	95.89%	96.04%
Training EOC 1	100%	100%	100%
Testing EOC 1	98.69%	98.61%	98.65%
Training EOC 2	100%	99.89%	99.94%
Testing EOC 2	92.54%	89.94%	91.24%
Training EOC 3	100%	100%	100%
Testing EOC 3	88.22%	89.59%	88.91%

Table 5.19: Error statistics of the target orientation determination of targets in the MSTAR database.

Database	Mean	σ	MAE	RMSE
MSTAR SOC 10	-0.51	33.56	11.84	33.55
MSTAR EOC 1	-0.08	24.71	11.23	24.70
MSTAR EOC 2	-4.13	52.44	19.58	52.603
MSTAR EOC 3	-5.84	60.97	24.51	61.25

Table 5.20: Error statistics of the full target orientation determination in the MSTAR database.

In both databases, the RMSE increased compared to the RMSE in the previous section as the highest errors can now attain 180° instead of only 90°. Comparably to the AlexNet classification scores, the direction of the target was harder to determine in the MGTD images. The full 360° orientation has rarely been investigated on the MSTAR data and, when it has been, it was often not on the standard datasets defined in Chapter 2. A statistical

method has been tested on the MSTAR EOC 2 and EOC 3 [99]. Results are given with the Hilbert-Schmidt distance given in Eq. (5.10) with an equivalent error in degree. However, as the cosine is not linear, the MAE cannot be obtained by directly inverting the cosine as it is done in [99], thus the squared Hilbert-Schmidt distance obtained with our method is calculated in order to be able to compare the results with this statistical method. A value of 0 corresponds to a perfect estimation of the orientation, 8 corresponds to a 180° error. An average distance of 0.8 was achieved on the MSTAR EOC 2 and 1.0 on the MSTAR EOC 3 dataset, while the statistical method reported a distance of 1.7 on the MSTAR EOC 2 and 2.0 on the MSTAR EOC 3 dataset. The statistical method also assumed knowledge of the target type to achieve those results. The proposed method is thus more precise and requires less prior information.

$$d_{HS}^2 = 4 - 4\cos(\theta - \hat{\theta}) \tag{5.10}$$

where

θ	is the groundruth orientation of the target,
$\hat{\theta}$	is the estimated orientation of the target found from the association of a direct Hough transform and the direction CNN.

Dataset	0°	180°	Direction classification score
Training	99.56%	99.91%	99.76%
Validation	99.77%	100%	99.89%
Testing	93.28%	93.75%	93.51%

Table 5.21: Error statistics of the target orientation determination of targets in the MGTD.

Mean	σ	MAE	RMSE
-1.46	45.29	14.34	45.28

Table 5.22: Error statistics of the full target orientation determination in the MGTD.

5.4.3 Training of the pose-informed CNNs

Once the orientation of the target is estimated, the image is analysed by the appropriate CNN from the pose-informed architecture. An example of the separation of the pose-informed CNNs, each focusing on a specific orientation range, is given in Fig.5.17.

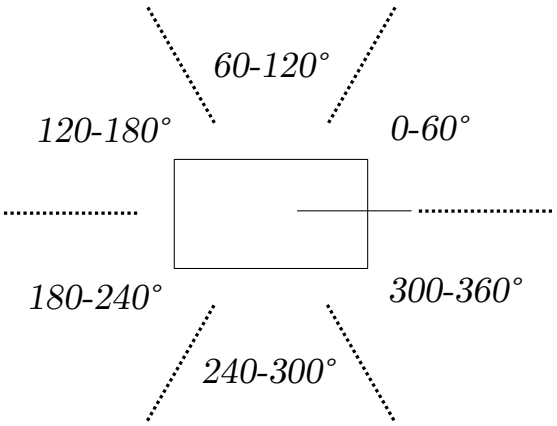


Fig. 5.17: Orientation ranges for an architecture with 6 pose-informed CNNs.

Two image pre-processing options are possible once the target orientation is determined. The aspect angle of the target has an important impact on the target appearance [19, 96]. It can be thus interesting to minimise its influence by rotating targets so that they always face the same direction. The location of the most prominent features should then remain constant. The change of illumination with the rotation will still affect the features but their locations will ease physical interpretation and be related to a specific part of the target. However, the rotation of images could also hinder the classification as the classification method will have to handle the hard task of distinguishing between the change of backscattering related to the change of target orientation and target class. The orientation can only be seen from the change of illumination in the rotated image as the target retains the same location and direction. Both the possibilities of supplying the original image and the image rotated so that the target appears at a 0° orientation to the pose-informed CNN are investigated in this chapter.

Instead of training each pose-informed CNN directly, a parent CNN is trained on the full SAR training set with all possible target orientations. The evolution of the training

loss reflecting the modality transfer learning is shown in Fig.5.18 (a). This parent CNN is re-trained later on a specific orientation range to become the pose-informed CNN specialised on this orientation range. The operation is repeated to obtain the *n* pose-informed CNNs composing the full architecture. Transfer learning is achieved with the parent CNN by supplying SAR images with targets in a specific orientation range. A second transfer learning step is the orientation-speciality transfer learning shown in Fig.5.18 (b). This transfer learning by stage is a training strategy comprising of a modality transfer learning followed by an environmental specific transfer learning, in this case orientation-specific, to make the pose-informed CNNs fully aware of the feature changes in a specific orientation. Transfer learning by stage optimises the number of images the pose-aware CNNs have at their disposal for training because the more common SAR features are learned by the parent CNN, while the pose-informed CNNs focus on finer and more specific features during the orientation-speciality transfer learning.

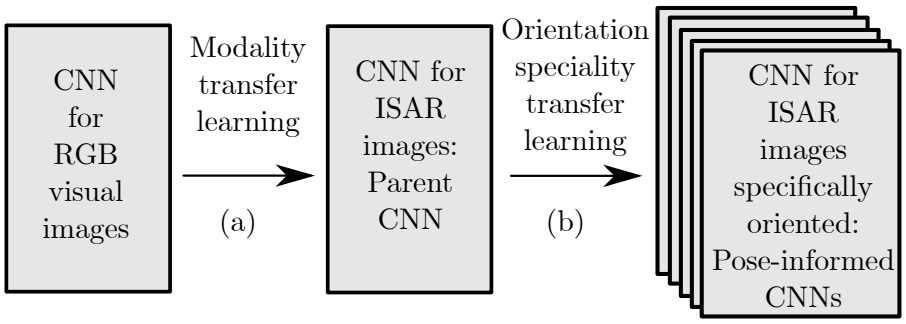


Fig. 5.18: Training by stage of the pose-informed CNN. A modality transfer learning followed by an orientation transfer learning.

The same AlexNet presented in Section 5.3.1 was used. A random search was used to find the hyper parameters for the modality and orientation-speciality transfer learning. The data augmentations described in Section 5.3.2 are included to increase the number of images for training.

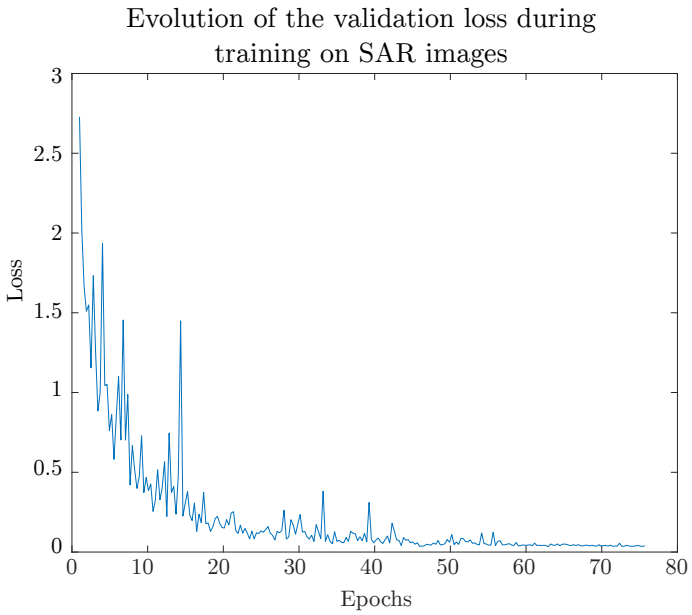
With traditional transfer learning, the pose-aware CNN would be trained directly on the ISAR images in a specific orientation range. With transfer learning by stage, the pose-informed CNN is trained on the full ISAR training set before training on the ISAR images in a specific orientation range. The evolution of the loss during transfer learning

by stage with the first traditional transfer learning (from the visual to SAR domain) and the orientation-speciality transfer learning is shown in Figs. 5.19 (a) and 5.19 (b). The loss corresponding to the orientation-speciality training has a lower starting value than for the modality transfer learning as the network to be trained already underwent a number of training epochs in the SAR domain. At the end of the second training, the validation loss is lower than at the end of the modality transfer learning (Less than 0.01 against 0.05).

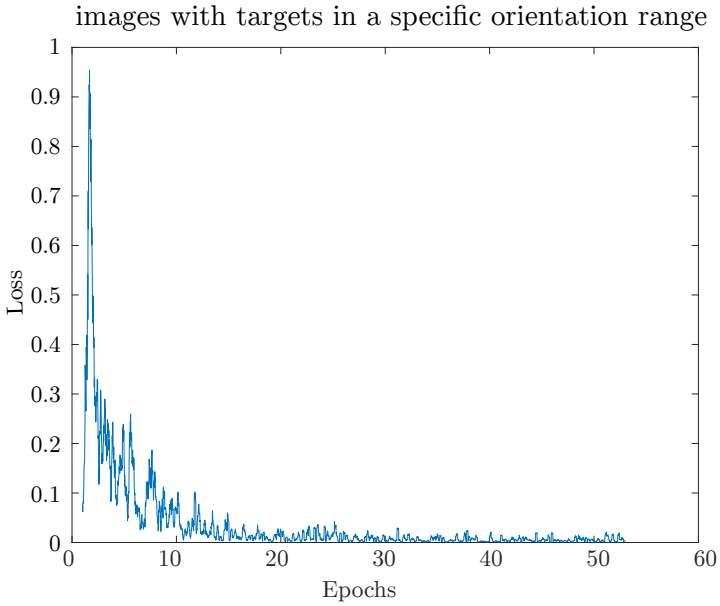
The first stage of transfer learning makes the pose-informed CNNs learn standard SAR features. The second stage facilitates their specialisation in a particular orientation range. This method optimises the use of all training samples and promotes the learning of a specialised CNN. It becomes possible to learn features that are present in the overall training set but are sparse in the specialisation areas.

5.4.4 Computation of the result range

It is not possible to have a proper separation between the training and the validation set in the MSTAR database, because for example, the MSTAR SOC 10 and EOCs provide only one sequence of images for each target in the training set. Thus even if part of the training set is dedicated to validation only, a high validation score does not prevent overfitting as the images are taken with the same target, in the same configuration, with the same depression angle and at the same time period. The MGTD instead provides four series of images, for each target with different configurations at different time period, of which one can be dedicated to validation purposes. However, the same procedure was applied to all datasets to guarantee a uniform testing method and 10% of the training set was randomly allocated as validation set. As a result, CNNs with the same validation score could have different testing results. This is mainly due to the initialisation of the last layers. This applies even more for datasets with a small number of images in the validation set, such as the EOCs datasets. In an attempt to report the results fairly, a range of the scores achieved on the testing set is given rather than a single percentage. The assumed best performing CNNs selected are those with the highest classification score on the validation set. The



(a) Evolution of the validation loss during the modality transfer learning of the network.



(b) Evolution of the validation loss during the orientation-speciality transfer learning of the network.

Fig. 5.19: Evolution of the validation loss during transfer learning.

lowest and highest scores achieved on the testing set are then reported. The range result of the pose informed architecture has to take into account the different networks involved. For each orientation range, the best and worst performing CNN with the highest validation score are saved. The combination of the worst CNNs in each orientation range into one

pose-informed model gives the minimum of the classification rate achievable. The same process is adopted for the best CNNs in each orientation range.

5.4.5 Classification results for the pose informed CNN architecture

	Classification rate	Standard CNN	Number of pose-informed CNN						
			2	3	4	5	6	7	8
Target free rotating	Max.	96.87	98.80	**99.13**	98.85	**98.97**	**98.97**	98.60	**99.01**
	Min.		98.06	97.53	97.16	97.11	96.25	96.70	96.45
Fixed target orientation	Max.	97.56	98.10	98.43	98.19	**98.47**	98.30	**98.39**	**98.47**
	Min.		97.60	97.32	96.95	97.20	96.74	96.33	96.66

Table 5.23: Range results (%) of the pose-informed classification method compared to a standard CNN on the MSTAR SOC 10. The 3 best scores are highlighted in each category.

	Classification rate	Standard CNN	Number of pose-informed CNN						
			2	3	4	5	6	7	8
Target free rotating	Max.	77.32	78.54	81.84	81.92	**83.40**	82.88	**85.58**	**83.58**
	Min.	66.81	69.85	71.85	70.72	71.24	71.07	70.37	69.68
Fixed target orientation	Max.	85.06	85.31	86.19	**88.10**	**88.27**	**88.97**	87.92	86.79
	Min.	70.29	73.76	73.06	73.32	72.98	73.85	73.24	71.33

Table 5.24: Range results (%) of the pose-informed classification method compared to a standard CNN on the MSTAR EOC 1. The 3 best scores are highlighted in each category.

	Classification rate	Standard CNN	Number of pose-informed CNN						
			2	3	4	5	6	7	8
Target free rotating	Max.	92.32	93.30	93.50	93.64	**94.09**	94.00	**93.89**	**94.03**
	Min.	87.53	88.68	87.58	87.28	85.34	86.97	85.68	83.61
Fixed target orientation	Max.	91.71	93.22	92.55	**93.33**	92.71	92.60	**93.39**	**93.22**
	Min.	87.76	88.56	86.52	84.76	84.45	83.80	83.25	82.90

Table 5.25: Range results (%) of the pose-informed classification method compared to a standard CNN on the MSTAR EOC 2. The 3 best scores are highlighted in each category.

Each table relates the scores achieved for both the standard CNN and the proposed pose-informed architecture. Both methods have been tested on the original images and

	Classi-fication rate	Standard CNN	Number of pose-informed CNN						
			2	3	4	5	6	7	8
Target free rotating	Max.	**95.24**	89.37	89.78	91.07	92.51	**93.54**	**93.10**	92.91
	Min.	85.46	79.37	78.67	77.60	72.99	72.21	74.95	71.62
Fixed target orientation	Max.	88.34	90.19	90.92	90.81	**91.51**	90.66	**91.00**	**91.33**
	Min.	76.75	75.68	79.08	78.52	74.02	73.54	75.09	71.73

Table 5.26: Range results (%) of the pose-informed classification method compared to a standard CNN on the MSTAR EOC 3. The 3 best scores are highlighted in each category.

	Classi-fication rate	Standard CNN	Number of pose-informed CNN						
			2	3	4	5	6	7	8
Target free rotating	Max.	91.20	91.43	91.89	**92.12**	**94.29**	**92.35**	89.95	91.43
	Min.	82.19	89.38	88.35	84.02	86.07	84.36	82.42	85.84
Fixed target orientation	Max.	88.24	90.30	90.64	**92.58**	**93.04**	**92.69**	91.67	90.98
	Min.	83.79	87.75	85.05	85.39	86.99	86.07	84.93	82.64

Table 5.27: Range results (%) of the pose-informed classification method compared to a standard CNN on the MGTD. The 3 best scores are highlighted in each category.

on rotated images with targets in a fixed orientation. Results are reported for the MSTAR SOC 10 (Table 5.23), MSTAR EOC 1 (Table 5.24), MSTAR EOC 2 (Table 5.25), MSTAR EOC 3 (Table 5.26) and the MGTD (Table 5.27). Only Table 5.23 has no range for the standard CNN as the larger number of images in the validation set enabled a finer distinction between scores and only one CNN achieved the highest validation score.

The best rates for both methods on all datasets are: 97.56% for the standard CNN against 99.13% for the pose-informed on the MSTAR SOC 10, 85.06% against 88.97% on the MSTAR EOC 1, 92.32% against 94.09% on the MSTAR EOC 2, 95.24% against 93.54% on the MSTAR EOC 3, 91.20% against 94.29% on the MGTD. Overall, the pose-informed architecture outperforms the standard method, even though the amount of training data for the orientation-speciality transfer learning was very limited. Concerning the MSTAR EOC 3, the pose-informed architecture performs less than the standard CNN with a drop of 6% in the worst case scenario with 2 pose-informed CNNs, and 3% for 5 pose-informed CNNs.

The scores of the 5 CNNs pose-informed method are also compared with a 2 CNNs pose-informed method on the free rotating target images to evaluate the importance of a higher number of CNNs and thus orientation ranges. In the MSTAR SOC 10, the pose-informed method with 5 CNNs performs equivalently to the pose-informed method with only 2 CNNs with an improvement of only 0.17%. Similarly, it performs 4.86% better in the MSTAR EOC 1, 0.79% better in the MSTAR EOC 2, 3.14% better in the MSTAR EOC 3, 3.09% better in the MGTD. The 5 CNNs pose-informed achieves thus better results than the 2 CNNs pose-informed with an average improvement of 2.41% which is not negligeable for scores over 90%. The 2 CNNs pose-informed has less possibility to adapt to a specific aspect angle as the images provided for training are less orientation specific. It would seem that the proposed method has indeed been able to learn extra information about specific orientations even without additional data by applying transfer learning in two stages. The pose-informed method would probably deeply benefit from additional data because of the low number of images in the second training set resulting from the aspect angle partition of the training data.

It seems that the pose-informed architecture with 5 CNNs performed the best overall. It can be noted that the maximum of the pose-informed architecture performed better than the standard CNN with the exception of the MSTAR EOC 3 dataset while the minimum achieved better or similar in the MGTD, MSTAR SOC 10, MSTAR EOC 1 with lower scores in the MSTAR EOC 2 and MSTAR EOC 3. That means that even if the worst performing CNNs from the pose-informed CNNs set are selected out of the CNNs with the best validation score, the method still often achieves higher scores than the standard CNN method. The poor results on the MSTAR EOC 3 could be caused by the orientation determination results which achieved the worst results on this database in Section 5.4.2. If the target in one orientation is analysed by the CNN from another orientation range, the results could be worse than those of a standard CNN trained on the whole SAR training set.

The standard CNN achieves better results on images with a free rotating target than

on images with a fixed target orientation except for the MSTAR SOC 10 and EOC 1. Both those datasets are the only datasets with the same target variants and configurations between training and testing. It is thus possible that the back scattering process is more similar between training and testing. In that case, the algorithms working on images with a target in a fixed orientation could overfit better the data, as CNNs are not rotation invariant, and improve the classification rate on the testing set.

5.5 Conclusion

In this chapter, a new realistic noise based data augmentation is proposed to make up for the small amount of data available to train deep networks for SAR ATR. It is directly applied to the range profile before the image computation. Combined with classical translation data augmentation, the score of the AlexNet on the MGTD improves from 77% to 91%.

A pose-informed architecture is also proposed, taking into account the target orientation in the classification process. The target orientation is determined first, followed by the target classification using a CNN specialised in a certain target orientation range.

The orientation determination is handled over 360° with a proposed association between a Hough transform, a study of the image intensity over specific zones and a CNN recognising the target direction. This orientation determination performs better and does not require prior knowledge on the target type, on the contrary to former statistically based method.

The proposed pose-informed architecture performs better than the standard CNN on its own, except on the MSTAR EOC 3 which has the poorest precision for orientation determination. It achieves respectively 99.01%, 85.58%, 94.09%, 93.54% and 94.29% on the MSTAR SOC 10, EOC 1, EOC 2, EOC 3 and MGTD, so an improvement over the standard CNN of respectively 2.14%, 8.26%, 1.77%, -1.7% and 3.09% for an average improvement of 2,71% overall. This architecture is a trade-off between extra precision

and storage needed for the various CNNs composing the model. Shallower networks have been proved to also work for SAR ATR and could replace the AlexNet in the pose-informed architecture [73, 134].

Chapter 6

Deep learning network explainability through feature analysis

Contents

6.1 Summary

This chapter presents analyses of a multitude of factors that have an influence in the decision process of a deep learning ATR method. Three ways to analyse the data are presented [11].

Firstly, an analysis on the individual contribution to the classification of the target, shadow and background area is carried out. The respective influence of the target, shadow and background area appears to be target dependant. The CNN bases its classification,

depending on the target class, on one or a combination of those areas. Even so, over the 10 targets in the MSTAR, the shadow area appears to influence deeply the classification (a drop in classification rate higher than 50% when the corresponding area is hidden) for only 3 targets, compared to 9 targets for the background area and 9 targets for the target area over a total of 10 targets. These scores are confirmed when looking at how critical the features are in each of these zones. However, if the shadow appears to be the less used area of the image, the removal of the shadow area still causes a drop of at least 20% in the classification score for 7 target types. This would require some further study but indicate that target-only segmentation before attempting classification could be harmful as significant information could be lost.

Secondly, an analysis of the most influential features, defined as the features without which the classification process is deeply weakened, is undertaken by looking at their location and the intensity distribution of the areas in which such features are present. The location of such features are determined using occlusion maps and the proposed classification map, which made it possible to see the evolution of the classification rate when specific areas of the image are hidden as the target is always located and oriented in the same way. It appears that for most of the targets that have a higher part such as a turret or elevated cabin, it is one of their critical feature. For other targets, areas with critical features can be in the front or sides of the target. Location of the critical features is diverse and target specific. The same work is carried out for the orientation of the target. The areas that reflected the most the signal are the closest to the radar as they are facing the radar, on the contrary to the side areas or even the back area that is not directly illuminated. These areas are the most critical areas to determine the target orientation as seen in the classification maps. The central-back area of the target is also often an important area regardless of the orientation. It could be due to the fact that it often contains an elevated turret or cabin and this area is often less occluded. Another reason if that the geometry of the turret makes it likely to reflect the illumination towards the receiver. It also appears that the CNN trained with data augmentation performs better

and its classification process is more easily understood than a CNN trained without it. The average and standard deviation of the intensities distribution of the pixels containing critical features is compared to areas containing less important features. The distribution of critical features has a higher mean (difference of 17.17 bins) as well as a higher standard deviation (difference of 9.35 bins). The high variance could be related to surfaces with varied RCS such as the turret as it is composed of multiple plans not reflecting the signal evenly and thus carrying more specific features.

Lastly, a study on the feature specialisation along the CNN depth relative to a specific target type or orientation of target is presented to investigate the rate of differentiation between such groups. When applied to groups gathered by target type, which is exactly what the CNN has been trained to differentiate, it can be seen that the differentiation between target specific feature increases at a quite steady rate with CNN depth and feature complexity. This increase was expected. The same study is carried out on groups gathered by orientation range. Even without having a loss specific to this environmental variable, the CNN still learns to differentiate between the various orientations of the target. That encourages the possibility of training networks using transfer learning in different contexts, such as from one task to another rather than only from one database to another.

6.2 Introduction

Previous research has shown that deep learning often outperforms classical feature methods on several modalities such as in the visual domain on ImageNet and in the SAR domain on the MSTAR database [12, 110]. As shown in this thesis, classical feature-based models for SAR ATR are now giving a way to deep learning based methods. Unlike the feature-based models, for which features were man-made, features used by neural networks are created using artificial intelligence concepts. Deep learning features are quite complex, as they are the result of stacked convolutions and activations and this makes it very difficult to understand which information triggers a CNN decision. Unlike classical

features such as SIFT or SURF, deep learning features, in general, cannot be easily improved or even understood by humans, especially those generated by the deepest layers. Knowing the origin of CNN decisions and explaining them is a very important problem, that, if solved, would make it possible to choose the most rational network among several solutions. Artificial intelligence algorithms would be more trustworthy and also could take advantage of the impressive human visual understanding [135]. The understanding of the internal work of deep learning solutions is a recent research area and is essential to further improve deep learning methods, to validate them over former techniques, and to trust them enough to adopt them in real scenarios. Several approaches have been proposed in the visual domain with deconvolutional networks enabling the visualisation of high level features [136, 137], the analysis of the role of features for each class respectively [138], or the influence of choice of training data over specific misclassifications [139]. Deep learning network understanding in the SAR domain currently remains limited to the visualisation of the deep learning low level features [80, 81, 91]. As SAR images are totally different from visual images, their respective features are likely to be different and contribute differently to the network decisions. Indeed, SAR images have additional phase information, no colours and have a lower resolution than visual images.

This chapter presents three streams of analysis of one deep learning algorithm after training. Firstly, the individual contribution to classification performance of the different part of the image, respectively the target, the shadow and the background is assessed. Target classification relying only on the target shadow has been investigated and shows that the addition of features from the shadow can improve classification rates [140, 141]. Current SAR ATR algorithms are fed with full images as well as segmented target images. Thus, it would be interesting to see the extent of the information lost through the segmentation. This is achieved by classifying images with specific segmentations and by studying the presence of critical features in each image part as presented in Section 6.4.

Secondly, the location and distribution of intensities composing the areas containing features critical to achieve correct classification are investigated and compared to that of

unessential features in Section 6.5. To do so, occlusion maps are created to highlight the location of SAR features essential for correct classification by the deep learning method tested [136]. Another option for more precise results is the use of guided backpropagation to visualise the patterns, or common elements such as shapes, lines and colours characterising an object, learned by the CNN [142, 143]. However, visualising patterns is useful only if it is possible to interpret them according to previous knowledge of the target appearance that leads to reasonable CNN classification. In the optical domain, with a specific training set for example, it could be seen that the CNN classified between wolves and huskies according to the presence of respectively snow and grass [144]. Recognizing specific patterns is already challenging in the visual band even if humans use visual images everyday, and it is even more difficult for SAR images. Indeed, an untrained person may not be able to distinguish different targets in SAR images. The occlusion maps are also extended in this chapter to classification maps to analyse a group of images rather than a single image. Applying this method on a group of images with common environmental factors clarifies their role on the choice of the features learned by the CNN. The generating process of occlusion maps and classification maps is described in Section 6.3. Classification maps will be used in Section 6.6 and 6.7 to determine the location of features critical for classification for targets of a specific class or with a specific orientation. Results achieved are compared for a well-trained network and a less performing network, which did not benefit from data augmentation during training.

Lastly, Section 6.8 investigates the specificity of features, that is how much distinct are patterns that activate different features. The network tends to develop features to be sensitive to a specific target class. A feature specific to a target class would not be activated by patterns issued from other target classes. On the contrary, a non-specific feature could be activated with any target image analysed. In Sections 6.6 and 6.7, the specificity of features to the target class and to the target orientation is investigated. The features investigated for specificity are those that are most activated when the network is presented with images of a target with a certain target type or orientation. The objective is to evaluate the

power of discrimination of the network against target classes and target orientations along the network depth as the computed features grow more complex. The specificity of the features is shown for both a high-performing network and one that performs less because it did not benefit from a data augmented training.

6.3 Computation of occlusion maps and classification maps

An occlusion map shows the contribution of each location of an image to the classification performance of a deep network algorithm, and as such, it is specific to a trained CNN and the image to be classified. An occlusion map is obtained by hiding specific parts of the image and observing the impact on the correct class score. By collecting score variations by hiding alternatively various image parts, the location of the features that are critical for the CNN can be highlighted [136]. Gradient-weighted Class Activation Mapping (Grad-CAM) images have the same goal but are more precise [135]. Several methods have been investigated in the visual domain to locate the focus of attention of CNNs [144, 145]. They either block specifically chosen forward propagations or use the backpropagation of the gradient from the correct class. Grad-CAM images have already been computed on SAR images [25]. The gradient is however specific to one image and has not been extended to a full group of images. Grad-CAM method would be complex to adapt to an image group. As the analysis of features for a group of images is one objective of this section, Grad-CAM images will not be investigated further. Instead, in this thesis, occlusion maps are further extended to be applied to several images and are referred to as classification maps thereafter. Classification maps computed with a specifically chosen group of images highlight features related to environmental variables such as the target orientation or the target class. Using a group of images rather than a single image attenuate the influence of different other factors such as speckle, change of background, changes of the acquisition geometry on a single image. Different analyses are proposed and carried

out using the occlusion and classification maps in Sections 6.4–6.8 by studying the critical features of the images. The critical zones are defined as the areas in the occlusion maps or classification maps with a low intensity. This low intensity means that the absence of features from these areas impacted significantly the classification.

6.3.1 Methods to produce occlusion and classification maps

Dataset

The analysis proposed in this chapter is conducted on the MSTAR SOC 10 database and the MGTD presented in Chapter 2. Two different pre-processing techniques can be applied according to the parameter that is investigated with the occlusion maps. Either the target rotates, as it is the case of the standard databases presented in Chapter 2, or the rotation is compensated with the groundtruth orientation angle so that the target appears not to rotate. When the latter is used, the targets are in the same position in several images. This enables the creation of classification maps in which each part can be attributed to a specific target location.

Plain dataset The segmentation groundtruth is only available for the MSTAR, and is used to determine the centre of the target. The targets in the MSTAR dataset are centred. The targets in the MGTD are not centred because no segmentation groundtruth is available. Due to acquisition conditions, the target translation is in any case minimal. In the

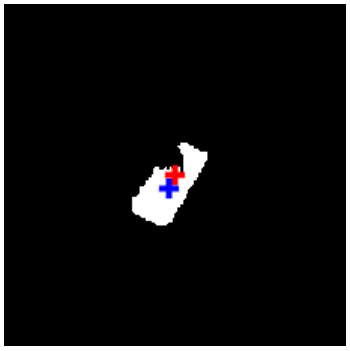


Fig. 6.1: MSTAR target centre of mass in blue and image centre in red.

plain dataset, the images used to compute occlusion maps are firstly centred if possible.

The centring makes it possible to analyse classification maps as the location of the target remains consistent. Centring is achieved by using the SARbake segmentation of the target [44]. The location of the centre of mass of the target area is calculated on the segmented binary image with Eq. (6.1).

$$\{C_x, C_y\} = \left\{ \frac{\sum_{i=1}^{n}\sum_{j=1}^{m} i.x(i,j)}{\sum_{i=1}^{n}\sum_{j=1}^{m} x(i,j)}, \frac{\sum_{i=1}^{n}\sum_{j=1}^{m} j.x(i,j)}{\sum_{i=1}^{n}\sum_{j=1}^{m} x(i,j)} \right\} \tag{6.1}$$

where $\{C_x, C_y\}$ are the coordinates of the centre of mass, $x(i,j)$ is the image intensity at point (i,j) of the binary image (0 or 1).

The image is then translated so that the target centre of mass is at the same location as the image centre as shown in Fig. 6.1.

Rotated dataset The MSTAR targets are first centred as for the plain dataset in Section 6.3.1. The image is then rotated using the real orientation of the target given by the groundtruth. An estimation of the orientation is not developed here as the objective is not to evaluate the classification algorithm but to precisely superimpose the targets to be able to measure the influence of the various environmental factor and target features.

CNN used to produce the maps

The evaluated CNNs are the same as the parent CNN presented in Chapter 5. They are an updated version of the AlexNet network with a new last fully connected layer to fit the number of classes considered in the studied database. A simpler architecture is chosen over the pose-informed architecture for this analysis for clarity and to limit the interactions of the various elements composing the pose-informed method. For example, after occluding parts of the image as it will be done in this analysis, the determined target orientation could change. It would be both fair, in this case, to take into account each of the two pose-informed CNNs corresponding either to the former or new target orientation. However, these networks could react differently. Another solution would consist of eval-

uating each network separately but would undermine the advantages of the pose-informed complete solution. These networks are trained using transfer learning from the visual domain to the appropriate SAR database as specified in Section 5.3.2. The images used for training are rotated and centred images of the target, thus maintaining the target in a fixed position. In order to evaluate what makes good features, the results of two CNNs are going to be compared for each database: The first is a CNN trained with data augmentation (translation for the MSTAR SOC 10, translation and Weibull noise based for the MGTD), performing at respectively 98.26% on the MSTAR SOC 10 and 91.32% on the MGTD. The second is a CNN trained without any data augmentation, which is less robust and only achieves 95.12% on the MSTAR SOC 10 and 67.00% on the MGTD. The selected networks must have a score on the validation set within 3% of their score on the training set. The results are slightly higher than what is achieved in Chapter 5, as the groundtruth target orientation is used rather than an estimated target orientation to rotate the images, and also because the CNNs are selected for their good testing scores rather than their validity scores. In the previous chapter, the selection could only be done according to the CNN score on the validation set to be sure not to compromise testing. In this chapter, it is not the robustness of the network itself that is evaluated but rather the decisive features that makes the network reach the highest classification scores on these datasets. that

Computation of the occlusion map

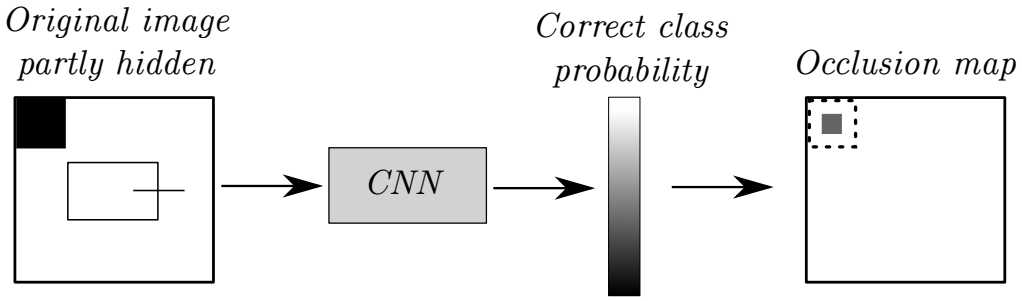


Fig. 6.2: Creation of the occlusion map.

Occlusion maps are already used in the visual field [135, 136]. The objective of such

maps is to study the location of the features that contribute the most to correct classification. Our implementation of the occlusion map starts by hiding the top left corner of the original image by applying a 11×11 black square mask. This partly masked image is fed to the CNN. The resulting score of the correct class is the new intensity of the 5×5 pixels in the centre of the 11×11 mask. The new intensity is thus between 0 (black or dark blue with the Matlab jet scale in Fig.6.3) if the score of the correct class drop to 0 and 1 (white or red with the Matlab jet scale in Fig.6.3) if the CNN is so confident that the score of the correct class is 1. The same process is repeated by masking another part of the image: the 11×11 mask is moved by 5 pixels to the right and this new partially masked image is fed to the evaluated algorithm to obtain the new score of the correct class and thus the intensity of the occlusion map in this specific location. The mask is shifted by 5 pixels step horizontally and vertically until all the intensities in the occlusion map are determined as shown in Fig. 6.2. The final map areas with a high intensity are the areas for which the score of the correct class was not severely degraded when this area and its surroundings were hidden. That means that no feature at this location was critical to achieve correct classification. On the contrary, if the intensity in the occlusion map is low in an area, this means that this area contains some features critical for the algorithm to correctly classify the target. Occlusion maps can be obtained on both the plain and the rotated images using the appropriately trained CNN. Fig. 6.3 is an example of an occlusion map obtained on one image from the rotated MSTAR SOC 10 dataset. The target is facing the right side of the image. The middle larger blue area corresponds to the target turret. The blue area on the right side of the target corresponds to the very front panel of the target that is more tilted than the rest of the front target area.

Computation of the classification map

The classification map is an extension of the occlusion map applied to a group of images containing a target with a fixed location. Having a group of images rather than a single image highlights the role of environmental variables shared by a group of images. Many

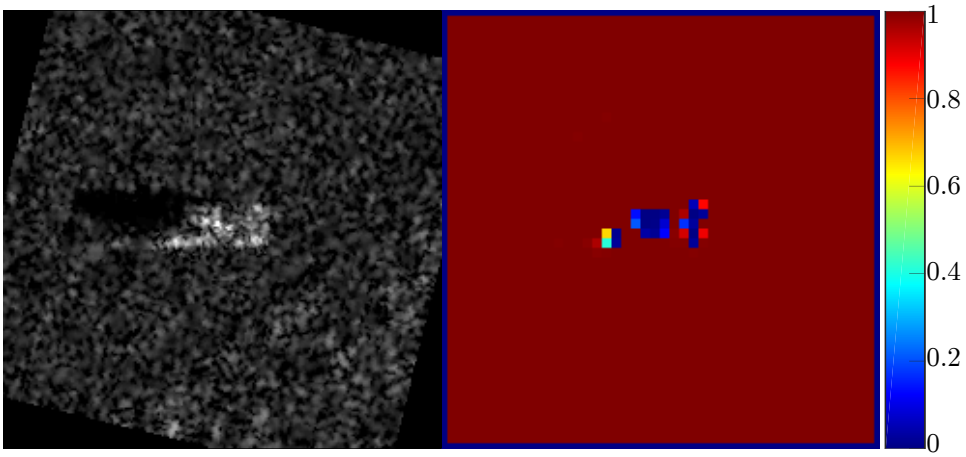


Fig. 6.3: Occlusion map of a rotated image of a 2S1.

variables can be presumed to have an influence on the activation of specific deep learning features such as, for example, the target class, target orientation, depression angle. The influence of these variables in the image over the location of the most activated deep learning features is studied with a classification map obtained with images sharing the same value for this variable. These groups of images in this section will be images with the same target class or with similar target orientation.

After the optional centring and translation are applied to the images, as explained in Section 6.3.1, all targets are in the same location in each test image. Images with the same target or orientation are then grouped together to evaluate respectively the influence of the target class or orientation on the location of the critical features learnt by the CNN. A 11×11 black square mask is applied to the top left part of all the images belonging to the studied group. The percentage of correctly classified images is used as the new intensity of the 5×5 pixels located in the centre of the black square in the classification map. The black square is shifted on all the images by 5 pixels vertically and horizontally until the classification map is fully completed as shown in Fig. 6.4.

The results obtained with a well-trained CNN and a CNN trained without data augmentation are compared to highlight better the location of features leading to good classification rates.

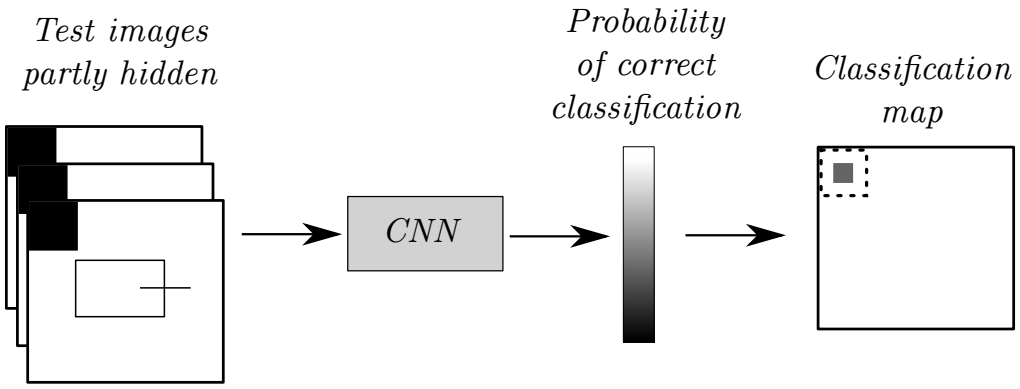


Fig. 6.4: Creation of the classification map.

6.4 Role of the target, shadow and clutter in the classification

The shadow of a target in SAR images contains information, for example, the target shape and height can be determined as the geometrical configuration of acquisition is known. This additional information could improve target detection and classification performance [140, 146]. Previous work in the literature investigated sharpening the shadow and the target parts of the image to improve the quality of information extracted from SAR images [147]. The goal of this section is to evaluate the amount of information present in the pixels that contain the target, its shadow and the clutter exploited by the neural network. The clutter area is also investigated for target classification as in the MSTAR images, the clutter area includes areas that could give information about the target through multipath. Additionally, because of the background correlation, the clutter area has an influence on the classification as the background area is sometimes the same in the training and testing set of a target. Firstly, two methods to evaluate both globally and in a detailed way the influence of each area are presented in Section 6.4.1. To evaluate the global contribution of each of these areas, Section 6.4.1 investigate the change in the classification scores following the occlusion of each and combination of these areas. The association of occlusion maps with segmentation information gives more details about the areas in which the critical features are present and this is investigated in Section 6.4.1. Then, the results

obtained from both of these methods are shown in Section 6.4.2.

6.4.1 Methods to evaluate the contribution of the target, shadow and clutter to the classification

Method to evaluate the global information loss in regards of partly segmented images classification

The classification scores achieved on segmented images, and thus with only partial information, are compared according to the segmented zone chosen. The impact of the loss of information from a specific zone of the image gives information about the importance of the features in that specific zone. The classification method used in this analysis is the same CNN presented in Section 6.3.1 trained on the plain MSTAR SOC 10 dataset presented in Section 6.3.1. The MSTAR SOC dataset was used for this study as it is the only dataset with a groundtruth segmentation [44].

The method is detailed to investigate the role of the target in the classification but the process is the same for the shadow and clutter. The SARbake segmentation detailed in Section 4.3.1 is used to get the location of the target, shadow and clutter area. All images in the testing dataset of the MSTAR SOC 10 dataset have the target area set to black, so that information from the target is removed. The CNN is then run on the incomplete images. The result is a classification score for which the CNN could not rely on features from the target area. This process is repeated to obtain the classification scores corresponding to all possible segmentation combinations of the three areas as shown in Fig. 6.5.

Criticity of features found in the occlusion maps for the target, shadow and clutter area for a detailed analysis

The previous method allows a global evaluation of the influence of image areas used by the CNN for classification. The method presented in this section is set on a finer level: The

Target		X	X	X		
Shadow	X		X		X	
Clutter	X	X				X

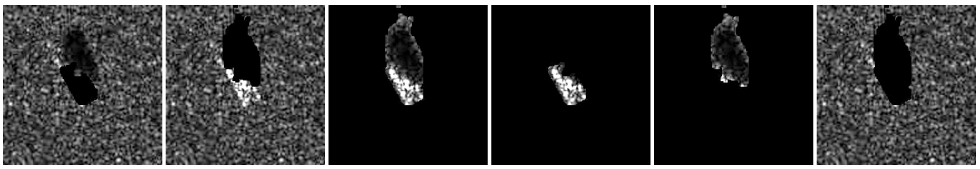


Fig. 6.5: Images with segmented area(s) hidden.

association of the occlusion maps and the SARbake segmentation is used to determine the presence of critical features in each area for each image.

To begin with, the occlusion maps are computed with the CNN from Section 6.3.1 on the MSTAR plain SOC 10 from Section 6.3.1. The lowest intensity of respectively the target, shadow and clutter area in each occlusion map is saved. The intensity in the occlusion map corresponds to the classification score of the correct class when a particular area is hidden. The lowest intensity is chosen to represent the criticality of the shadow, target or clutter area in a particular image for the classification. If a small part of the area is critical to the classification, the lowest intensity of this area in the occlusion map will represent the importance of the area. If the average was taken instead, the saved intensity would depend on the intensity of the rest of the area as well. The criticality of the small area would be undermined by the potentially predominant higher intensities. Thus, the average intensity would not reflect that without this small critical zone, the classification could result in a misclassification. This results in a list of the lowest intensities for each area in all images. A histogram of these lowest intensities is drawn to see the repartition of the critical features for each zone across all SAR images provided.

6.4.2 Results showing the individual contribution of the target, shadow and clutter to the classification

Results on the global information loss in regards of partly segmented images classification

	Visible areas in the images fed to the CNN						
Target area	X		X	X	X		
Shadow area	X	X		X		X	
Clutter area	X	X	X				X
Target	Classification scores achieved						
2S1	97%	5%	97%	4%	3%	1%	8%
BMP	96%	29%	60%	11%	7%	3%	9%
BRDM	99%	88%	13%	9%	3%	3%	37%
BTR60	97%	5%	44%	23%	30%	2%	2%
BTR70	100%	29%	98%	5%	6%	2%	70%
D7	100%	0%	15%	87%	88%	80%	0%
T62	99%	1%	60%	18%	26%	8%	0%
T72	97%	0%	94%	17%	16%	0%	0%
ZIL	97%	97%	74%	34%	36%	48%	97%
ZSU	99%	4%	72%	99%	99%	4%	4%
Total	98%	27%	65%	33%	34%	16%	23%

Table 6.1: Classification scores attained with partly hidden images.

The implementation of the method to evaluate the shadow, target and clutter contribution, described in Section 6.4.1, results in the classification scores obtained with partially masked images with a well-trained CNN in Table. 6.1. The results are target dependant as relative contributions of the various areas change greatly from one target to another. A higher level analysis of the detailed results is shown in Table 6.2.

Results suggest that the shadow is rarely used by the CNN despite contributing significantly to the classification of the BRDM, D7 and ZIL. Most of the time, the target and the clutter areas contain most of the information required for classification. In this case, the . The fact that the clutter area contains a lot of information for the CNN could either mean that multipath information is used or that the CNN learned the background correlation of the MSTAR SOC 10 dataset as seen in Section 2. This could be investigated further

with images of a segmented SAR dataset without background correlation. However, the SARBake segmentation is not supplied for the MSTAR EOCs.

Target	Most influential areas
2S1	Target and clutter (97%).
BMP	Target and clutter (60%).
BRDM	Shadow and clutter (88%).
BTR60	Target and clutter (44%).
BTR70	Target and clutter (98%), but mainly clutter (70%).
D7	Target and shadow (87%), no extra information from the clutter (0%).
T62	Target and clutter (60%).
T72	Target and clutter (94%).
ZIL	Clutter (97%). Also learnt independently target (36%) and shadow (48%).
ZSU	Target (99%).

Table 6.2: Analysis target per target of the most influential areas.

Results on the criticality of features found in the occlusion maps for the target, shadow and clutter area for a detailed analysis

The implementation of the method to evaluate the shadow, target and clutter contribution in more details, described in Section 6.4.1, results in the histograms in Fig. 6.6 obtained by associating the occlusion maps produced by a well-trained CNN with the image segmentation. The full analysis of all histograms of all targets can be found in Fig. C.1 of the appendix. Only the most significant histograms associated with the BMP, T62, ZIL are shown in Fig. 6.6.

Fig. 6.6 shows that each classified target contains in the large majority of cases at least one critical feature, apart from the ZIL and to a lesser extent the BTR70 and BRDM. These results confirm a stronger influence of the target area compared to the shadow area for classification (Section 6.4.2). ZIL appears the easiest target to recognise perhaps because it is only truck in the database (Chapter 2). A unique target in the dataset present distinctive features that are easier to spot even partly hidden. The shadow and clutter areas contain a lot less critical features than the target area, although the shadow is used consistently for the BMP and the D7. Clutter information is mainly used for the BMP and

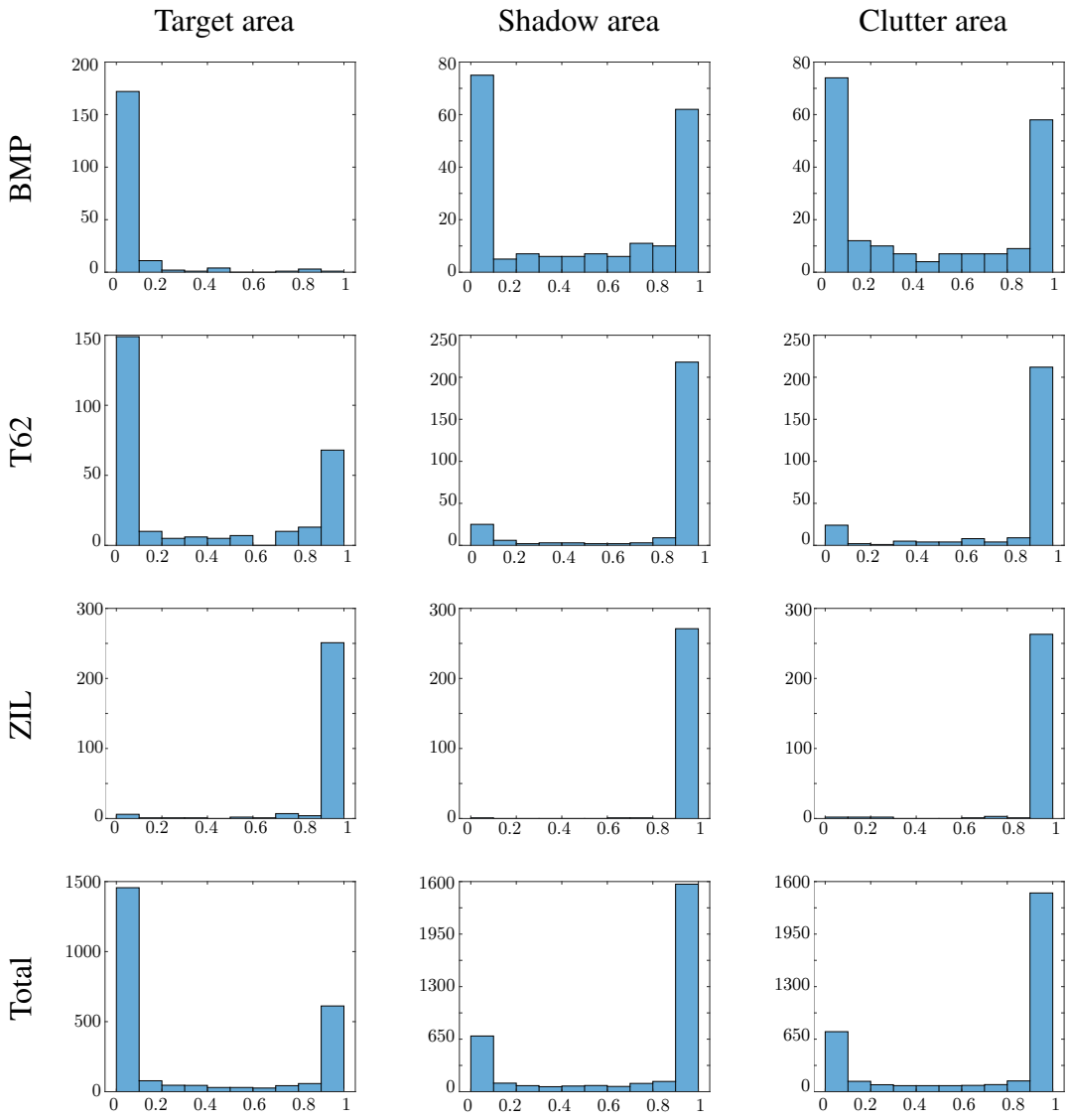


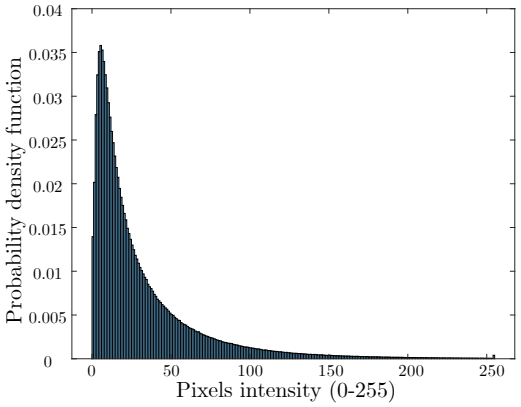
Fig. 6.6: Histograms of the most critical features (minimal intensity in the occlusion map) in each image per target per area of interest. The totality of the histograms for each target can be seen in annex in Fig. C.1

to a lesser extent to classify the 2S1 and the BRDM. It is still not possible, however, to conclude if the clutter area can be deemed critical because of the background correlation or other genuine reasons as target information from multipath effects.

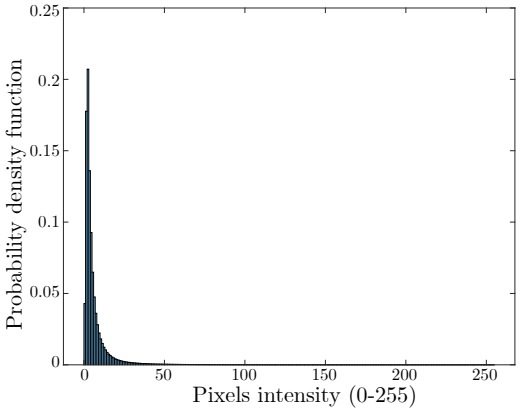
6.5 Study of the intensities of the pixels composing the critical features

In this section, the intensity repartition of the pixels that belong to the critical zones of the occlusion maps are compared to those in the less important zones. The objective is to assess if the network relies on parts of images with a specific intensity distribution compared to the rest of the image. If differences can be identified, they could be used by a pre-screener, after detection of the target, to focus the classification method on the potentially critical areas of the image. This would facilitate a better understanding of the deep learning classification process as well as enabling the more classical classification methods, such a feature-based classification methods, to compute descriptors directly on the potentially more interesting areas of the image.

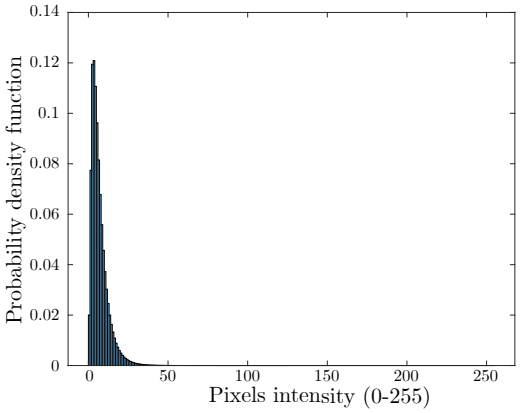
The intensity distribution is however also influenced by the area the studied pixels belong to as can be seen in Fig. 6.7 The target area is in the MSTAR databases the brightest zone of the SAR images as shown by the success of threshold based segmentations presented in Section 8. To avoid biasing the results, the analysis of the critical areas original pixel intensities will be focused on the target area only, discarding the other areas. The target area also contains the highest number of critical zones as seen in Section 6.4.2.



(a) Intensity repartition of the target area. The mean intensity is 30.6 and the standard deviation is 35.4.



(b) Intensity repartition of the shadow area. The mean intensity is 6.0 and the standard deviation is 10.8.



(c) Intensity repartition of the clutter area. The mean intensity is 6.4 and the standard deviation is 5.4.

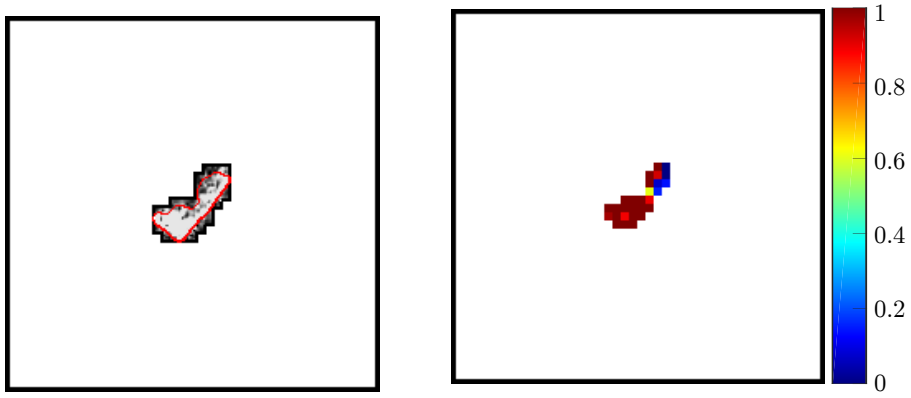
Fig. 6.7: Probability density function of the intensity of the target, shadow, clutter area determined by the SARBake segmentation on all test images of the MSTAR SOC 10.

6.5.1 Method to characterise the intensity repartition of the pixels in the critical zones for classification

Choice of the pixels whose intensities will be investigated

For this analysis, the plain MSTAR SOC 10 dataset from Section 6.3.1 will be used. The first step consists in determining which pixels of the original image correspond to the critical and unimportant zones in the occlusion map. When occlusion maps are computed in Section 6.3.1, 11×11 squares of the original image are hidden to determine the intensity of the 5×5 occlusion map central square. As explained previously, the objective is to

investigate the intensity repartition of the critical and non-critical areas in the target area. However, squares do not follow exactly the border of a target and therefore it is important to determine if the square belongs or not to the target area. If the 11×11 square in the original image overlaps with the target area determined by the SARbake segmentation for more than 50%, the intensities inside the square will be investigated to draw an intensity repartition of critical features. Fig. 6.8 (a) shows the proper target segmentation in red, and all the intensities in squares that will be potentially be investigated to draw the intensity repartition of critical areas in black.



(a) SARbake proper target segmentation in red compared to the pixels inside the target masked squares leading to the occlusion map on the right. These pixels will potentially be investigated for the intensity repartition of critical areas.

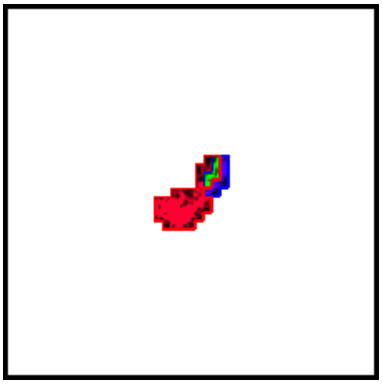
(b) 5×5 occlusion map obtained when the appropriate squares in the target area as seen in the left image are masked. The colour represents the probability of correct classification with the corresponding area masked.

Fig. 6.8: Comparison of the proper segmentation of the target and the target segmentation using the masked squares leading to the target occlusion map.

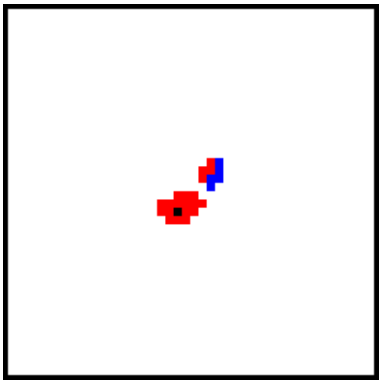
The corresponding central 5×5 occlusion map squares are also considered to belong to the target as shown in Fig. 6.8 (b). The whole target zone of the occlusion map will be investigated for critical and unimportant feature zones.

The values of the occlusion map are thresholded to isolate the two most extreme zones. Unimportant zones are the areas with a correct class probability over 90% in the occlusion map. Critical zones are the areas with a correct class probability under 60% in the occlusion map. Both of these zones are shown in Fig. 6.9 (a).

If the 5×5 block belongs to either zones in the occlusion map, the associated 11×11 pixels in the original image are retained. Fig. 6.9 (a) is an example of the pixels allocation to the critical and unimportant zone. Pixels in the original image can be assigned to both zones if they are at the border between both zones or to neither if the occlusion map correct class probability is in between the two thresholds.



(a) Pixels hidden while computing parts of the critical feature zone (blue) and parts of the unimportant feature zone (red). The green area consists of pixels hidden while computing both zones.



(b) Occlusion map target zone separated into the critical, in blue, and unimportant, in red, feature zone.

Fig. 6.9: Allocation of the pixels in the original image to the critical feature zone or the unimportant feature zone.

At the end, two pools of intensities of pixels are obtained in the target area, corresponding to the area that the CNN considered respectively as critical or unimportant, to achieve correct classification of the target for each input image as seen in Fig. 6.9 (b).

Computation of the histograms characterising the intensities of pixels in the critical and unimportant zones

Detailed histograms on the individual pixels intensities in the critical and unimportant zones Two histograms are drawn for all intensities of the pixels in the original image belonging respectively to the critical or the unimportant feature zones. Each test image will provide the original intensities of the pixels belonging to the critical and unimportant areas to the appropriate histogram. The comparison of the two intensity repar-

titions is used to assess differences between the critical and unimportant zones intensity repartition.

Global histograms on the statistics of pixels intensities in the critical and unimportant zones in each image This section presents a method based on statistics descriptors per image that complements the detailed intensity repartition presented earlier. The objective is to investigate if there are strong variations of intensity of the various areas from one image to another. The mean and standard deviation of the pixels intensities in the critical feature zone and the unimportant feature zone are extracted for each image. Histograms are drawn focusing either on the mean or the standard deviation of the critical and unimportant areas. One image will contribute with a unique value to each of these four histograms: the mean and standard deviation of the critical and unimportant area respectively. This is studied to investigate if the pixels in the critical feature zones have higher but also more spread intensities than pixels in the unimportant zones.

6.5.2 Results characterising the intensity repartition of the pixels in the critical zones for classification

Detailed histograms on the individual intensities of pixels in the critical and unimportant zones Fig. 6.10 shows that the distinction between the intensities of pixels in the critical and unimportant zone in the target cannot be achieved with a simple threshold, because most of the two histograms are superimposed. However, the two intensity repartitions in the critical and unimportant zones are different. The intensities of the pixels of the critical feature zone covers a wider and higher range of intensities than the pixels in the unimportant zones.

Global histograms on the statistics of pixels intensities in the critical and unimportant zones in each image Higher and wider range intensities in the critical zone is further shown in Fig. 6.11. Indeed, the average difference between the intensity mean

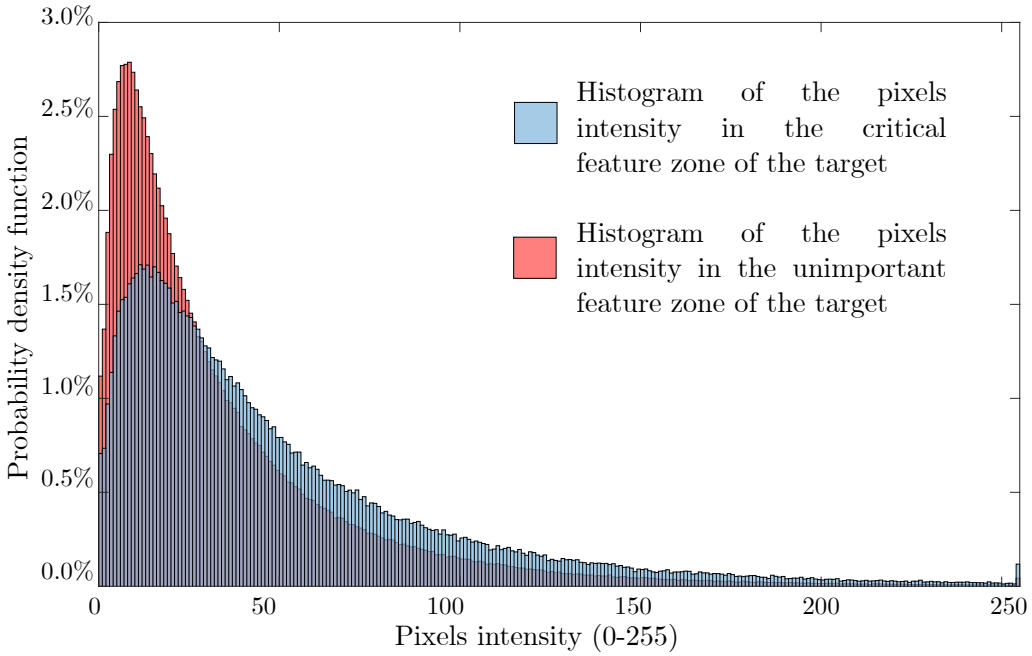
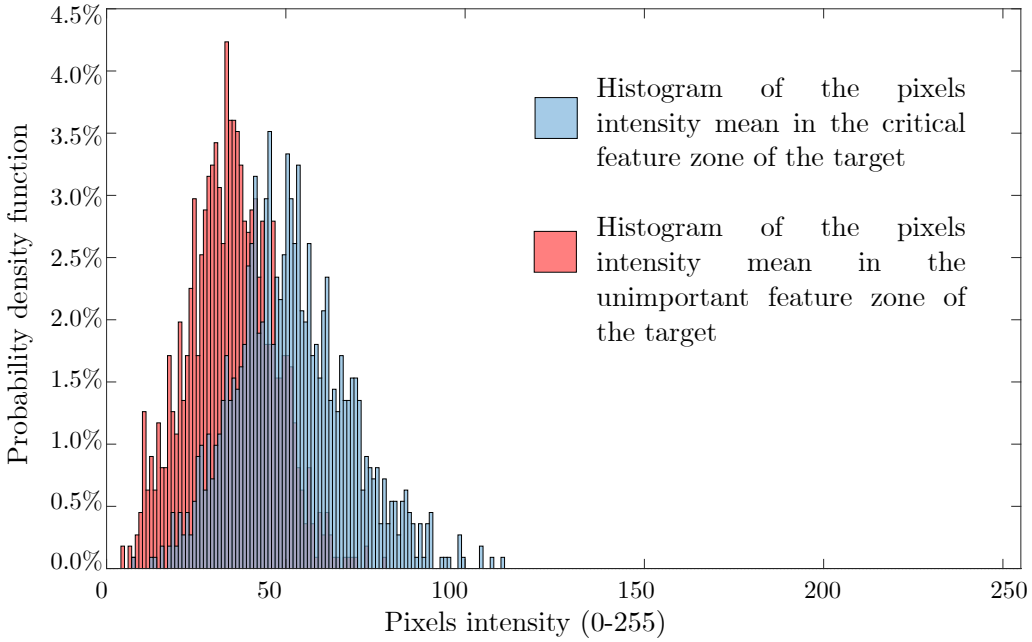
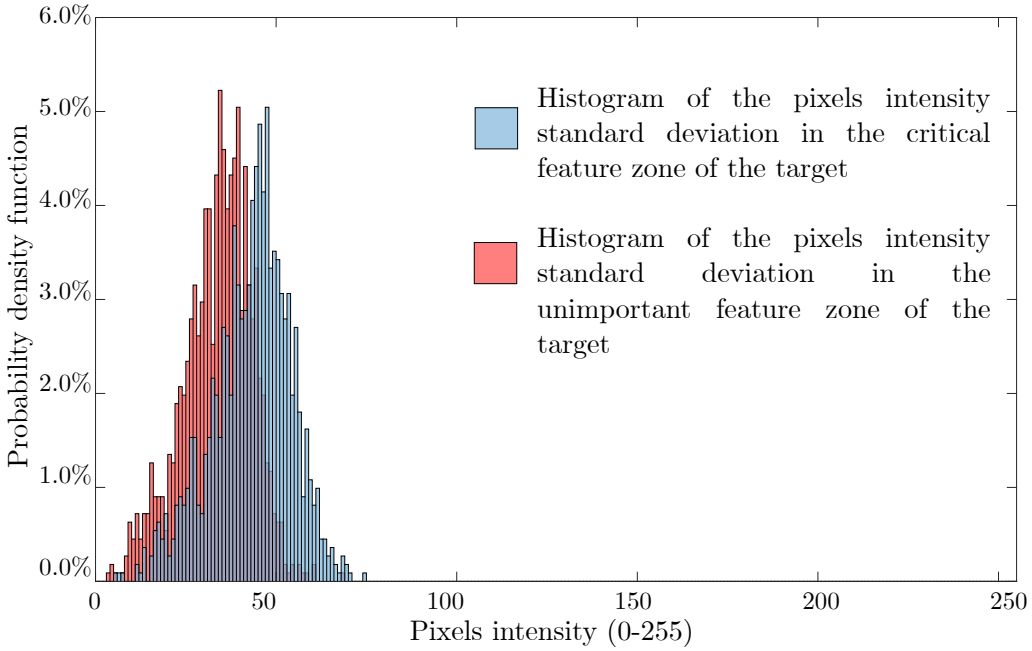


Fig. 6.10: Histograms of the pixel intensity repartition for pixels in the target area in the critical and unimportant feature zone across all test images.

distributions is 17.2 in Fig. 6.11 (a) and the average difference between the intensity standard deviation distributions is 9.4 in Fig. 6.11 (b) Pixels that provide crucial information have a higher mean, suggesting that the radar returns from these specific locations have more power than that from other areas. The standard deviation is also higher on average which means that, in critical areas, the range of intensities is larger than zones resulting in less interesting classification features. A high standard deviation area provides more diverse information that could be interpreted with complex kernels if this diverse intensity repartition is not due to a low SNR. The resulting convolution will span a wider range of values suitable for a finer activation and eventually more specific features for higher classification rates as long as the diversity of the features relates to the target specific rather than noise or speckle. Specificity of the feature is defined as the potential of a feature to be activated in only a few relevant occurrences, for example, only if the target is of a certain type.



(a) Histograms of the mean for intensities in the target area in the critical and unimportant feature zone per image.



(b) Histograms of the standard deviation for intensities in the target area in the critical and unimportant feature zone per image.

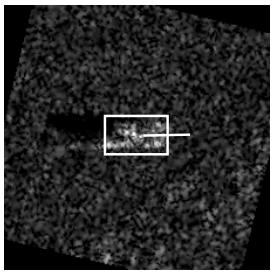
Fig. 6.11: Histograms of the statistics for intensities in pixels in the target area in the crucial and unimportant feature zone per image.

6.6 Influence of the target in the location of the critical features

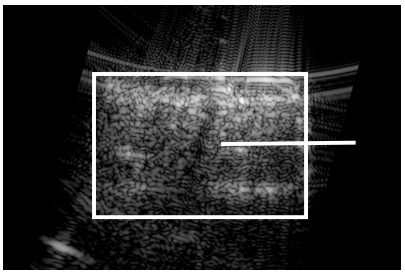
In order to achieve good classification, CNNs have to learn differences between targets. The specific targets characteristics in terms of location of critical features is studied for all targets. The objective is to determine which zone, for each target, is important for classification. These features are expected to vary especially between targets of a different type (tank, truck...).

6.6.1 Method using classification maps to see the location of the critical features for classification according to the target type

The classification maps are computed the same way as in Section 6.3.1. The images used are from the rotated MSAR SOC 10 dataset or the MGTD (Section 6.3.1) so that targets are aligned. The images are grouped to produce classification maps according to the target they represent. Thus, 10 classification maps are produced for the MSTAR dataset and 3 classification maps are produced for the MGTD. In all classification maps, all targets after being rotated are looking to the right. Fig. 6.3 shows the approximate position of the rotated target in both the MSTAR dataset and the MGTD.



(a) Approximate position of the target in an image of the MSTAR dataset.



(b) Approximate position of the target in an image of the MGTD.

Fig. 6.12: Approximate position of the target after rotation in the SAR images.

6.6.2 Results showing the classification maps and the critical zones for classification for each target type.

MSTAR dataset

Results obtained with a well-trained CNN Figs. 6.13 (a), 6.13 (b), 6.13 (h) and 6.13 (j) representing respectively the 2S1, BMP2, T72, ZSU show that the back-centre area of the target (centre left of the image) is the darkest area for tanks and armoured personnel carriers with the exception of Fig. 6.13 (g) representing a T62. It is the most critical area of the classification map and represents the highest and usually the most distinctive part of the target which corresponds to the turrets for the tanks. It is also true for the cabin of the D7 bulldozer in Fig. 6.13 (f). However, it is not noticeable on some of the other target types that do not have such prominent features. The central darker spot is then absent as can be seen in Figs. 6.13 (c) to 6.13 (e) representing respectively the BRDM, BTR60 and BTR70.

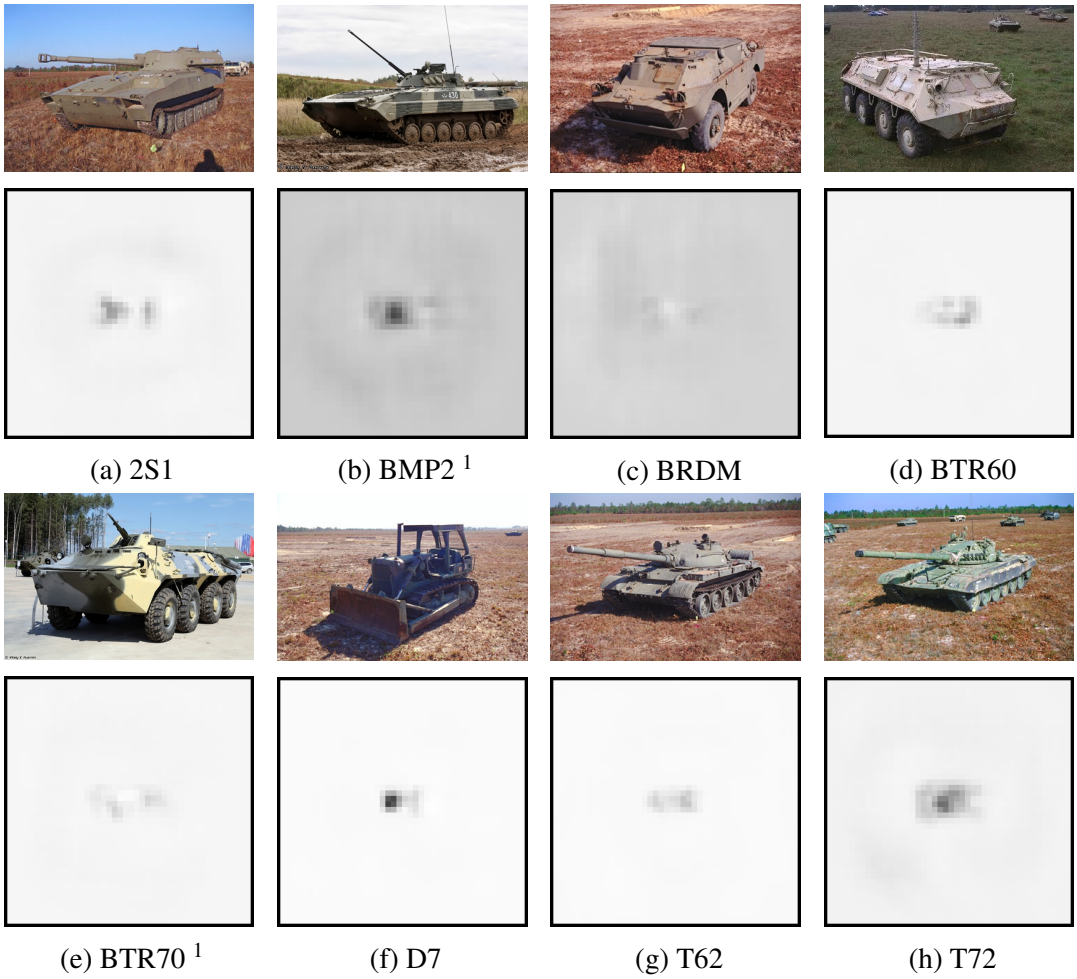
Some targets are also recognised with the very front of the target, and this is somehow expected for the bulldozer blade of the D7 as seen at the front of the target in Fig. 6.13 (f). The same occurs for the 2S1, BTR60, T62 and ZSU as seen respectively in Figs. 6.13 (a), 6.13 (d), 6.13 (g) and 6.13 (j).

The darker areas around the angles at the front of some targets in Figs. 6.13 (f) to 6.13 (h) could highlight the corners present in the targets.

The darker background in Figs. 6.13 (b) and 6.13 (c) for the BMP2 and BRDM shows that the CNN is less confident in the classification of these targets in general.

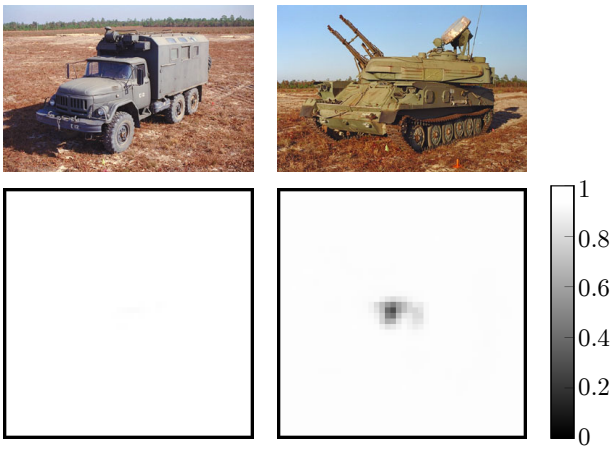
The fact that the target appears lighter than the rest of the image in Figs. 6.13 (c) and 6.13 (e), representing the BRDM and BTR70, shows the background correlation identified in Chapter 2. The background plays in this case a bigger role in the target classification than the target itself, which shows potential limit of ATR methods evalua-

[1]Photos of the BMP2 and BTR70 from the MSTAR dataset were not found and are replaced with alternative photos of the same tank models. These photos were taken by Vitaly Kuzmin (`https://www.vitalykuzmin.net`) and are licensed under the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License.



(a) 2S1 (b) BMP2 [1] (c) BRDM (d) BTR60

(e) BTR70 [1] (f) D7 (g) T62 (h) T72



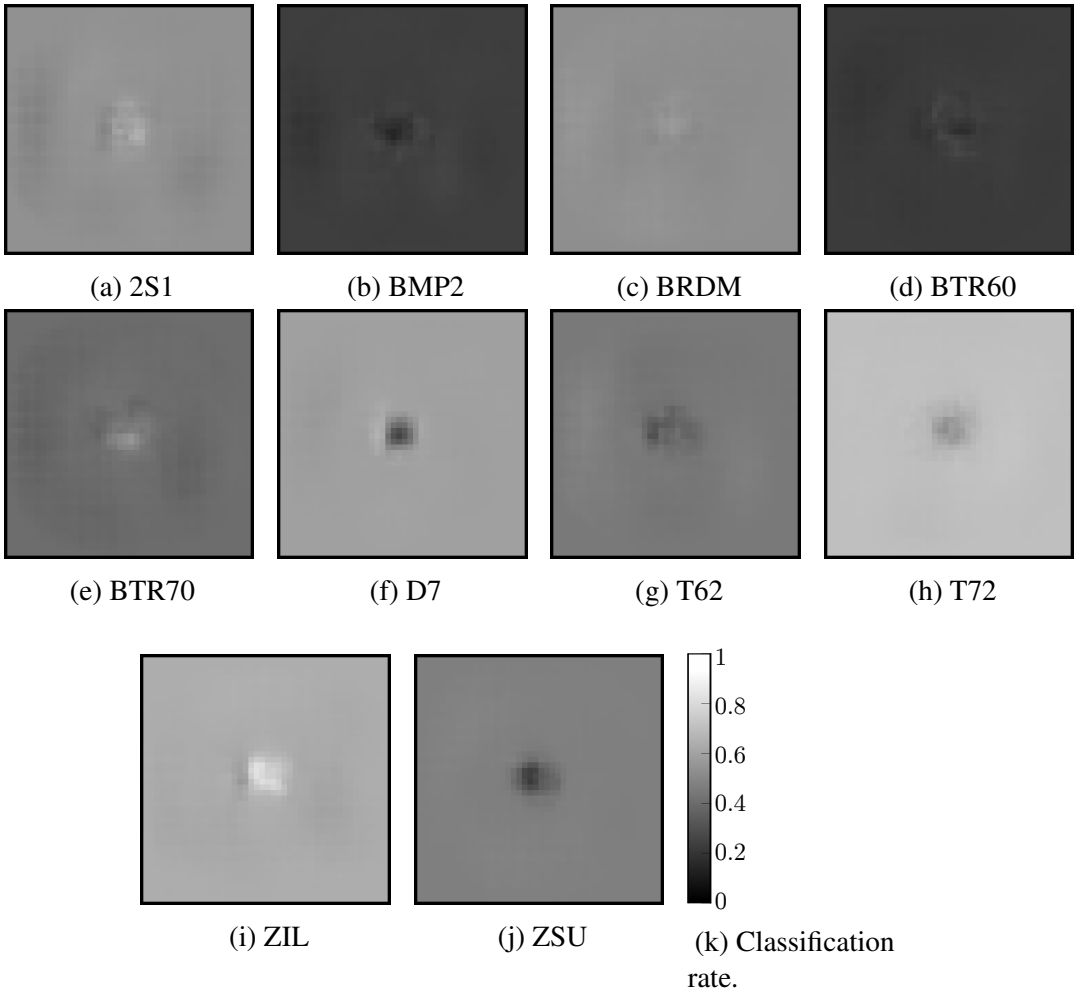
(i) ZIL (j) ZSU (k) Classification rate.

Fig. 6.13: Target classification maps with the original target image.

tion using the MSTAR SOC 10.

It seems that the ZIL in Fig. 6.13 (i) has no critical features. As the ZIL falls in the

longest targets of the database, the absence of critical features could be linked to one of the shortcomings of the classification map computation: the impossibility to take into account combinations of several features. Indeed, only a part of the target is hidden and, if features in different locations enable the classification, hiding only one of these critical features could leave the score of the correct target unchanged. Another possibility is that the CNN chooses the ZIL in case of a very uncertain prediction. In this case, the ZIL would be chosen whenever features related to specific target are not present.



(a) 2S1 (b) BMP2 (c) BRDM (d) BTR60

(e) BTR70 (f) D7 (g) T62 (h) T72

(i) ZIL (j) ZSU (k) Classification rate.

Fig. 6.14: Target classification maps with the original target image using a CNN trained without data augmentation.

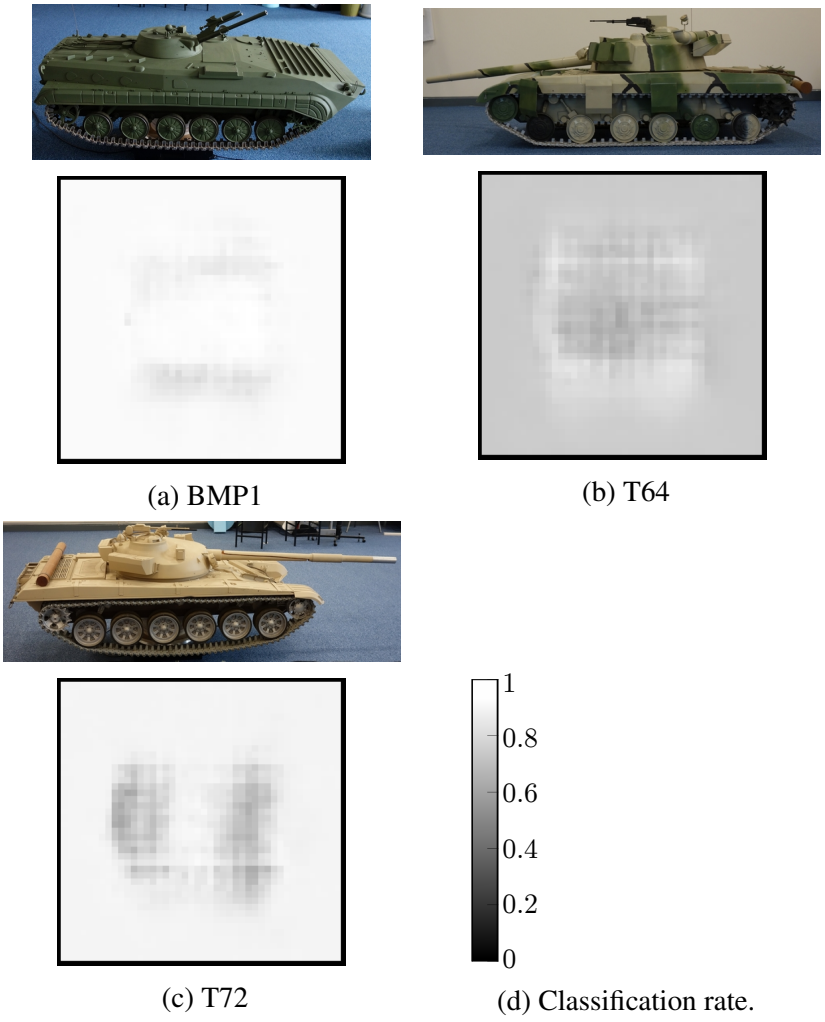
Results obtained with a CNN trained without data augmentation In order to better understand what areas are essential for classification, the location of the features critical

for classification for a well-trained network are compared to the location deemed critical by a network trained without data augmentation and less performing. The same process is applied to obtain the classification map but with the CNN trained without data augmentation and the results can be seen in Fig. 6.14. The images are overall darker as the probability of correct classification is lower. The CNN trained without data augmentation seems to rely, in some cases, more on the background than the target itself as it can be seen with the lighter shade in the 2S1, BRDM, BTR70 and ZIL. It is also noticed that the CNN did not narrow down the areas of importance as the CNN trained with data augmentation. The darker areas on the targets are larger and blurry. They are not focused on specific areas of the target as it could be seen for the CNN trained with data augmentation. Less explanations can be given for the classification choices of the CNN without data augmented training. The augmented training not only improves the classification score of the network but also improves its understandability. This is key, as the understandability of classification decisions is at least equally important to performances for implementing classification solutions under real conditions.

MGTD

Results obtained with a well-trained CNN The previous experiments are also conducted on the MGTD and give different results as shown in Fig. 6.15. The CNN focused on different areas for each target on the contrary to what happened in the MSTAR database where, for example, the higher turret central area seemed to be a focus point for the CNN. For the BMP1, it is the top and bottom parts of the Fig. 6.15 (a) which are darker and those correspond to the sides of the target. Fig. 6.15 (b) that represents the T64 target shows that the the CNN is focused on central part. The CNN highlights the front and back of the T72 target represented in Fig. 6.15 (c) as the right and left part of the image.

The darker classification maps for Figs. 6.15 (b) and 6.15 (c) representing the T64 and the T72, indicate that the confidence of the network in classifying these targets drops and that they are harder to classify. The CNN indeed is less likely to mistake the BMP1



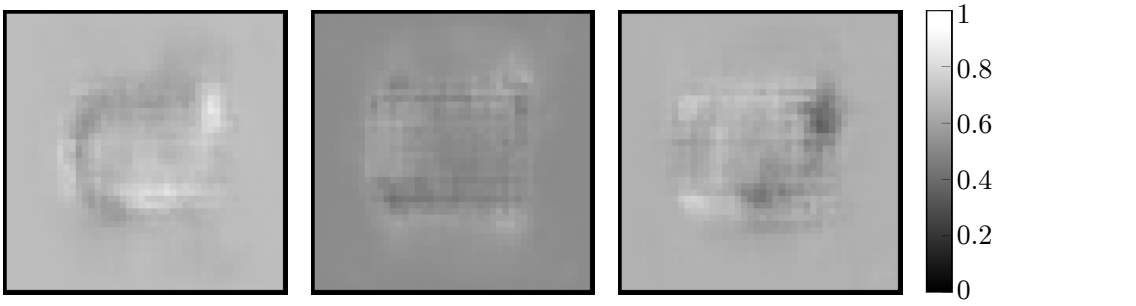
(a) BMP1 (b) T64

(c) T72 (d) Classification rate.

Fig. 6.15: Target classification maps with the original target image from the MGTD.

for another target than the T64 or the T72 because the latters are very similar and differ largely from the BMP1. This explains the higher confidence of the network in the BMP1 classification. The darker shade for the T64 confirms, with the confusion matrices shown in Chapter 5, that in case of a confusion between the T64 and the T72, the network will likely classify it as a T72.

Results obtained with a CNN trained without data augmentation To better understand the reasons of the location of the critical features, the location of critical areas found by a well-trained network is compared to those of a CNN trained without data augmentation. The results can be seen in Fig. 6.16. The darker images overall are due to the lower classification rate on the testing set achieved by the CNN trained without data aug-



(a) BMP1 (b) T64 (c) T72 (d) Classification rate.

Fig. 6.16: Target classification maps with the original target image from the MGTD with a CNN trained without data augmentation.

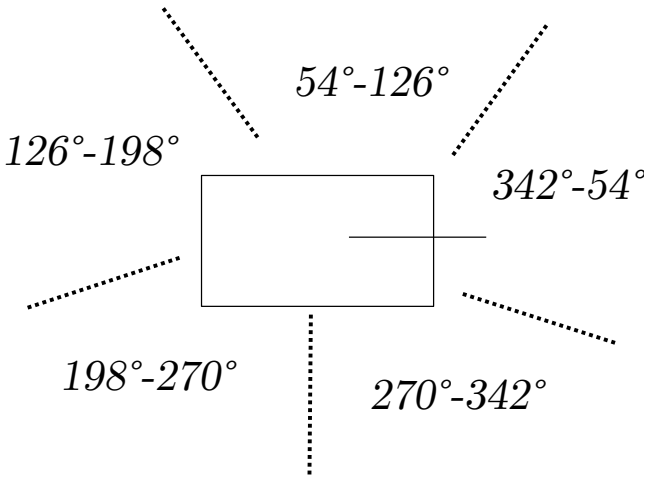
mentation. The two CNNs, one trained with and the other without data augmentation, concentrate on different areas for each target. However, the reasons behind the difference of location of the critical area seem more uncertain than for the MSTAR database. The first difference with the results achieved on the MSTAR is that the critical areas for each target are different (i.e. if the critical area of the T72 is its front, the front of other targets will not be their most critical area). This strategy could be because only 3 targets are present in the MGTD but this cannot be reproduced in the MSTAR which contains 10 targets. It would be interesting to see the evolution of the critical areas with the introduction of more target classes. The CNN without data augmentation focuses only on the front of the T72, the rear of the BMP1 and the sides and centre of the T64 whereas the well-trained CNN focuses respectively on the sides of the BMP1, the centre of the T64 and on both the front and the rear of the T72.

6.7 Influence of the orientation in the location of the critical features

The orientation or aspect angle of the target has an important impact on the appearance of the target in the image [19, 96]. This criteria is thus isolated to see its influence on the zones the CNN considers as important.

6.7.1 Method using classification maps to see the location of the critical features for classification according to the orientation of the target

(a) Target with a 0° orientation.



(b) Orientation ranges used to compute the 5 classification maps.

Fig. 6.17: Definition of the orientation ranges used to compute the orientation classification maps.

The classification maps are computed the same way as in Section 6.3.1. The images used are from both the rotated MSTAR SOC 10 dataset or the MGTD as in Section 6.3.1 so that the targets are aligned. The images are grouped to produce classification maps according to the orientation of the target it represents.

Five bins are chosen to represent the target azimuth groundtruth provided with each

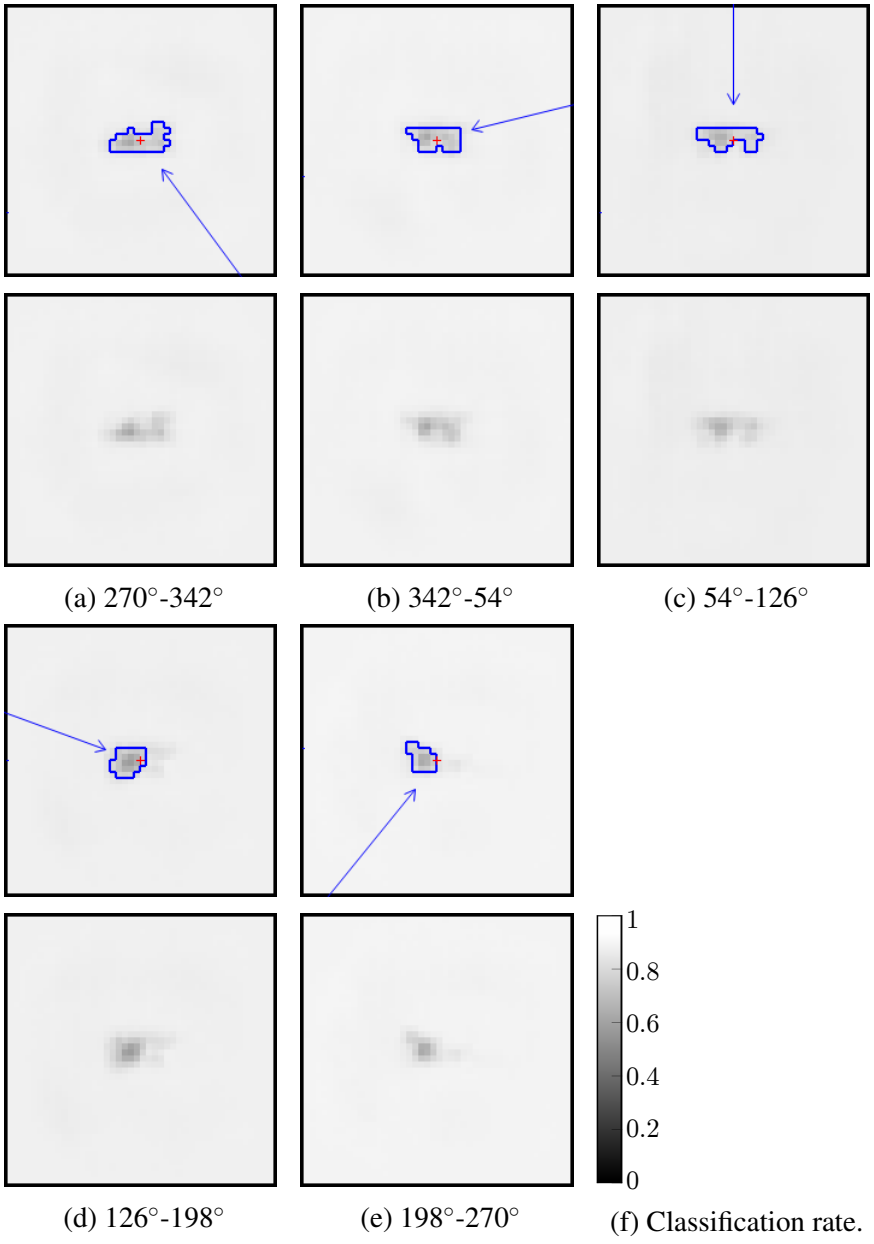
image starting from 0° and equally distributed up to 360°. The target looking to the right defines the new 0° arbitrarily in the rotated dataset. An example of the new 0° orientation is Fig. 6.17 (a). In this new frame of reference, the five groups of target orientations are as seen in Fig. 6.17 (b). Each group of images represent all the images with a target orientation belonging to one range bin. One classification map is computed for all these images. The result is 5 classification maps representing the 5 different orientation bins.

6.7.2 Results showing classification maps to see the influence of the target orientation on the location of the critical features for classification

In order to help the interpretation of these maps, a blue contour on the classification map around the lowest intensities and a red dot in the middle of the target are added. The blue arrow represents the direction of the main illumination, or centre of the orientation range bin, from the radar to the target.

MSTAR dataset

Results obtained with a well-trained CNN The classification maps obtained are summarised in Fig. 6.18 with and without the graphical help showing the critical features, the main illumination direction for each illumination range and the target centre. Fig. 6.18 (a) shows that the bottom right of the target is the most critical for a radar placed between 270° and 342°. This corresponds to the area with the best signal reflection. Because of the shape of tanks, the parts of the target facing the radar are likely to produce a specular reflection and therefore likely to reflect more the illumination than the sides perpendicular to the radar. The back side of the target is not directly illuminated at all but can be slightly highlighted through diffraction effects. Thus, the area surrounding the surface facing the radar, and the closest to the radar, is brighter in the SAR images. This area is also the critical area in most of classification maps with respectively Fig. 6.18 (c) highlighting the



(a) 270°-342° (b) 342°-54° (c) 54°-126°

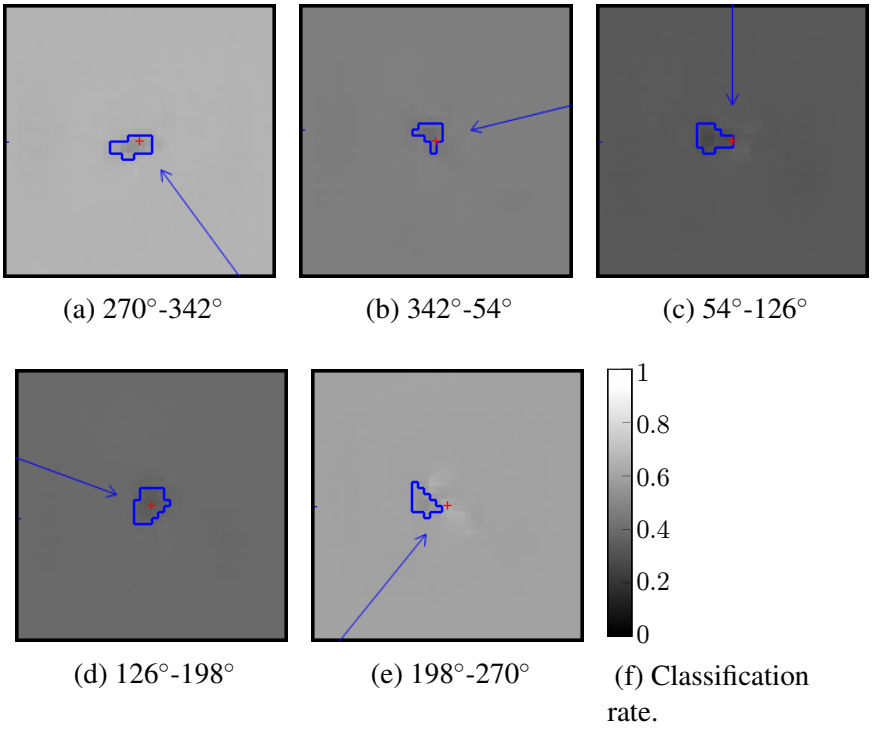
(d) 126°-198° (e) 198°-270° (f) Classification rate.

Fig. 6.18: Illumination direction and contour of the most critical areas in each orientation range classification map with the original classification map for the MSTAR SOC 10 targets.

top of the map, Fig. 6.18 (d) focusing on the left of the map and Fig. 6.18 (e) highlighting the bottom left of the map. It is however less clear for Fig. 6.18 (b) that the most critical area is the front of the target on the right of the map, even though this part is still critical. The areas reflecting the best the signal, usually in the area the closest to the radar as the front side faces the receiver, appear to be more critical.

It can also be noticed that the target rear, in the left part of the classification maps is

always highlighted. It is indeed always inside the blue contour, which shows the darkest parts of the classification maps. This higher area corresponds to the turret for the tanks and the cabin for the bulldozer. The rear part of the target was also highlighted as a critical area in Section 6.6.



(a) 270°-342° (b) 342°-54° (c) 54°-126°

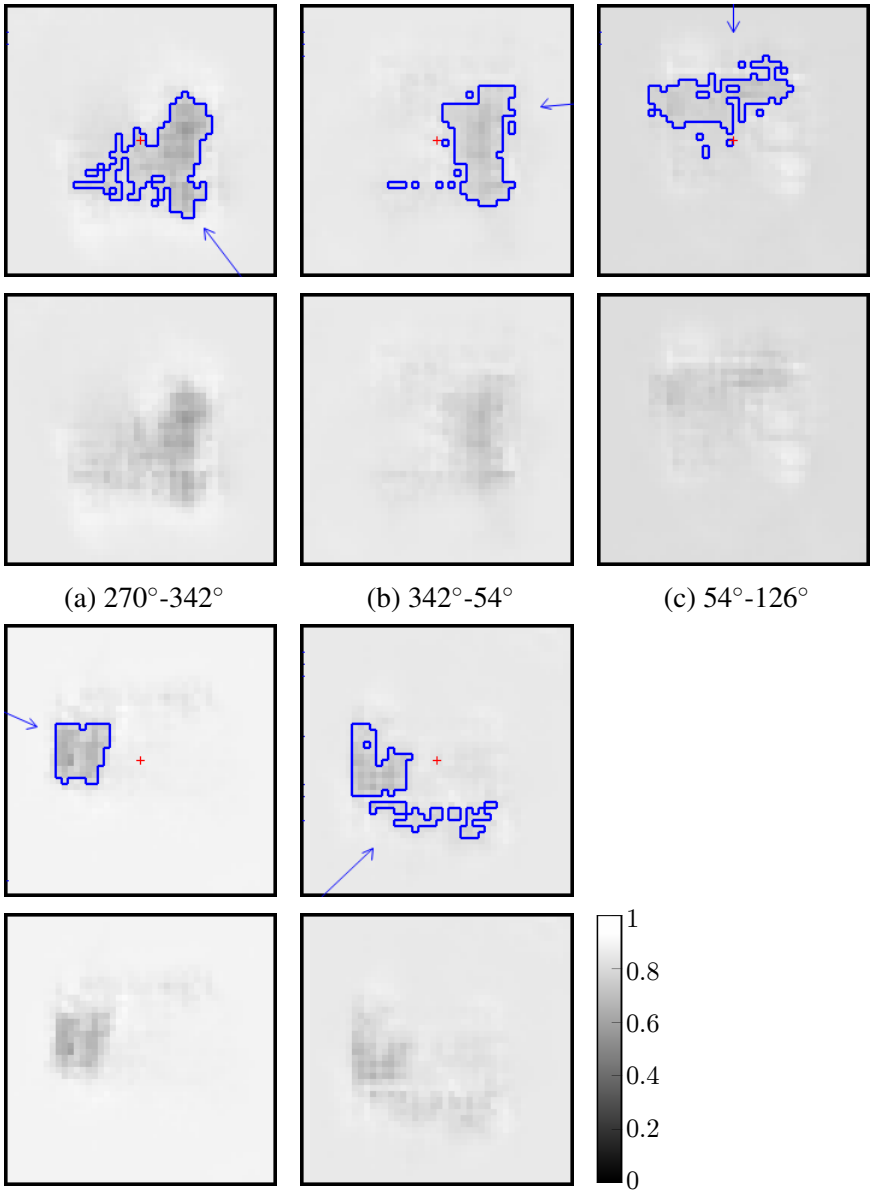
(d) 126°-198° (e) 198°-270° (f) Classification rate.

Fig. 6.19: Illumination direction and contour of the most critical areas in each orientation range classification map with the original classification map for the MSTAR SOC 10 targets obtained with a CNN trained without data augmentation.

Results obtained with a CNN trained without data augmentation In order to better understand what areas are essential for classification, the location of the features critical for classification for a well-trained network are compared to the location deemed critical by a network trained without data augmentation and less performing. The resulting classification maps can be seen in Fig. 6.19. The first thing that can be noticed is that the classification maps are overall darker, meaning that this CNN does not achieve the same quality of classification as the CNN with data augmented training. Moreover, the intensity on the target is not a lot darker compared to the intensity seen in the background area. The network seems to optimise less the information present in the target even though it is still

the most important area. It can be also seen that the darkest areas are not always on the area that is facing and the closest to the radar as previously such as in Fig. 6.19 (c). The critical zone is smaller and the rear of the target is not used in all orientations as it is the case in the CNN trained with data augmentation.

MGTD



(a) 270°-342° (b) 342°-54° (c) 54°-126°

(d) 126°-198° (e) 198°-270° (f) Classification rate.

Fig. 6.20: Illumination direction and contour of the most critical areas in each orientation range classification map with the original classification map for the MGTD.

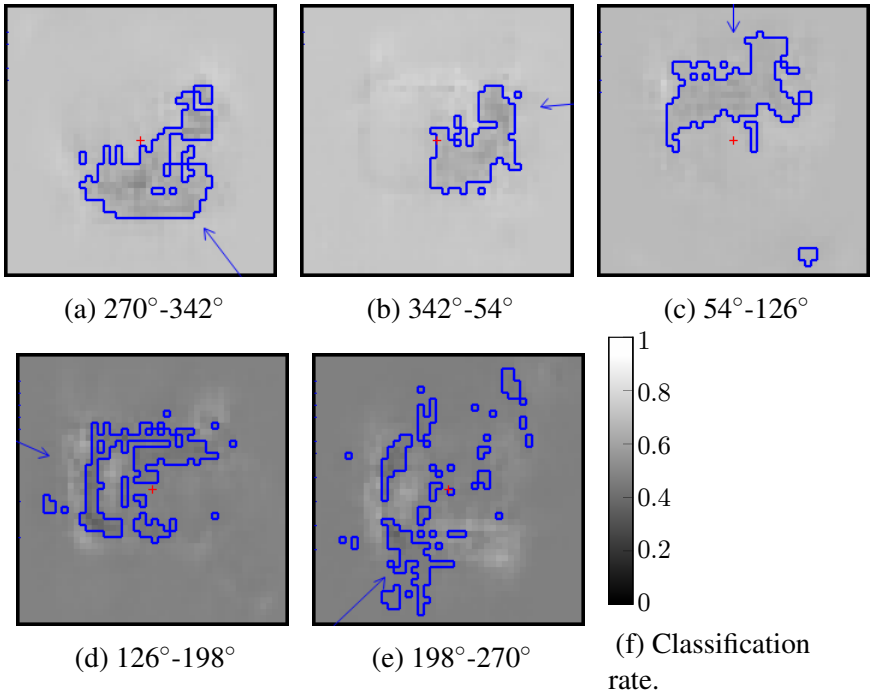
Results obtained with a well-trained CNN As for the MSTAR, classification maps relative to orientation ranges are produced using the data from the MGTD. As what was deduced on the MSTAR in Section 6.7.2, the critical areas are mostly located in the areas facing the radar as shown in Fig. 6.20. Indeed, Figs. 6.20 (a) to 6.20 (e) highlight respectively the bottom right, the right, the top, the left, the bottom-left of the map.

The results in Fig. 6.20 (a) show that, in this case, the classification relies on both the bottom-right but also the right and top-right of the map, that corresponds to the front of the target. The whole front of the target is used, even the further points that could be less illuminated. These further points are located around the same areas containing corners, that are likely to return signals directly in the direction of the emitter, and thus the receiver just next to it, because of their geometry. Corners are visible are present in the T72 and the T64 on the target front.

However, unlike for the MSTAR, the highest part of the target, on the left of the map is not always a critical area. Indeed, it is not always included in the blue contour which shows the most critical areas. This could be due to the different depression angle used to acquire both databases, or the turret material which is plastic in the MGTD and metal in the real targets in the MSTAR. Only the tracks of the model tank are in metal. Also, all targets in the MGTD have a round turret which minimise returns of the signal in the receiver direction compared to a planar area.

Results obtained with a CNN trained without data augmentation As for the MSTAR, classification maps related to orientation ranges in the MGTD are both created with a well-trained CNN and, here, with a CNN trained without data augmentation. The resulting classification maps can be seen in Fig. 6.21 (e). Results show that the classification maps are overall darker, as it was for the MSTAR, suggesting that the classification quality dropped over the whole testing set.

The location of the most important parts of the classification maps are relatively comparable for the first 3 ranges in Figs. 6.21 (a) to 6.21 (c). However, they are quite different



(a) 270°-342° (b) 342°-54° (c) 54°-126°

(d) 126°-198° (e) 198°-270° (f) Classification rate.

Fig. 6.21: Illumination direction and contour of the most critical areas in each orientation range classification map with the original classification map for the MGTD.

for the last 2 ranges in Figs. 6.21 (d) and 6.21 (e). Indeed, the CNN, in this case, does not seem to use the most illuminated area which should contain most of the information on the target. In Fig. 6.21 (e), parts of the background are also used. It is not known if the background is used because of correlation or a multipath effect. The well-trained CNN focuses on the target unlike the CNN trained without data augmentation.

As stated for the MSTAR, the features of the CNN trained with data augmentation seem better learnt. Besides achieving higher classification scores, the CNN trained with data augmentation can also be better understood. Indeed, its critical areas are focused on the target and especially on the target area surrounding the surface facing the radar, thus reflecting well the signal because of the geometry of the target. Indeed, the front surface facing the radar is more likely to reflect the signal towards the radar than the perpendicular or back surfaces. Having a better explainable network is essential if deep learning is to be implemented to operate in real scenarios.

6.8 Evolution of the features along the CNN depth

The previous sections investigate the location of the critical features. In this section, the specificity of the CNN features to a class and to a target orientation are examined. Specificity of the feature is defined as the potential of a feature to be activated in only a few relevant occurrences, for example only if the target is of a certain type. This will be conducted at different depth levels of the CNN as the complexity of features increases. Histograms summarising the most used features for specific targets or orientations are computed and compared. The histogram comparison shows the growing specificity of these features along the depth of the network as they become more complex.

6.8.1 Method to characterise the specificity of features along the network's depth

The images used are from the testing set of the MSTAR 10 SOC and MGTD plain dataset (the original images without centring or rotation) presented in Section 6.3.1. The CNNs are the same as those in Section 6.3.1, however they are trained on the plain training set. The CNN trained with data augmentation achieves 98.17% on the MSTAR SOC 10 and 92.47% MGTD while, without data augmentation, it only achieves 95.51% on the MSTAR SOC 10 and 78.53% on the MGTD. These scores are slightly higher than what was achieved in Chapter 5. Indeed, these networks are selected based on performances on the testing set rather than on the validation set as the focus of this work are the features that enable the network to achieve the best scores. In any case, the networks chosen must have a score on the validation set within 3% of their score on the training set.

The same steps will be carried out after each convolutional layer of the CNN to evaluate the evolution of the feature differentiation along the depth of the network as features become more complex.

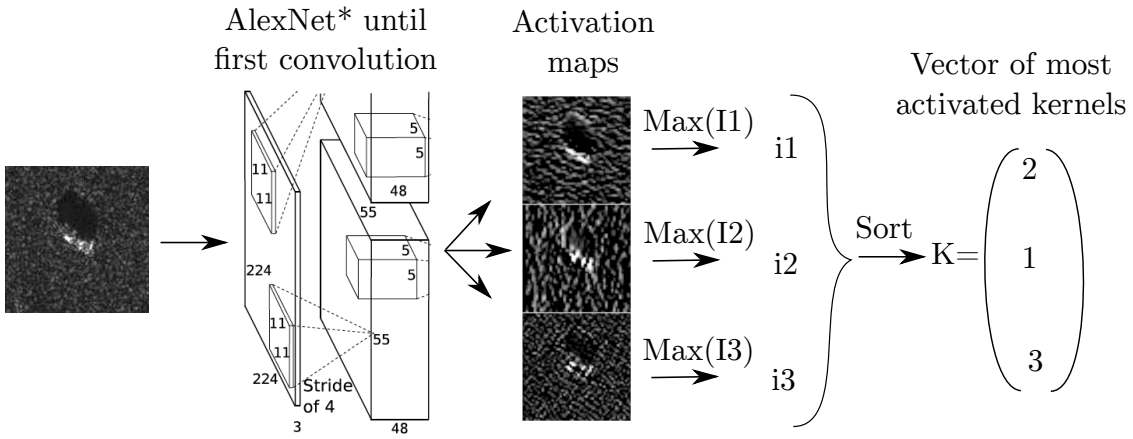


Fig. 6.22: Diagram of how the most influential kernels are determined for each image. *Image of AlexNet from [103].

Determination of the kernels associated with the strongest activations

The complete images are fed to the CNN. The intermediary activations or activations maps are extracted after the studied convolutional layer as seen in Fig. 6.22. The activation maps are the result of the convolution between the input (input image or previous activation map) and the kernels in the current convolutional layer. The maximum intensity of each activation map is then isolated. The kernels are then ranked according to the maximal intensity in their corresponding activation map. The kernel with the resulting highest intensity in its activation map will be the first in the vector of most activated kernels K in Fig. 6.22. Each image results in a vector K containing the number of each kernel from the most activated to the least activated activation map. That means that the first kernel leads to the strongest activation while the last kernel could result in a black map without any potential activation.

Histogram of the most used features

After the computation of the ordered vector K of the kernels leading to the strongest activations, the vector K is truncated to keep only the 20 best kernels as in Eq. (6.2).

$$\text{where } K(1:n) = \begin{bmatrix} k_1 \\ k_2 \\ \vdots \\ k_n \end{bmatrix} \tag{6.2}$$

Once the kernel lists have been produced for a group of images, a histogram of the frequency at which kernels are strongly activated by the network for a specific group of images is built as shown in Fig. 6.23. This histogram presents which kernel is mostly used in a group of images. These images can be grouped by target or orientation.

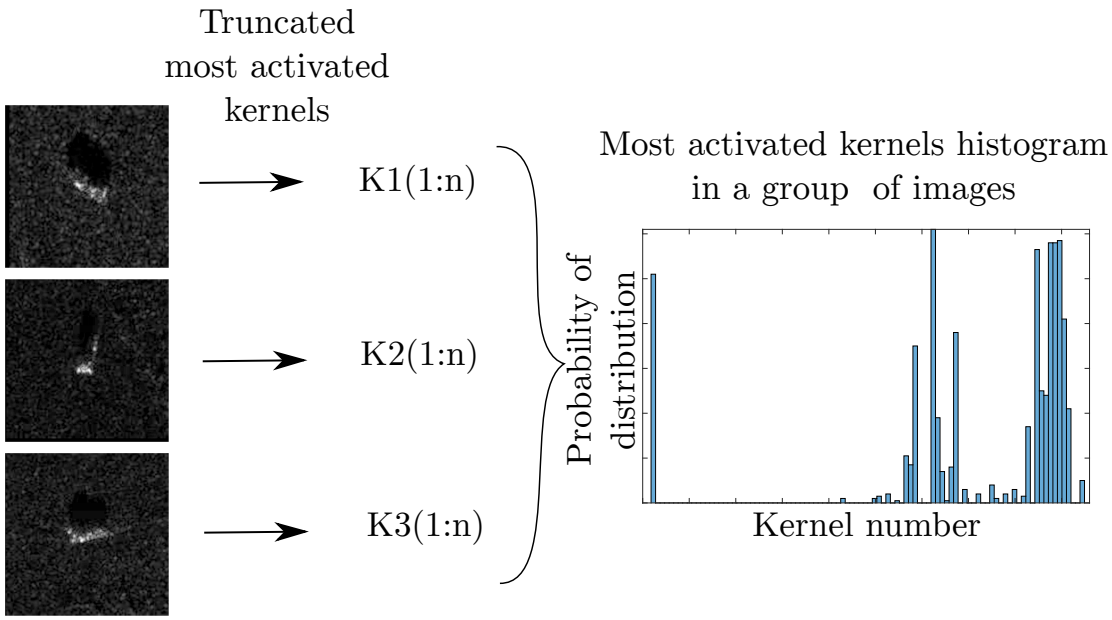


Fig. 6.23: Diagram representing the computation of the histogram of the most influential kernels for a group of images.

Comparison of the features mostly used by the CNN for a specific class (target class or target orientation)

The last step consists in an evaluation of the similarity or difference between the histograms produced with different groups of images. To that end, a normalised Chi-Square distance is introduced in Eq. (6.3).

$$D(H_{1,20}, H_{2,20}) = \frac{100}{2 \cdot m} \sum_{j=1}^{m} \frac{(H_{1,20}(j) - H_{2,20}(j))^2}{H_{1,20}(j) + H_{2,20}(j)} \tag{6.3}$$

where $H_{i,20}$ is the histogram of the lists of top kernels $K(1:20)$ for the images in the i^{th} group, and m is the number of kernels of this convolutional layer (ex: 96 for the first convolu--tional layer). It is also the number of bins of the histograms H_1 and H_2.

The Chi-Square distance is a common measure to evaluate the resemblance between histograms, and it is here normalised over the number of bins, so that this distance could be compared for histograms of different length as the number of kernels increases with the network depth. The average distance express the difference of feature representation by the network for a specific class or for a specific orientation. For example, 3 histograms using the method represented in Fig. 6.23 are computed using all the test images in the MGTD respectively specific to the T64, the T72 and the BMP1. These histograms are produced by investigating the activation maps generated after the first convolutional layer. The average of all normalised Chi-Square distances between the histograms (T64 with T72, T64 with BMP1, BMP1 with T72) gives an insight of the specificity of the kernels in the first layer to the target class. The bigger the distance, the more specific the features for the concerned target.

This distance will be computed after each convolutional network between all histograms generated by images with a specific target class or orientation. For the orientation "classes", 5 orientation categories are used as in Section 6.7. The distance evolution along the network depth is used to evaluate the state of differentiation of features specific to a

certain class.

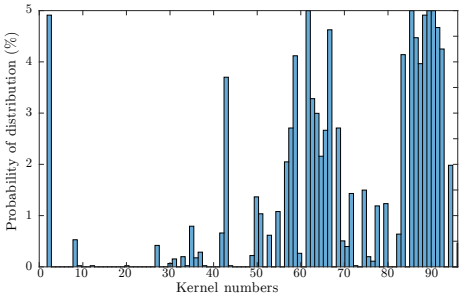
Distances are not only computed for specifically chosen groups of images but also with random groups of images of the same size to provide a control distance and ensure that the evolution of the distance is not only due to the feature complexification but really dependant of the common factor in the image group. This is also done to quantify how different the features are. Indeed, the lowest distance can never be null as long as the images are different in each group.

6.8.2 Results showing the differentiation of the features along the network depth

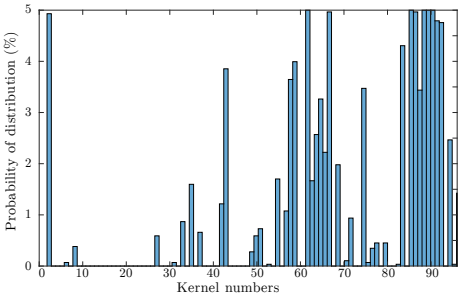
Histogram specificity to the orientation compared to target specific histograms

The first layer in the visually trained CNN provides very basic information such as intensity variation and direction, or the colours used in the initial image. Low-level features are not yet specific to a variable which explains the histogram similarity between a group of images focused on either the target class or orientation as seen in Fig. 6.24. However, with the depth of the network, the features become more complex and variable specific. The feature specialisation at the 5^{th} convolutional layer can be seen in Fig. 6.25 as the target specific histogram greatly differs from the orientation specific histogram as opposed as both histograms after the 1^{st} layer.

This shows that the features relative to the orientation are learnt on their own and are not only a by-product of the features learned for target classification. The CNN learned features specific to some orientations even if the training loss is dependant only on the target type. The orientation is thus key in the classification of SAR images. The features become overall more specific to each orientation group with the depth of the CNN. Following these observations, a network trained for target classification could easily be retrained to evaluate the orientation of a target.

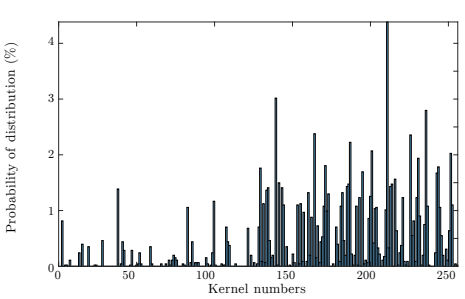


(a) Distribution of most active kernels after the first layer for 0°-72° images.

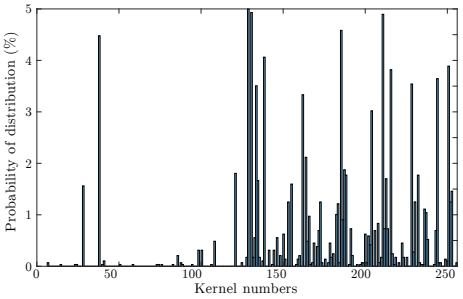


(b) Distribution of most active kernels after the first layer for BTR60 images.

Fig. 6.24: Histogram of the features specific to one target and one orientation range at the 1^{st} convolutional layer.



(a) Distribution of most active kernels after the fifth layer for 0°-72° images.

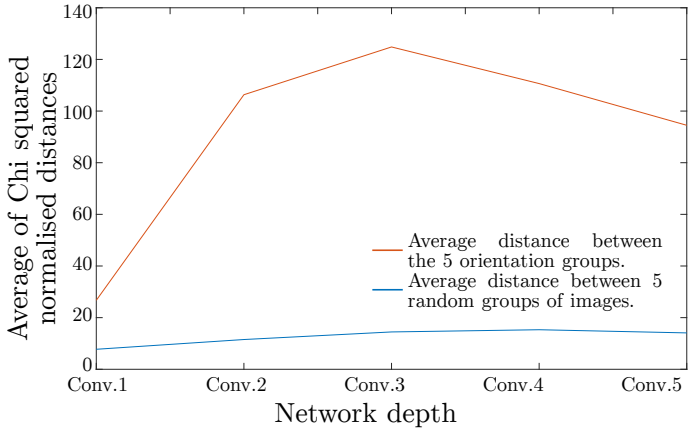


(b) Distribution of most active kernels after the fifth layer for BTR60 images.

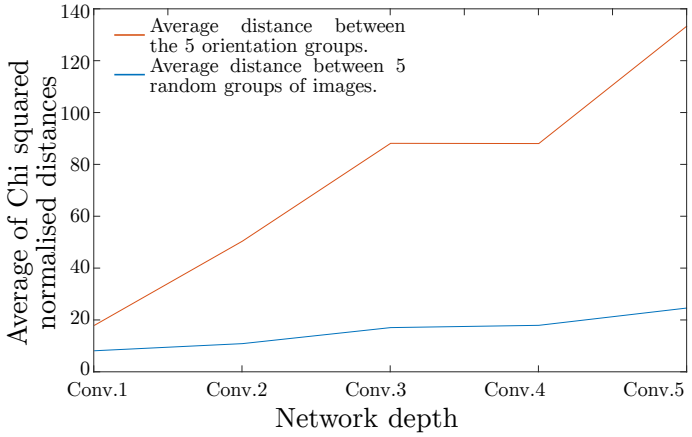
Fig. 6.25: Histogram of the features specific to one target and one orientation range at the 5^{th} convolutional layer.

Differentiation of features specific to a target orientation

Results obtained with a well-trained CNN It can be seen in Fig. 6.26 that both for the MSTAR dataset and the MGTD, the average distance between specific orientation groups is a lot higher than that between random groups of images (between 3 and 8 times higher). Images represent targets of different classes as well as different orientations. Thus, this high distance cannot be only a side effect of the network learning to recognise targets, which could be the case if some target classes were more represented with a specific orientation than others. The CNN specifically learned the orientation features. The network is able to learn environmental variable even when they are not included directly in the loss computation. The fact that the network is able to independently learn related environmental variables linked to the classification task is probably part of the



(a) MSTAR database.



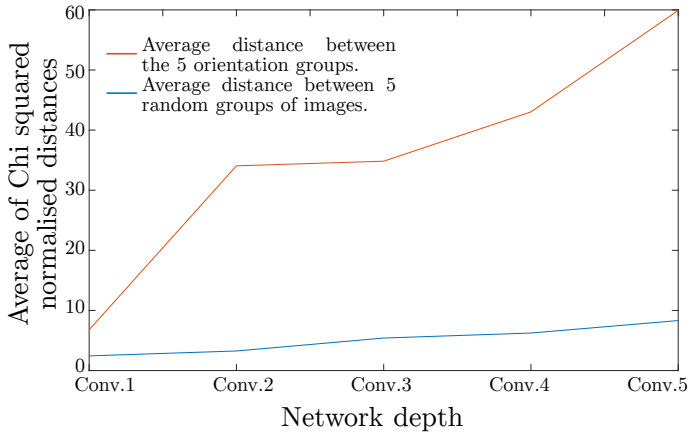
(b) MGTD.

Fig. 6.26: Average distance along the network's depth between histograms of kernels activated the most for 5 different orientation bins.

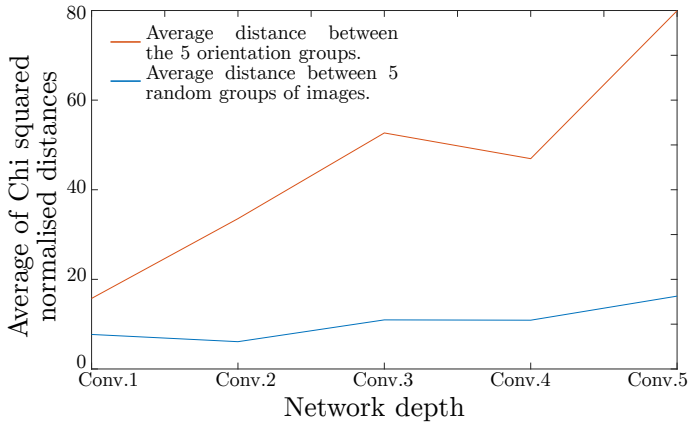
success of neural networks on SAR images which are affected by many variables. It puts into perspective the inclusion of external variables in the loss to force the network to learn about the target environment, as the network already carry this task to a certain extent on its own [97].

The creation of features specific to environmental variables without specific training also indicates that transfer learning could potentiality be pushed further. Instead of retraining a CNN to fit another database or different targets, the network could be re-purposed with entirely different output classes related to any environmental variable present in the dataset. Indeed, the network has probably already partly learned the appropriate features in addition to the features directly related to the initial task. In this case, a network finding

the target orientations could be quickly learned from a network dedicated to target type classification.



(a) MSTAR database.



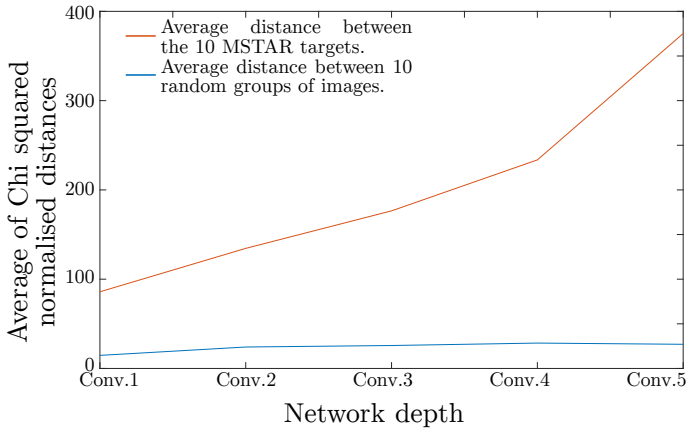
(b) MGTD.

Fig. 6.27: Average distance along the network's depth between histograms of kernels activated the most for 5 different orientation bins with networks trained without data augmentation.

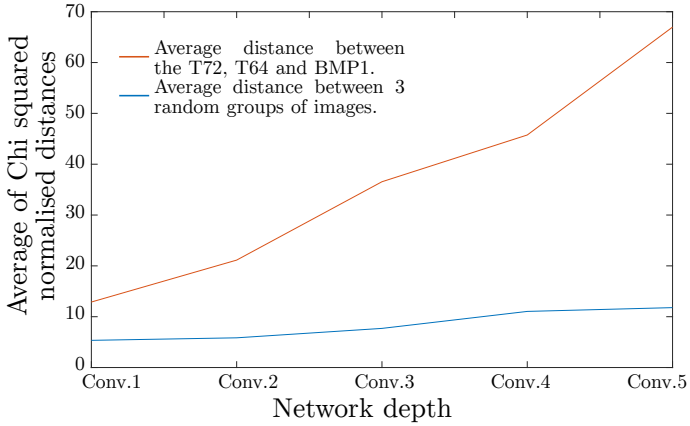
Results obtained with a CNN trained without data augmentation All of the above is conducted again with a CNN trained without data augmentation as can be seen in Fig. 6.27. The same conclusion can be drawn looking at the distance between orientation specific features learned by this CNN. The distance grows as the complexity increases with the depth of the network. It can be noticed, however, that the distance between features is lower at all depths for the CNN without data augmentation than for the well-trained CNN. There is a ratio between 1.6 and 2 between the distances generated using the

two CNNs with different training methods. The data augmentation created a more challenging training set which means more specific features had to be found in order to still be able to tell the target classes apart. The features learnt by the CNN trained with data augmentation, because they are more specific to each target, enable better classification as the targets can be more precisely described. Some of the learnt features relate also to the target orientation, hence the higher distances that can be seen not only in the distance of features specific to targets but also features specific to certain orientation ranges as seen in Fig. 6.26.

Differentiation of features specific to a target class



(a) Average distance between the histograms representing the most active kernels used in each MSTAR target group.

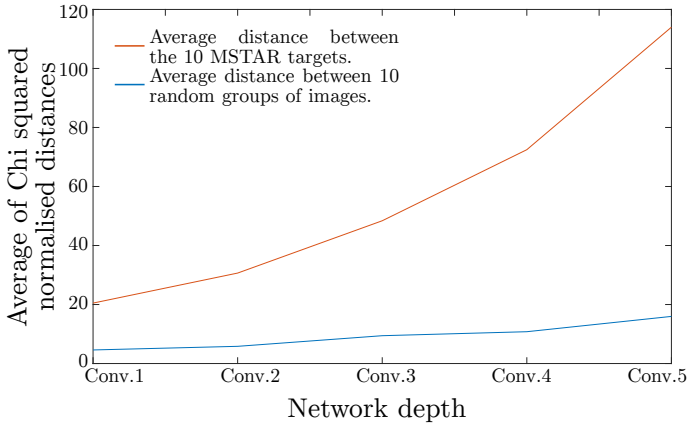


(b) Average distance between the histograms representing the most active kernels used in each MGTD target group.

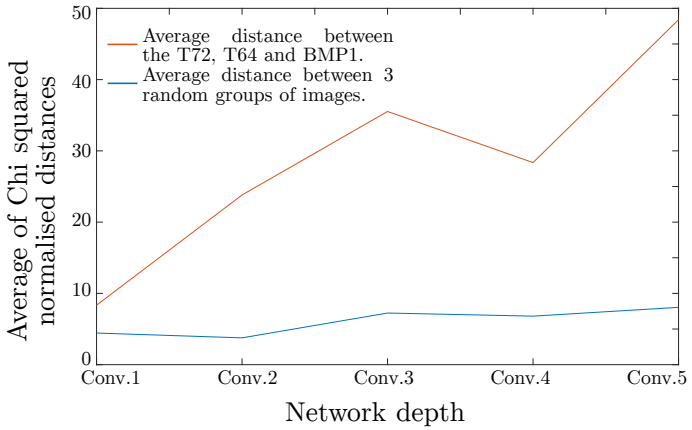
Fig. 6.28: Average distance along the depth of the CNN between groups of images of different targets.

Results obtained with a well-trained CNN The average distance between histograms representing the most active kernels specific to each target class grows constantly as shown in Fig. 6.28 and goes from 9 to 37 times the control distance in the MSTAR dataset and from 2 to 6 times the control distance in the MGTD. The CNN manages to increase the distance between targets class with features whose complexity reflects specificities of each target. The distance is four times higher in the MSTAR compared to the MGTD even though it contains 10 targets, whereas the MGTD contains only 3 targets, and thus should have been harder to differentiate the targets. The CNN trained on the MGTD is less able to tell the targets apart than the CNN trained on the MSTAR. Various hypothesis can be made on the reasons of a greater distance between targets on the MSTAR trained CNN. An uncompleted training could be argued for the CNN trained on the MGTD because of a lack of diverse training data. Indeed, both training and validation scores are more than 99% and the incentive to update the network weights is very low. Thus, the features will not be improved unless the training set is extended. Another possibility concerns the data itself. The data from the MGTD could be harder to classify as it is less correlated as shown in Chapter 2. It is also composed of model targets that are mainly made of hard plastic and not of metal, reflecting less clearly the emitted radar signal. Features seen on the SAR images could be thus less precise and lower the distance between targets. As the distance continues to increase even at the deepest layer, increasing the depth of the network could lead to better scores.

Results obtained with a CNN trained without data augmentation This process is repeated with CNNs trained without data augmentation and the specialisation of the features can be seen in Fig. 6.29. Similarly to the more robust CNN, the distance between features dedicated to a specific target class grows larger in the network. However, it can be noticed that those features could be even more specific as there is a ratio of 1.4 to 3.3 between the highest distance achieved between specific features with the CNN trained without data augmentation and the CNN trained with it. Data augmentation enhanced the



(a) Average distance between the histograms representing the most active kernels used in each MSTAR target group.



(b) Average distance between the histograms representing the most active kernels used in each MGTD target group.

Fig. 6.29: Average distance along the depth of the CNN between groups of images of different targets for the network trained without data augmentation.

specificity of features to the target class.

6.9 Limitations of the feature analysis carried out

This chapter provides some tools to better understand deep learning algorithms trained on SAR images and attempts a series of possible explanations and interpretations of the achieved results. However, the understanding and explanation of deep learning algorithms decisions remain difficult and the hypothesis remain mainly conjectures. In this section, the limitations of this work are pointed out in order for the reader to have a clear perspective on the results.

All of the above results are based on a single type of CNN for each database (MSTAR or MGTD, plain or rotated). Even though this CNN is well trained as it was selected for its high classification rate on the validation and testing set, it is only a single type of CNN and the results cannot be generalised to all CNNs before being tested on other CNNs with other various architectures.

Regarding the occlusion and classification maps, the choice of the mask size (11×11 in Section 6.3.1) is problematic as it will hide only features smaller than 3.3 m (30 cm$\times$11). Bigger features than that and only partly hidden are partially immune from the mask and the score obtained for the correct class will still take them into account. However, having a bigger mask would provoke a precision loss as it would become difficult to pinpoint the location of the features having an impact on the score. Moreover, features present over multiple locations are never totally hidden. One part only is hidden and the other parts can still influence the score of the correct target.

6.10 Conclusion

In this chapter, some insights are given on the decision process of one trained CNN on both the MSTAR dataset and the MGTD. The analyses are carried out on a single type of CNN. Other types of CNNs with a different training method or a different architecture could react differently.

The analysis begins by differentiating the influence of the target, shadow and back-

ground zone in the deep learning classification process. It appears that the shadow is mainly ignored by the CNN with an essential role in only 2 cases out of 10. The information important for classification is mainly taken from the target zone and completed by the clutter zone. However, the results do not make possible to globally assess the contribution of the shadow in terms of information relevant for the classification. Indeed, the impact of the shadow, target and background area can be very different from one target to another.

The areas that are the most critical for the classification have an intensity distribution with a higher average and a bigger standard deviation than the areas contributing little to the classification. The higher average confirms that the most important areas are those that reflected the most the signal due to the target geometry. The higher standard deviation would point out to specific surfaces of the target which reflect the signal in diverse directions improving the specificity of the features. The frequent focus of the CNN for the turret area would confirm this hypothesis as the turret is more specific to a target than the planar surfaces.

Classification maps showed that the most important areas for the CNN are often located on specific parts of the target. The location of these areas are also influenced by the orientation of the target during the measurements with the areas facing the radar mattering more. The important features are located on zones specific to each target but the higher parts of the target such as the cabin or turret were often a focus point. A network that benefits from data augmentation during training not only performed better but also its classification process can be better explained. This makes the usage of such networks more acceptable for real solutions from a safety and acceptability point of view.

It was expected and shown that the features become specific to a precise target as they increase in complexity with the network depth. Classes become more easily distinguishable which is anticipated with the trend to design always deeper networks. It is shown that without adapting the loss, only related to the targets class, the CNN still learns to build features specific to some environmental variables and in particular specific to the target orientation.

Chapter 7

Discussion and future work

Contents

The main objective of this work is to develop novel techniques to perform SAR ATR. In order to be able to evaluate the performance of SAR ATR algorithms properly, datasets for evaluation are presented in Chapter 2. Then, various SAR ATR methods are tested with the implementation of feature-based classification in Chapter 4 followed by the implementation of a deep learning method taking into account the target orientation issue of SAR in Chapter 5. Chapter 6 focuses on the analysis of the decisions of the deep neural network with a main interest in the location of the areas motivating the classification.

In this chapter, the main findings will be summarised, the limitations of the approaches taken will be addressed and further possible work will be suggested.

7.1 Research summary

There are multiple advantages to perform SAR ATR. Indeed, SAR data is robust against a wide range of weather conditions and can even be taken during the night. However, SAR data is challenging to analyse, even for experts. In addition to that, with the number of sensor increasing, humans have to handle more and more information. Thus, automation or even simplification of the decision process using ATR is important. Many challenges are faced while performing SAR ATR such as for example data changes when a change in the environment occurs, the impact of speckle, and the lack of varied training data due to the acquisition constraints.

Without a sufficient amount of varied SAR data, it is difficult to have robust classification methods, which will tend to overfit to the training data, especially for deep learning methods. A new ISAR dataset called the MGTD and acquired in a laboratory is thus proposed. It contains 3 different targets and images sequences obtained in different environments with different target configuration, depression angle and laboratory backgrounds between the training and testing set for a total of 1728 images.

As the amount of data is much higher and the applications for algorithms working in the optical domain are numerous, much more research has been carried out in this domain than in the SAR domain. One of our objective was to see to what extent the work carried out for the optical domain could be transferred to the SAR domain. A GMM segmentation adapted for SAR images with a single polarisation is proposed. The objective is to model the image background along the image sequence and thus isolate the target. Compared to a simpler threshold method, the recall rate increases from 55% to 88%. In addition, various features issued from the optical domain are compared in order to perform target classification by matching the most similar areas of the targets

between the training and testing set. Both gradient and binary features are tested and on the contrary to the optical domain, it is the binary features such as BRISK (92%) that perform the best and not the gradient features such as SIFT (57%) as they may be more affected by speckle. The influence of the target orientation is also shown with the matching of the target to recognise reduced to only the targets in the training set that share a similar orientation. The classification score is improved from 80% to 92%, showing the influence of the target orientation on its appearance.

Deep learning has improved the classification scores in the optical and SAR domain. Indeed, the features are more numerous and fit the training data better than the features developed by humans. Improvements to deep learning classification solutions are proposed in this work by taking into account SAR specificities. One of the main constraint for deep learning methods is the lack of massive and varied databases, especially in the SAR domain. Data augmentation increases artificially the number of images for the algorithm training by providing altered versions of the original images. A data augmentation is proposed that reproduces speckle noise in the images. A simple AlexNet achieves classification rates of 77%, 86% and 91% respectively without any data augmentation, with translation data augmentation only and with both the translation and noise based data augmentation on the MGTD. A deep learning architecture, named Pose-informed, that takes into account the target pose in its classification is proposed. The target orientation is first determined using a Hough transform and a CNN. This target orientation determination performs better than the former Rician model and does not need the target class beforehand. Once the target orientation determined, the target classification is assigned to a CNN specialised in the appropriate target orientation range. This method achieves higher classification rates on the MGTD, MSTAR SOC 10, MSTAR EOC 1, MSTAR EOC 2 and is outperformed only on the MSTAR EOC 3 by an AlexNet with respective deltas of 3.09%, 2.14%, 8.26%, 1.77% and -1.7%. A classification rate of 99% is achieved compared to 97% for the AlexNet on the MSTAR SOC 10.

If the main current focus point of SAR ATR research is the classification performances

using artificial intelligence, there is the emergence of questions relative to the lack of understanding of deep learning solutions decisions. Indeed, the implementation of deep learning methods in real scenarios is directly linked to the trust that can be had in such algorithms. The lack of understanding means that the reasoning leading to a decision could lack robustness and that the correction of errors is not straightforward. Several tools are proposed in order to explain the classifications made by an AlexNet.

- It is shown that the relative role of the shadow, target and background is relative to the target. However, overall, the shadow area is not exploited as much as expected, given its information on the target structure used by the operators.

- The distribution of the intensities of pixels in areas essential for the CNN to handle classification is different from other areas. However, it is difficult to distinguish those areas beforehand.

- The location of the zones essential for classification by the CNN are studied per target. For the MSTAR, the most important role in the classification is played by the higher areas such as the target turret or cabin and the front area such as the bulldozer blade. In the MGTD, the important zones are specific to each of the three targets. These important zones are not only studied for each target, but also for each target orientation range. It can be seen that it is the areas directly facing the radar and not occluded that gather most of the CNN attention.

- The specialisation of the features to the target class and target orientation along the CNN depth is also studied. It appears that not only do the features get specific to the target class, but also to the target orientation, even though it is not included directly in the loss which is optimised during the CNN training. As CNNs learn environmental conditions during training in addition to their main objective, transfer learning could be extended to benefit from bigger datasets from the same domain but intended for other uses.

7.2 Evaluation of SAR ATR methods

It is essential to ensure that the algorithm tested for SAR ATR are evaluated and compared fairly. However, the complex acquisition process of SAR and ISAR images makes the creation of an optimal dataset difficult. Currently, SAR ATR methods are evaluated on the MSTAR dataset. In Chapter 2, an alternative is proposed in the MGTD.

7.2.1 Lack of variety in the SAR ATR dataset

Overfitting generated by the lack of image diversity

In Chapter 2, it was shown that there is correlation between the training and testing sets of the MSTAR for standard conditions. This correlation can be in part attributed to the multipath, but the correlation extent shows that this is probably not the only cause. For the standard conditions, without the target present in the image, still 61% of the image were correctly classified using a simple method on the background only for the MSTAR SOC 10 containing 10 targets. On the MGTD, this score reached only 56% compared to 83% on the MSTAR SOC 3 dataset which has the same number of targets.

In addition, there is a lack of diversity in the training set as only one image sequence describes each target in the training set for the MSTAR SOC 10, EOC 1, EOC 2 and EOC 3. Thus only one specific version of the target, with one unique configuration and depression angle is present in training. Images in a sequence are taken one after the other and are thus not independent. As a result, choosing a good performing model is difficult as the validation set is a random subset of images taken from the same sequences present in training. Overfitting can be in this case not detected or even rewarded. The lack of diversity during training encourages the model to overfit. Few parameters vary between training and testing sets, especially for the MSTAR SOC.

In Chapter 2, an alternative dataset is thus proposed, with different target configurations and depression angles during training. There are thus 4 sequences of image for each target in the training set, with one of which that can be used independently for validation

of the model.

Difficulty to evaluate the performance of SAR ATR in real conditions

The tolerance to overfitting and homogeneity of the training and testing data leads to extremely high scores in the MSTAR. The classification scores saturate, with scores over 99% in the MSTAR SOC 10 and scores over 96-98% in the MSTAR EOCs, leaving little room for testing and comparing different methods. Possible over-fitting also makes the results of existing models questionable in real conditions. Thus, further work could be done in the direction of a generalisation of robustness tests.

7.2.2 Propositions concerning the dataset on which SAR ATR methods are evaluated

A robustness test against target translation has been carried out in Chapter 5. Even though the model trained in a standard way and the model trained with translation data augmentation perform equivalently on the standard testing set, the results are different on the testing set with randomly translated images. That shows the limitations of the current testing method that does not prepare for the robustness needed for a system to work under real conditions. Even though the MGTD provides new configurations and depression angles in the testing set, several other improvements could be made.

It would be interesting to have a publicly released improved testing set, to have a more challenging, but still standard way or evaluating the SAR ATR algorithms. As the acquisition of new real data is expensive in terms of time and means, some alternatives are proposed. Several transformations, even if synthetically produced, could be applied to the images in the testing sets in order to evaluate the robustness of the models proposed. These transformations could include translation but also occlusion by masking part of the target, or noise addition. The noise addition proposed initially for data augmentation in Chapter 5 could be applied to produce highly noisy test images. GAN can also be used to provide advanced simulations of realistic SAR images taken under diverse environmental

conditions with, for example, a variety of backgrounds [89].

7.3 Application of classification methods from the optical to the SAR domain

Chapters 4 and 5 show methods currently applied to the visual domain and investigate how these can be transferred to SAR. The specific properties of SAR images can make the transfer of such techniques from the visual domain difficult.

7.3.1 Visual feature classification

Speckle has a strong influence on detection and feature classification. It makes the detection of corners harder than in optical images and also corrupts the feature description. Binary features, that have less description capabilities, end up performing better than gradient features as seen in Chapter 4. The speckle is less described to the advantage of target characterisation. The proposed classification comparison for different features is limited by the number of features compared. Other features could be included to further the study such as features issued from wavelets [148, 149], or features issued from trained CNNs [75–77]. In addition, detectors specific to the SAR image and more robust against speckle could be designed. Indeed, the accurate description of the features cannot be achieved if the areas with important information are not located first. In Chapter 4, a grid is used as a detector to compute features at a constant distance. Out of the detection algorithms, AGAST performed the best. Even though the detection adapts to the environment and should not require additional training, it could be interesting to see if the performance of the whole classification method could be improved by training the decision trees of the AGAST on SAR images rather than on visual images.

Standard feature classification, and especially with a higher number of targets, is currently outperformed by deep learning methods. It can be thus interesting to use features determined by a trained CNN instead of standard features. Further work could also be

carried out on the input given to CNNs. Indeed, this work has shown that even with less information, binary features still achieve higher scores. Thus, to undermine the influence of speckle, images with less intensities could be supplied to the network. An option could also be to include a layer comparing intensities, rather than going through convolutional layers only.

7.3.2 Deep learning

Chapter 4 shows that the target orientation affects deeply the features leading to classification. Target features with a different orientation are not simply rotated and classification scores drop severely when trying to match targets with different orientations. Thus, a deep learning method is implemented which takes target orientation into account for classification. The classification is handled after the orientation determination by assigning the image classification to a neural network trained on targets with a similar orientation. Overall, the CNNs trained on a specific orientation range performed better than a standard CNN with the same architecture trained on images with all target orientations altogether. As the worst results occurred on the dataset on which the orientation determination was the poorest, it would be interesting to evaluate to what extent the classification score was damaged by attributing the image to the wrong pose-informed CNN (usually in the next range if the Hough transform failed or in the opposite range if the orientation CNN failed). A simpler method to determine the target orientation without requiring deep learning could be investigated according to the results on the damage on the classification score done by misattributing the image to the wrong pose-aware CNN.

The main limitation of the proposed architecture is the additional storage needs. Another aspect to investigate is thus the extension of the architecture proposed to other type of networks and particularly to shallower networks. The increase of performance could also be studied for more complex networks to see if the classification score can be increased even more.

Orientation is not the only environmental variable that can affect the target image.

This architecture could also be extended to adapt to other environmental changes, such as different depression angles or background types.

7.4 Influence of the acquisition environment on the classification scores

In Chapter 6, the features enabling target classification with the CNN are analysed. The parts of the image that are the most used to achieve classification for the various target classes and for the various target orientations are investigated. The feature adherence to specific areas of the image is studied to understand the extent of the contribution of the target, shadow and clutter. An analysis of the distributions of the features deemed of interest by the network compared to other parts of the image is also carried out. The strongest limitation of this analysis is that it is carried out on a single well-trained AlexNet network. The results were reproduced only for other AlexNet with similar training. However, in order to be able to generalise those results, similar analysis should be carried out on networks with other architectures or at least different training parameters.

If these results can be generalised, more robust results could be achieved by encouraging the CNN during training to better use the information contained in the image by focusing more on the target shadow, for example. Faster methods could also be implemented by focusing the feature analysis to specific parts of the target critical for target classification such as the most illuminated and highest areas of the target.

In Chapter 6, the specialisation of the features learnt by the CNN along its depth is shown. This specialisation is specific to the target classes learned, as choosing the wrong target will impact the loss function during training. The specialisation is also specific to other parameters that are not included in the loss function such as the target orientation. This is particularly interesting considering the lack of data for SAR ATR. If features influencing the environment without a direct link to the main objective of the network are partly learnt anyway, an option could be to do task transfer learning. The CNN would be

firstly trained to perform another task on data close to the data available for SAR ATR and then be trained with transfer learning to perform ATR on SAR dataset. An example could be to train a semantic pixel segmentation method such as an encoder-decoder on SAR satellite images in a first instance. Some data is available on that subject with the potential of intersecting satellite data with Google Earth optical images [150]. The encoder could serve as an already trained base to perform SAR ATR on the MSTAR, with a less challenging transfer learning, as the features learnt by the encoder would already be specific to SAR data. No further SAR data designed for ATR would be necessary while the classification performance of the network could be improved with a complete SAR training on more diverse data.

Appendix A

Generation of SAR and ISAR data for ATR

A.1 MGTD

A.1.1 Nomenclature

Each sequence is labelled with an identifier following this nomenclature : *x-vd* where *x* is the number of the sequence (1-66), *v* is the variant of the target (T64n, T64s, T64f, T72n, BMP1n) and *d* is the height of the antenna, linked to the depression angle of the sequence (*l* for the 1.54 m radar, *m* for the 1.63 m radar, *h* for the 1.72 m radar). An example of a sequence name could be *1-T64sh*. In this case, the sequence number is 1, the target used is the T64s and the radar height is 1.72 m.

A.1.2 Sequence details

No improvement was noticed by including the extra training in the training for the classification algorithms but still included it for potential further work.

Table A.1: Details of measurements. Part 1

Serie nb	Target	Target variant	Radar height	Radar range	Turret Orientation	Gun Orientation	Orientation correction	Laboratory background	Cables state	Dataset separation
1	T64	s	1.72 m	5.1 m	0°	down	0°	1	Good	-
2	T64	f	1.72 m	5.1 m	0°	down	0°	1	Good	-
3	T64	f	1.72 m	5.1 m	0°	up	0°	1	Good	-
4	T64	n	1.72 m	5.1 m	−45°	down	−1.5°	1	Good	-
5	T64	n	1.72 m	5.1 m	−45°	up	−1°	1	Good	-
6	T64	s	1.72 m	5.1 m	0°	up	0°	1	Good	-
7	T64	s	1.54 m	5.1 m	30°	down	0°	1	Good	-
8	T64	s	1.54 m	5.1 m	−30°	up	−1°	1	Good	-
9	T64	n	1.54 m	5.1 m	30°	up	−1°	1	Good	Testing
10	T64	n	1.54 m	5.1 m	0°	down	−1.5°	1	Good	Testing
11	T64	f	1.54 m	5.1 m	30°	down	0°	1	Good	-
12	T64	f	1.54 m	5.1 m	−30°	up	−1°	1	Good	-
13	T64	f	1.63 m	5.12 m	−30°	down	−1°	1	Good	-
14	T64	f	1.63 m	5.12 m	30°	up	3.5°	1	Good	-
15	T64	n	1.63 m	5.12 m	30°	up	3.5°	1	Good	Testing
16	T64	n	1.63 m	5.12 m	0°	down	1°	1	Good	Testing
17	T64	s	1.63 m	5.12 m	−30°	down	0°	1	Good	-
18	T64	s	1.63 m	5.12 m	30°	up	2.5°	1	Good	-
19	T72	n	1.72 m	5.16 m	30°	down	−1.5°	1	Good	-
20	T72	n	1.72 m	5.16 m	−30°	up	−1.5°	1	Good	-
21	T72	n	1.54 m	5.08 m	30°	up	−1.5°	1	Good	Testing
22	T72	n	1.54 m	5.08 m	0°	down	−1.5°	1	Good	Testing
23	T72	n	1.63 m	5.09 m	−30°	down	−1.5°	1	Good	Testing
24	T72	n	1.63 m	5.09 m	0°	up	−1.5°	1	Good	Testing
25	T64	n	1.72 m	5.18 m	90°	up	3.5°	1	Good	-
26	T64	f	1.72 m	5.18 m	−90°	down	0°	1	Good	-

Table A.2: Details of measurements. Part 2

Serie nb	Target	Target variant	Radar height	Radar range	Turret Orientation	Gun Orientation	Orientation correction	Laboratory background	Cables state	Dataset separation
27	BMP1	n	1.54 m	4.65 m	−30°	down	0°	2	Good	Testing
28	BMP1	n	1.54 m	4.65 m	30°	up	0°	2	Good	Testing
29	BMP1	n	1.63 m	4.7 m	−30°	up	0°	2	Good	Testing
30	BMP1	n	1.63 m	4.7 m	−45°	down	0°	2	Good	Testing
31	BMP1	n	1.72 m	4.75 m	0°	down	0°	2	Good	-
32	BMP1	n	1.72 m	4.75 m	−45°	down	0°	2	Good	-
33	BMP1	n	1.72 m	4.75 m	45°	up	0°	2	Good	-
34	T72	n	1.72 m	4.83 m	−45°	down	−2°	2	Good	-
35	T72	n	1.72 m	4.83 m	−45°	down	−1.5°	2	Good	-
36	T72	n	1.72 m	4.83 m	−90°	down	−1°	2	Good	-
37	T64	n	1.72 m	4.5 m	−45°	up	−1°	3	Noisy	Extra training
38	T64	n	1.72 m	4.5 m	45°	down	3°	3	Noisy	Extra training
39	T64	n	1.72 m	4.5 m	90°	down	7°	3	Noisy	Extra training
40	T64	n	1.72 m	4.5 m	−90°	up	2.5°	3	Noisy	Extra training
41	T72	n	1.72 m	4.63 m	45°	down	3°	3	Noisy	Extra training
42	T72	n	1.72 m	4.63 m	−45°	up	3°	3	Noisy	Extra training
43	T72	n	1.72 m	4.63 m	−90°	down	5°	3	Noisy	Extra training
44	T72	n	1.72 m	4.63 m	90°	up	3.5°	3	Noisy	Extra training

Table A.3: Details of measurements. Part 3

Serie nb	Target	Target variant	Radar height	Radar range	Turret Orientation	Gun Orientation	Orientation correction	Laboratory background	Cables state	Dataset separation
45	BMP1	n	1.72 m	4.43 m	45°	down	0°	3	Noisy	Extra training
46	BMP1	n	1.72 m	4.43 m	−45°	up	−1°	3	Noisy	Extra training
47	BMP1	n	1.72 m	4.43 m	−90°	down	−1.5°	3	Noisy	Extra training
48	BMP1	n	1.72 m	4.43 m	90°	up	3.5°	3	Noisy	Extra training
49	BMP1	n	1.72 m	4.45 m	45°	down	0°	3	Good	Training
50	BMP1	n	1.72 m	4.45 m	−45°	up	0°	3	Good	Training
51	BMP1	n	1.72 m	4.45 m	−90°	down	0°	3	Good	Training
52	BMP1	n	1.72 m	4.45 m	90°	up	0°	3	Good	Training
53	T72	n	1.72 m	4.39 m	−90°	down	0°	3	Good	Training
54	T72	n	1.72 m	4.39 m	90°	up	0°	3	Good	Training
55	T72	n	1.72 m	4.39 m	45°	down	0°	3	Good	Training
56	T72	n	1.72 m	4.39 m	−45°	up	0°	3	Good	Training
57	T64	n	1.72 m	4.26 m	−90°	down	0°	3	Good	Corrupted
58	T64	n	1.72 m	4.26 m	90°	up	0°	3	Good	Corrupted
59	T64	n	1.72 m	4.26 m	45°	down	0°	3	Good	Corrupted
60	T64	n	1.72 m	4.26 m	−45°	up	0°	3	Good	Corrupted
61	T64	n	1.72 m	4.28 m	90°	up	0°	3	Good	Corrupted
62	T64	n	1.72 m	4.28 m	45°	down	0°	3	Good	Corrupted
63	T64	n	1.72 m	4.33 m	−90°	down	-3°	3	Good	Training
64	T64	n	1.72 m	4.33 m	−45°	up	0°	3	Good	Training
65	T64	n	1.72 m	4.33 m	90°	up	0°	3	Good	Training
66	T64	n	1.72 m	4.33 m	45°	down	2.5°	3	Good	Training

Appendix B

Deep learning classification

As presented in Section 5.3.2, these graphs shows the investigation to achieve the best learning rates on the MSTAR EOCs. In the end, the same learning rate as in the MSTAR SOC 10 is used. The accuracy achieved is achieved after 5 epochs and the choice of the learning rate is random inside a pre-defined range.

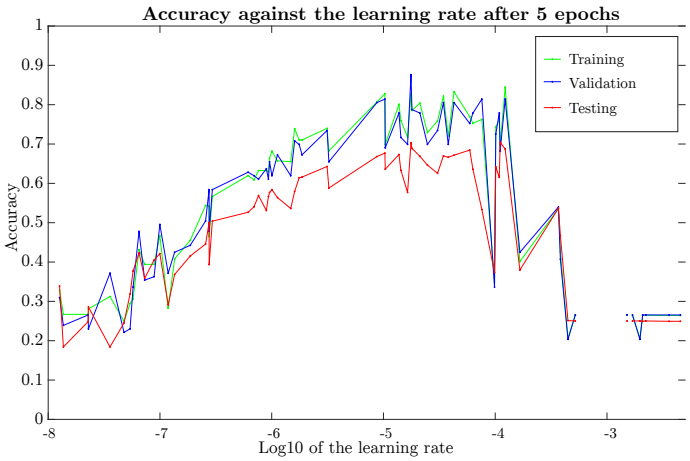


Fig. B.1: Learning rate study for the EOC 1 MSTAR dataset.

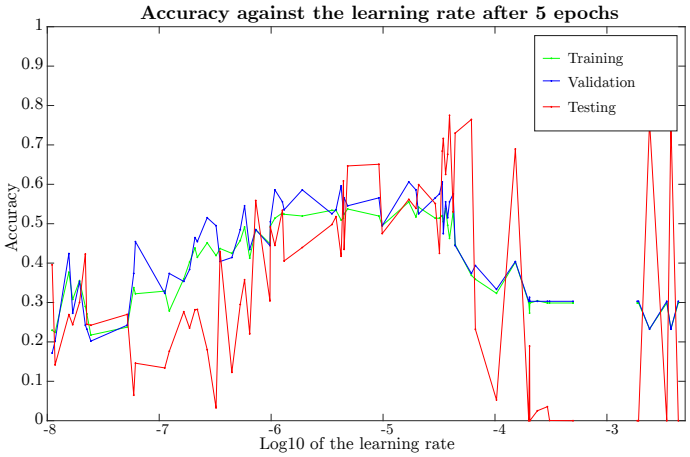


Fig. B.2: Learning rate study for the EOC 2 MSTAR dataset.

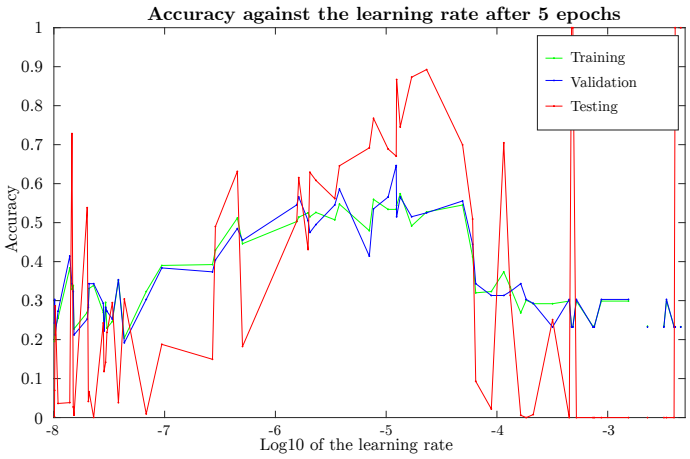


Fig. B.3: Learning rate study for the EOC 3 MSTAR dataset.

Appendix C

Deep learning network explainability through feature analysis

Here are presented the the full results partially described in Section 6.4.2. The histograms represent for each area of the image, i.e. the target, shadow of clutter area of the image, the criticality of the features present for the deep network. Low intensities means that the probability of correct classification is low for the target and that this feature is deemed essential to the network. High intensities show that those features are not critical. The histograms are diverse for each target but it appears that the main concentration of critical features is in the target area followed by the clutter area.

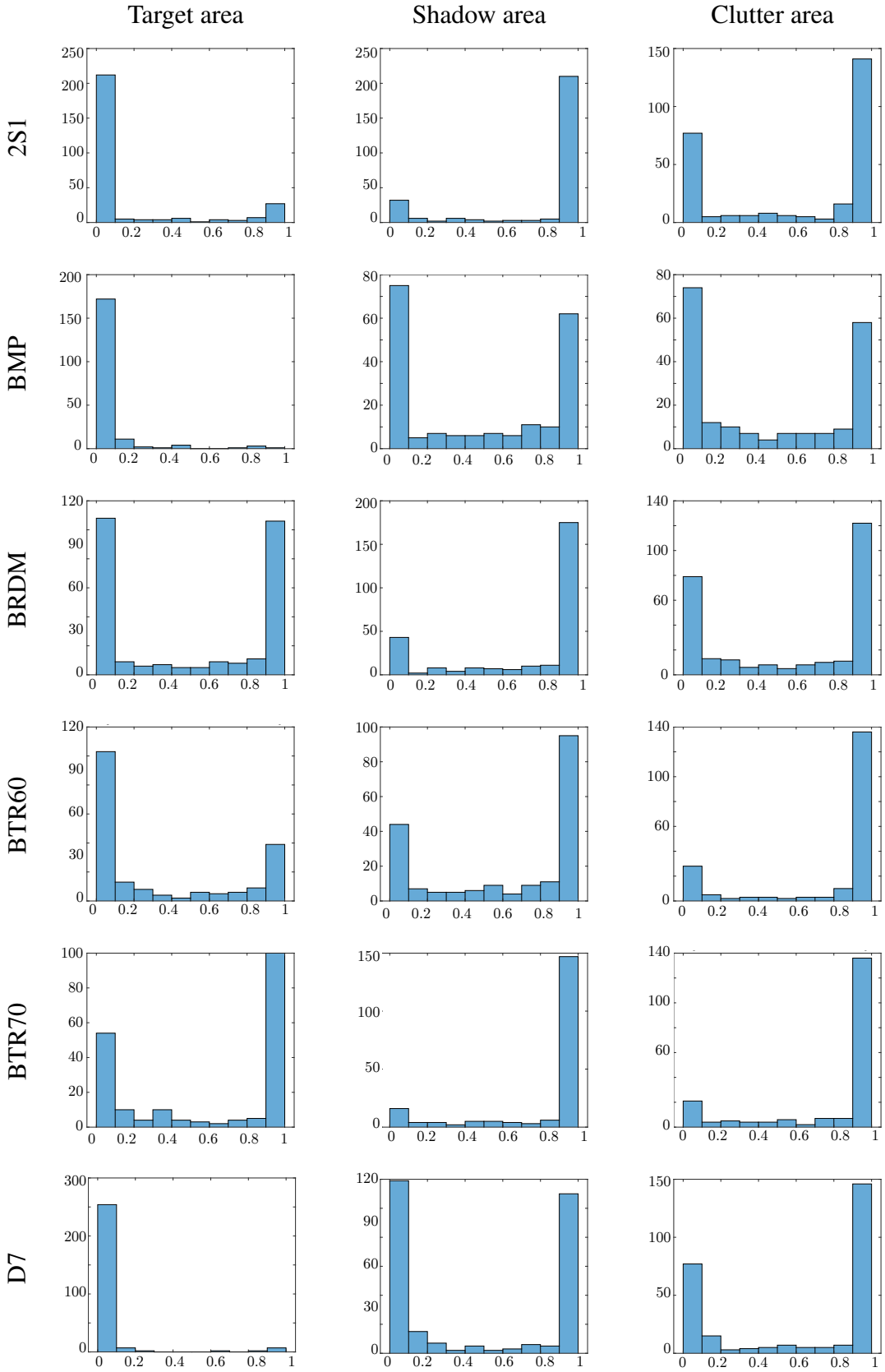


Fig. C.1: Histograms of the most critical features (minimal intensity in the occlusion map) in each image per target per area of interest

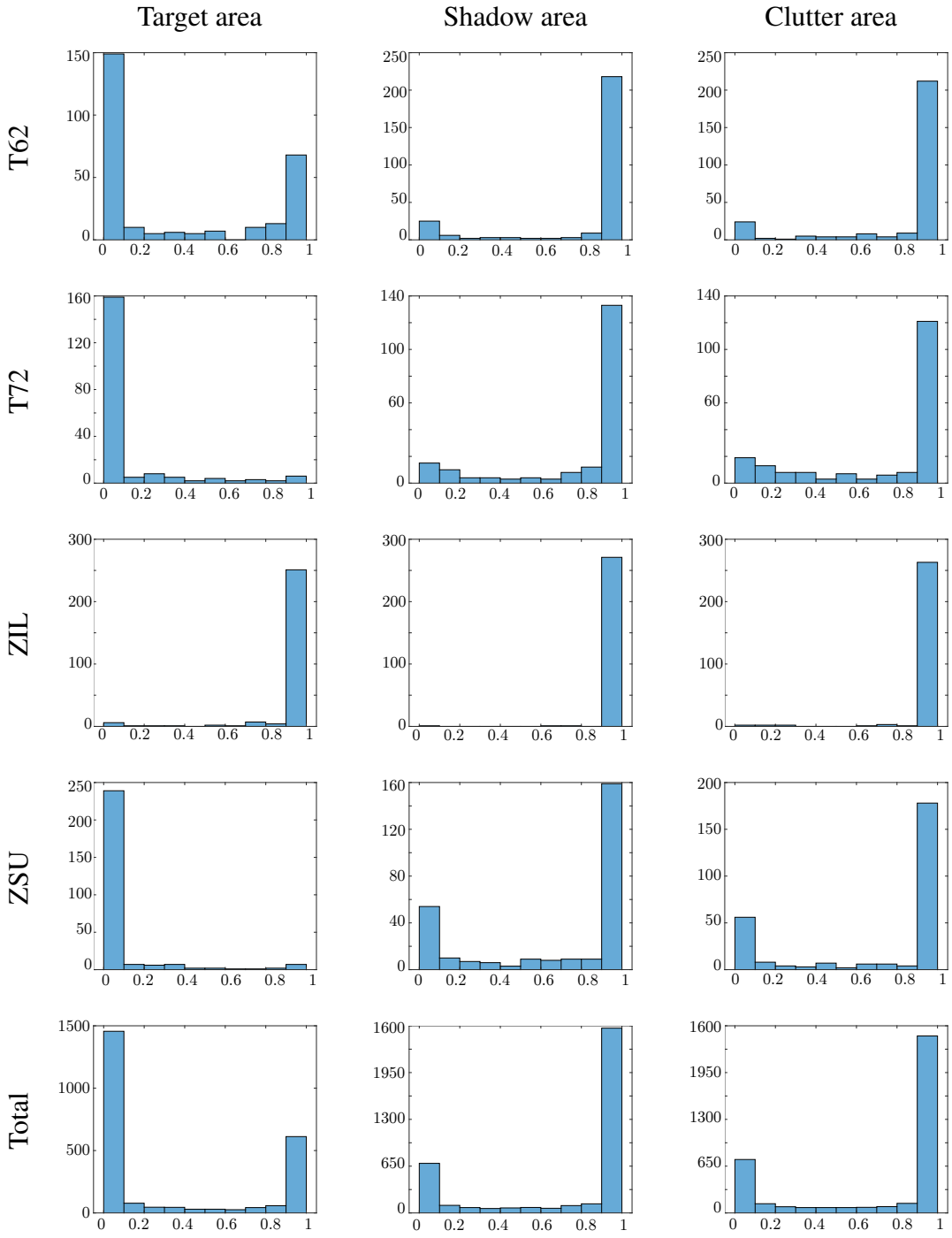


Fig. C.1: Histograms of the most critical features (minimal intensity in the occlusion map) in each image per target per area of interest

Bibliography

[1] Cédric Villani, Yann Bonnet, Charly Berthet, François Levin, Marc Schoenauer, Anne Charlotte Cornut and Bertrand Rondepierre. *Donner un sens à l'intelligence artificielle: pour une stratégie nationale et européenne [Giving meaning to artificial intelligence: For a national and european strategy]*. Conseil national du numérique, 2018.

[2] Titus J Brinker, Achim Hekler, Alexander H Enk, Joachim Klode, Axel Hauschild, Carola Berking, Bastian Schilling, Sebastian Haferkamp, Dirk Schadendorf, Tim Holland-Letz et al. "Deep learning outperformed 136 of 157 dermatologists in a head-to-head dermoscopic melanoma image classification task". In: *European Journal of Cancer* 113 (2019), pp. 47–54.

[3] Fourtinon Luc, Balleri Alessio, Quere Yves, Person Christian, Martin-Guennou Annaig, Rius Eric, Lesueur Guillaume and Merlet Thomas. "Beampattern and Polarisation Synthesis of 3D Rf-Seeker Antenna Arrays". In: *2016 Sensor Signal Processing for Defence (SSPD)*. Sept. 2016, pp. 1–5.

[4] Robert M O'Donnell. "Radar Systems Engineering Lecture 9 Antennas". In: *IEEE New Hampshire Section* (2010).

[5] Carole Belloni, Alessio Balleri, Nabil Aouf, Thomas Merlet and Jean-Marc Le Caillec. "SAR image dataset of military ground targets with multiple poses for ATR". In: *Target and Background Signatures III*. Vol. 10432. International Society for Optics and Photonics. Warsaw, Sept. 2017, 104320N.

[6] Carole Belloni, Alessio Balleri, Nabil Aouf, Thomas Merlet and Jean-Marc Le Caillec. *Cranfield Online Research Data, MGTD database*. URL: https://doi.org/10.17862/cranfield.rd.7240742.

[7] Carole Belloni, Nabil Aouf, Jean-Marc Le Caillec and Thomas Merlet. "Comparison of Descriptors for SAR ATR". In: *2019 IEEE Radar Conference (RadarConf19)*. IEEE. Boston, Apr. 2019.

[8] Carole Belloni, Nabil Aouf, Thomas Merlet and Jean-Marc Le Caillec. "SAR image segmentation with GMMs". In: *International Conference on Radar Systems (Radar 2017)*. IET. Belfast, Oct. 2017.

[9] Carole Belloni, Nabil Aouf, Jean-Marc Le Caillec and Thomas Merlet. "SAR Specific Noise Based Data Augmentation for Deep Learning". In: *2019 IEEE International Radar Conference*. IEEE. Toulon, Sept. 2019.

[10] Carole Belloni, Nabil Aouf, Alessio Balleri, Jean-Marc Le Caillec and Thomas Merlet. "Pose-informed deep learning method for SAR ATR". In: *IET Radar, Sonar & Navigation* (2020).

[11] Carole Belloni, Nabil Aouf, Alessio Balleri, Jean-Marc Le Caillec and Thomas Merlet. "Explainability of Deep SAR ATR Through Feature Analysis". In: *IEEE transactions on aerospace and electronic systems* (2020). Submitted. Accepted with minor revision.

[12] *Sensor Data Management System website, MSTAR database*. URL: https://www.sdms.afrl.af.mil/index.php?collection=mstar.

[13] Rolf Schumacher and Kh Rosenbach. "ATR of battlefield targets by SAR classification results using the public MSTAR dataset compared with a dataset by QinetiQ UK". In: *RTO SET Symp. Target Identification and Recognition Using RF Systems*. Citeseer. 2004.

[14] LeRoy A Gorham and Linda J Moore. "SAR image formation toolbox for MATLAB". In: *Algorithms for Synthetic Aperture Radar Imagery XVII*. Vol. 7699. International Society for Optics and Photonics. 2010, p. 769906.

[15] Marco Martorella, Elisa Giusti, Fabrizio Berizzi, Alessio Bacci and Enzo Dalle Mese. "ISAR based technique for refocusing non-cooperative targets in SAR images". In: *IET Radar, Sonar & Navigation* 6.5 (2012), pp. 332–340.

[16] Dean L Mensa. *High resolution radar imaging*. Artech House Dedham, MA, 1981.

[17] Leslie M Novak, Gregory J Owirka and Allison L Weaver. "Automatic target recognition using enhanced resolution SAR data". In: *IEEE Transactions on Aerospace and Electronic systems* 35.1 (1999), pp. 157–175.

[18] Timothy D Ross, Steven W Worrell, Vincent J Velten, John C Mossing and Michael Lee Bryant. "Standard SAR ATR evaluation experiments using the MSTAR public release data set". In: *Algorithms for Synthetic Aperture Radar Imagery V*. Vol. 3370. International Society for Optics and Photonics. 1998, pp. 566–574.

[19] John C Mossing and Timothy D Ross. "Evaluation of SAR ATR algorithm performance sensitivity to MSTAR extended operating conditions". In: *Algorithms for Synthetic Aperture Radar Imagery V*. Vol. 3370. International Society for Optics and Photonics. 1998, pp. 554–566.

[20] Jifang Pei, Yulin Huang, Weibo Huo, Yin Zhang, Jianyu Yang and Tat-Soon Yeo. "SAR Automatic Target Recognition Based on Multiview Deep Learning Framework". In: *IEEE Transactions on Geoscience and Remote Sensing* 56.4 (2018), pp. 2196–2210.

[21] Sizhe Chen, Haipeng Wang, Feng Xu and Ya-Qiu Jin. "Target classification using the deep convolutional networks for SAR images". In: *IEEE Transactions on Geoscience and Remote Sensing* 54.8 (2016), pp. 4806–4817.

[22] Jong-Il Park and Kyung-Tae Kim. "Modified polar mapping classifier for SAR automatic target recognition". In: *IEEE Transactions on Aerospace and Electronic Systems* 50.2 (2014), pp. 1092–1107.

[23] Ganggang Dong, Gangyao Kuang, Na Wang, Lingjun Zhao and Jun Lu. "SAR target recognition via joint sparse representation of monogenic signal". In: *IEEE Journal of selected topics in applied earth observations and remote sensing* 8.7 (2015), pp. 3316–3328.

[24] Rob J Dekker. "Texture analysis and classification of ERS SAR images for map updating of urban areas in the Netherlands". In: *IEEE Transactions on Geoscience and Remote Sensing* 41.9 (2003), pp. 1950–1958.

[25] Matijs Heiligers and Albert Huizing. "On the importance of visual explanation and segmentation for SAR ATR using deep learning". In: *2018 IEEE Radar Conference (RadarConf18)*. IEEE. 2018, pp. 0394–0399.

[26] Eric R Keydel, Shung W Lee and John T Moore. "MSTAR extended operating conditions: A tutorial". In: *Aerospace/Defense Sensing and Controls*. International Society for Optics and Photonics. 1996, pp. 228–242.

[27] David G Lowe. "Distinctive image features from scale-invariant keypoints". In: *International journal of computer vision* 60.2 (2004), pp. 91–110.

[28] Herbert Bay, Tinne Tuytelaars and Luc Van Gool. "SURF: Speeded up robust features". In: *European conference on computer vision*. Springer. 2006, pp. 404–417.

[29] Ethan Rublee, Vincent Rabaud, Kurt Konolige and Gary Bradski. "ORB: An efficient alternative to SIFT or SURF". In: *2011 International conference on computer vision*. IEEE. 2011, pp. 2564–2571.

[30] Alexandre Alahi, Raphael Ortiz and Pierre Vandergheynst. "FREAK: Fast retina keypoint". In: *Computer vision and pattern recognition (CVPR), 2012 IEEE conference on*. Ieee. 2012, pp. 510–517.

[31] Stefan Leutenegger, Margarita Chli and Roland Y Siegwart. "BRISK: Binary robust invariant scalable keypoints". In: *2011 International conference on computer vision*. IEEE. 2011, pp. 2548–2555.

[32] David G Lowe. "Object recognition from local scale-invariant features". In: *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*. Vol. 2. Ieee. 1999, pp. 1150–1157.

[33] Edward Rosten and Tom Drummond. "Machine learning for high-speed corner detection". In: *European conference on computer vision*. Springer. 2006, pp. 430–443.

[34] Elmar Mair, Gregory D Hager, Darius Burschka, Michael Suppa and Gerhard Hirzinger. "Adaptive and generic corner detection based on the accelerated segment test". In: *European conference on Computer vision*. Springer. 2010, pp. 183–196.

[35] Martin A Fischler and Robert C Bolles. "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography". In: *Communications of the ACM* 24.6 (1981), pp. 381–395.

[36] Yann LeCun, Léon Bottou, Yoshua Bengio, Patrick Haffner et al. "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324.

[37] Nitish Srivastava, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever and Ruslan Salakhutdinov. "Dropout: a simple way to prevent neural networks from overfitting." In: *Journal of machine learning research* 15.1 (2014), pp. 1929–1958.

[38] Yin Chen, Erik Blasch, Huimin Chen, Tao Qian and Genshe Chen. "Experimental feature-based SAR ATR performance evaluation under different operational conditions". In: *SPIE Defense and Security Symposium*. International Society for Optics and Photonics. 2008, 69680F–69680F.

[39] Yulin Huang, Jifang Pei, Jianyu Yang, Bing Wang and Xian Liu. "Neighborhood geometric center scaling embedding for SAR ATR". In: *IEEE Transactions on Aerospace and Electronic Systems* 50.1 (2014), pp. 180–192.

[40] Yang He, Si-Yuan He, Yun-Hua Zhang, Gong-Jian Wen, Ding-Feng Yu and Guo-Qiang Zhu. "A forward approach to establish parametric scattering center models for known complex radar targets applied to SAR ATR". In: *IEEE Transactions on Antennas and Propagation* 62.12 (2014), pp. 6192–6205.

[41] Panqu Wang, Pengfei Chen, Ye Yuan, Ding Liu, Zehua Huang, Xiaodi Hou and Garrison Cottrell. "Understanding convolution for semantic segmentation". In: *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE. 2018, pp. 1451–1460.

[42] Odysseas Kechagias-Stamatis, Nabil Aouf, David Nam and Carole Belloni. "Automatic X-ray image segmentation and clustering for threat detection". In: *Target and Background Signatures III*. Vol. 10432. International Society for Optics and Photonics. Warsaw, Sept. 2017, 104320O.

[43] Roger Fjortoft, Armand Lopes, Philippe Marthon and Eliane Cubero-Castan. "An optimal multiedge detector for SAR image segmentation". In: *IEEE Transactions on Geoscience and Remote Sensing* 36.3 (1998), pp. 793–802.

[44] David Malmgren-Hansen, Morten Nobel-J et al. "Convolutional neural networks for SAR image segmentation". In: *Signal Processing and Information Technology (ISSPIT), 2015 IEEE International Symposium on*. IEEE. 2015, pp. 231–236.

[45] Zoran Zivkovic. "Improved adaptive Gaussian mixture model for background subtraction". In: *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*. Vol. 2. IEEE. 2004, pp. 28–31.

[46] Shuhratchon Ochilov and David A Clausi. "Operational SAR sea-ice image classification". In: *IEEE Transactions on Geoscience and Remote Sensing* 50.11 (2012), pp. 4397–4408.

[47] Mehdi Amoon and Gholam-Ali Rezai-Rad. "Automatic target recognition of synthetic aperture radar (SAR) images based on optimal selection of Zernike moments features". In: *IET Computer Vision* 8.2 (2013), pp. 77–85.

[48] Anupam Agrawal, P Mangalraj and Mukul Anand Bisherwal. "Target detection in SAR images using SIFT". In: *Signal Processing and Information Technology (ISSPIT), 2015 IEEE International Symposium on*. IEEE. 2015, pp. 90–94.

[49] Anna Bosch, Andrew Zisserman and Xavier Muñoz. "Scene classification via pLSA". In: *European conference on computer vision*. Springer. 2006, pp. 517–530.

[50] Jianchao Yang, Kai Yu, Yihong Gong and Thomas Huang. "Linear spatial pyramid matching using sparse coding for image classification". In: *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE. 2009, pp. 1794–1801.

[51] Gregory J Power and Robert A Weisenseel. "ATR subsystem performance measures using manual segmentation of SAR target chips". In: *AeroSense'99*. International Society for Optics and Photonics. 1999, pp. 685–692.

[52] Moritz Menze, Christian Heipke and Andreas Geiger. "Object Scene Flow". In: *ISPRS Journal of Photogrammetry and Remote Sensing (JPRS)* (2018).

[53] Moritz Menze, Christian Heipke and Andreas Geiger. "Joint 3D Estimation of Vehicles and Scene Flow". In: *ISPRS Workshop on Image Sequence Analysis (ISA)*. 2015.

[54] *Results of the SARBake segmentation on the MSTAR SOC10*. URL: `http://www2.compute.dtu.dk/~dmal/project.html`.

[55] Yijun Sun, Zhipeng Liu, Sinisa Todorovic and Jian Li. "Adaptive boosting for SAR automatic target recognition". In: *IEEE Transactions on Aerospace and Electronic Systems* 43.1 (2007).

[56] Liviu I Voicu, Ronald Patton and Harley R Myler. "Multicriterion vehicle pose estimation for SAR ATR". In: *AeroSense'99*. International Society for Optics and Photonics. 1999, pp. 497–506.

[57] Lance M Kaplan. "Analysis of multiplicative speckle models for template-based SAR ATR". In: *IEEE Transactions on Aerospace and Electronic Systems* 37.4 (2001), pp. 1424–1432.

[58] Andrew Rabinovich, Andrea Vedaldi and Serge J Belongie. *Does image segmentation improve object categorization?* Department of Computer Science and Engineering, University of California ..., 2007.

[59] Xinzheng Zhang, Jianhong Qin and Guojun Li. "SAR target classification using Bayesian compressive sensing with scattering centers features". In: *Progress In Electromagnetics Research* 136 (2013), pp. 385–407.

[60] Mehdi Amoon and Gholam-Ali Rezai-Rad. "Automatic target recognition of synthetic aperture radar (SAR) images based on optimal selection of Zernike moments features". In: *IET Computer Vision* 8.2 (2014), pp. 77–85.

[61] John R Hershey and Peder A Olsen. "Approximating the Kullback Leibler divergence between Gaussian mixture models". In: *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07*. Vol. 4. IEEE. 2007, pp. IV–317.

[62] Chris Harris and Mike Stephens. "A combined corner and edge detector." In: *Alvey vision conference*. Vol. 15. Citeseer. 1988, p. 50.

[63] Richard O Duda and Peter E Hart. *Use of the Hough transformation to detect lines and curves in pictures*. Tech. rep. SRI INTERNATIONAL MENLO PARK CA ARTIFICIAL INTELLIGENCE CENTER, 1971.

[64] Lance M Kaplan and Romain Murenzi. "Pose estimation of SAR imagery using the two dimensional continuous wavelet transform". In: *Pattern recognition letters* 24.14 (2003), pp. 2269–2280.

[65] Ning Xin, Guo-hong Wang and Jing Zhang. "A synthetical pose estimation of sar imagery using hough transform and 2-d continuous wavelet transform". In: *Radar, 2006. CIE'06. International Conference on*. IEEE. 2006, pp. 1–4.

[66] Joseph A O'Sullivan, Michael D DeVore, Vikas Kedia and Michael I Miller. "SAR ATR performance using a conditionally Gaussian model". In: *IEEE Transactions on Aerospace and Electronic Systems* 37.1 (2001), pp. 91–108.

[67] Chao Yuan and David Casasent. "A new SVM for distorted SAR object classification". In: *Optical Pattern Recognition XVI*. Vol. 5816. International Society for Optics and Photonics. 2005, pp. 10–23.

[68] Qun Zhao and Jose C Principe. "Support vector machines for SAR automatic target recognition". In: *IEEE Transactions on Aerospace and Electronic Systems* 37.2 (2001), pp. 643–654.

[69] Rohit Patnaik and David Casasent. "MINACE filter classification algorithms for ATR using MSTAR data". In: *Automatic Target Recognition XV*. Vol. 5807. International Society for Optics and Photonics. 2005, pp. 100–112.

[70] Xin Yu, Yukuan Li and LC Jiao. "SAR automatic target recognition based on classifiers fusion". In: *2011 International Workshop on Multi-Platform/Multi-Sensor Remote Sensing and Mapping*. IEEE. 2011, pp. 1–5.

[71] Ying Wang, Ping Han, Xiaoguang Lu, Renbiao Wu and Jingxiong Huang. "The performance comparison of Adaboost and SVM applied to SAR ATR". In: *2006 CIE International Conference on Radar*. IEEE. 2006, pp. 1–4.

[72] Haibo Song, Kefeng Ji, Yunshu Zhang, Xiangwei Xing and Huanxin Zou. "Sparse representation-based SAR image target classification on the 10-class MSTAR data set". In: *Applied Sciences* 6.1 (2016), p. 26.

[73] David AE Morgan. "Deep convolutional neural networks for ATR from SAR imagery". In: *Proceedings of the Algorithms for Synthetic Aperture Radar Imagery XXII, Baltimore, MD, USA* 23 (2015), 94750F.

[74] Andrew Profeta, Andres Rodriguez and H Scott Clouse. “Convolutional neural networks for synthetic aperture radar classification”. In: *Algorithms for Synthetic Aperture Radar Imagery XXIII*. Vol. 9843. International Society for Optics and Photonics. 2016, p. 98430M.

[75] Simon Wagner. “Combination of convolutional feature extraction and support vector machines for radar ATR”. In: *17th International Conference on Information Fusion (FUSION)*. IEEE. 2014, pp. 1–6.

[76] Odysseas Kechagias-Stamatis, Nabil Aouf and Carole Belloni. “SAR Automatic Target Recognition based on Convolutional Neural Networks”. In: *International Conference on Radar Systems (Radar 2017)*. IET. Belfast, Oct. 2017.

[77] Maha Al Mufti, Esra Al Hadhrami, Bilal Taha and Naoufel Werghi. “SAR Automatic Target Recognition Using Transfer Learning Approach”. In: *2018 International Conference on Intelligent Autonomous Systems (ICoIAS)*. IEEE. 2018, pp. 1–4.

[78] Roland Memisevic, Christopher Zach, Marc Pollefeys and Geoffrey E Hinton. “Gated softmax classification”. In: *Advances in neural information processing systems*. 2010, pp. 1603–1611.

[79] Jia Cheng Ni and Yue Lei Xu. “SAR automatic target recognition based on a visual cortical system”. In: *Image and Signal Processing (CISP), 2013 6th International Congress on*. Vol. 2. IEEE. 2013, pp. 778–782.

[80] Sizhe Chen and Haipeng Wang. “SAR target recognition based on deep learning”. In: *Data Science and Advanced Analytics (DSAA), 2014 International Conference on*. IEEE. 2014, pp. 541–547.

[81] Xuan Li, Chunsheng Li, Pengbo Wang, Zhirong Men and Huaping Xu. “SAR ATR based on dividing CNN into CAE and SNN”. In: *Synthetic Aperture Radar (APSAR), 2015 IEEE 5th Asia-Pacific Conference on*. IEEE. 2015, pp. 676–679.

[82] Alec Radford, Luke Metz and Soumith Chintala. “Unsupervised representation learning with deep convolutional generative adversarial networks”. In: *arXiv preprint arXiv:1511.06434* (2015).

[83] Fei Gao, Yue Yang, Jun Wang, Jinping Sun, Erfu Yang and Huiyu Zhou. “A deep convolutional generative adversarial networks (DCGANs)-based semi-supervised method for object recognition in synthetic aperture radar (SAR) images”. In: *Remote Sensing* 10.6 (2018), p. 846.

[84] Benjamin Lewis, Jennifer Liu and Amy Wong. “Generative adversarial networks for SAR image realism”. In: *Algorithms for Synthetic Aperture Radar Imagery XXV*. Vol. 10647. International Society for Optics and Photonics. 2018, p. 1064709.

[85] Jiayi Guo, Bin Lei, Chibiao Ding and Yueting Zhang. “Synthetic aperture radar image synthesis by using generative adversarial nets”. In: *IEEE Geoscience and Remote Sensing Letters* 14.7 (2017), pp. 1111–1115.

[86] Dimitrios Korkinof, Tobias Rijken, Michael O’Neill, Joseph Yearsley, Hugh Harvey and Ben Glocker. “High-resolution mammogram synthesis using progressive generative adversarial networks”. In: *arXiv preprint arXiv:1807.03401* (2018).

[87] David Bau, Jun-Yan Zhu, Hendrik Strobelt, Bolei Zhou, Joshua B Tenenbaum, William T Freeman and Antonio Torralba. “GAN Dissection: Visualizing and Understanding Generative Adversarial Networks”. In: *arXiv preprint arXiv:1811.10597* (2018).

[88] David Malmgren-Hansen, Anders Kusk, Jørgen Dall, Allan Aasbjerg Nielsen, Rasmus Engholm and Henning Skriver. “Improving SAR automatic target recognition models with transfer learning from simulated data”. In: *IEEE Geoscience and Remote Sensing Letters* 14.9 (2017), pp. 1484–1488.

[89] Miriam Cha, Arjun Majumdar, HT Kung and Jarred Barber. “Improving Sar Automatic Target Recognition Using Simulated Images Under Deep Residual Re-

finements". In: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2018, pp. 2606–2610.

[90] Kangning Du, Yunkai Deng, Robert Wang, Tuan Zhao and Ning Li. "SAR ATR based on displacement-and rotation-insensitive CNN". In: *Remote Sensing Letters* 7.9 (2016), pp. 895–904.

[91] Jun Ding, Bo Chen, Hongwei Liu and Mengyuan Huang. "Convolutional neural network with data augmentation for SAR target recognition". In: *IEEE Geoscience and remote sensing letters* 13.3 (2016), pp. 364–368.

[92] Odysseas Kechagias-Stamatis and Nabil Aouf. "Fusing Deep Learning and Sparse Coding for SAR ATR". In: *IEEE Transactions on Aerospace and Electronic Systems* (2018).

[93] Miao Kang, Kefeng Ji, Xiangguang Leng, Xiangwei Xing and Huanxin Zou. "Synthetic aperture radar target recognition with feature fusion based on a stacked autoencoder". In: *Sensors* 17.1 (2017), p. 192.

[94] Fan Zhang, Chen Hu, Qiang Yin, Wei Li, Hengchao Li and Wen Hong. "SAR Target Recognition Using the Multi-aspect-aware Bidirectional LSTM Recurrent Neural Networks". In: *arXiv preprint arXiv:1707.09875* (2017).

[95] Pengfei Zhao, Kai Liu, Hao Zou and Xiantong Zhen. "Multi-stream convolutional neural network for SAR automatic target recognition". In: *Remote Sensing* 10.9 (2018), p. 1473.

[96] Yinan Yang, Yuxia Qiu and Chao Lu. "Automatic target classification—experiments on the MSTAR SAR images". In: *Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing, 2005 and First ACIS International Workshop on Self-Assembling Wireless Networks. SNPD/SAWN 2005. Sixth International Conference on*. IEEE. 2005, pp. 2–7.

[97] Yu Zhong and Gil J. Ettinger. "Enlightening Deep Neural Networks with Knowledge of Confounding Factors". In: *CoRR* abs/1607.02397 (2016). arXiv: 1607.02397. URL: http://arxiv.org/abs/1607.02397.

[98] Andrew W Learn. *Target pose estimation from radar data using adaptive networks*. Tech. rep. AIR FORCE INST OF TECH WRIGHT-PATTERSONAFB OH SCHOOL OF ENGINEERING, 1999.

[99] Michael D DeVore, Aaron D Lanterman and Joseph A O'Sullivan. "ATR performance of a Rician model for SAR images". In: *Automatic Target Recognition X*. Vol. 4050. International Society for Optics and Photonics. 2000, pp. 34–46.

[100] Lijiang Peng, Xiaohua Liu, Ming Liu, Liquan Dong, Mei Hui and Yuejin Zhao. "SAR target recognition and posture estimation using spatial pyramid pooling within CNN". In: *2017 International Conference on Optical Instruments and Technology: Optoelectronic Imaging/Spectroscopy and Signal Processing Technology*. Vol. 10620. International Society for Optics and Photonics. 2018, 106200W.

[101] Jose C Principe, Dongxin Xu and John W Fisher. "Pose estimation in SAR using an information theoretic criterion". In: *Algorithms for Synthetic Aperture Radar Imagery V*. Vol. 3370. International Society for Optics and Photonics. 1998, pp. 218–230.

[102] Qun Zhao, Dongxin Xu and J Principe. "Pose estimation of SAR automatic target recognition". In: *Proceedings of Image Understanding Workshop*. Vol. 11. 1998.

[103] Alex Krizhevsky, Ilya Sutskever and Geoffrey E Hinton. "Imagenet classification with deep convolutional neural networks". In: *Advances in neural information processing systems*. 2012, pp. 1097–1105.

[104] Alexander Toshev and Christian Szegedy. "Deeppose: Human pose estimation via deep neural networks". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2014, pp. 1653–1660.

[105] TodericiG KarpathyA et al. “Large-scale video classification with convolutional neural networks”. In: *Computer Vision and Pattern Recognition (CVPR), IEEE* 2014 (), p. 1725.

[106] Ross Girshick, Jeff Donahue, Trevor Darrell and Jitendra Malik. “Rich feature hierarchies for accurate object detection and semantic segmentation”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2014, pp. 580–587.

[107] Kaiming He, Xiangyu Zhang, Shaoqing Ren and Jian Sun. “Deep residual learning for image recognition”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[108] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke and Andrew Rabinovich. “Going deeper with convolutions”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 1–9.

[109] Karen Simonyan and Andrew Zisserman. “Very deep convolutional networks for large-scale image recognition”. In: *arXiv preprint arXiv:1409.1556* (2014).

[110] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li and L. Fei-Fei. “ImageNet: A Large-Scale Hierarchical Image Database”. In: *CVPR09*. 2009.

[111] Yoshua Bengio. “Deep learning of representations for unsupervised and transfer learning”. In: *Proceedings of ICML Workshop on Unsupervised and Transfer Learning*. 2012, pp. 17–36.

[112] Xavier Glorot and Yoshua Bengio. “Understanding the difficulty of training deep feedforward neural networks”. In: *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. 2010, pp. 249–256.

[113] Hao Su, Charles R Qi, Yangyan Li and Leonidas J Guibas. “Render for cnn: Viewpoint estimation in images using cnns trained with rendered 3d model views”. In:

Proceedings of the IEEE International Conference on Computer Vision. 2015, pp. 2686–2694.

[114] Hidetoshi Furukawa. "Deep Learning for Target Classification from SAR Imagery: Data Augmentation and Translation Invariance". In: *CoRR* abs/1708.07920 (2017). arXiv: 1708.07920. URL: http://arxiv.org/abs/1708.07920.

[115] Haipeng Wang, Sizhe Chen, Feng Xu and Ya-Qiu Jin. "Application of deep-learning algorithms to MSTAR data". In: *Geoscience and Remote Sensing Symposium (IGARSS), 2015 IEEE International*. IEEE. 2015, pp. 3743–3745.

[116] Simon A Wagner. "SAR ATR by a combination of convolutional neural network and support vector machines". In: *IEEE Transactions on Aerospace and Electronic Systems* 52.6 (2016), pp. 2861–2872.

[117] Jong-Sen Lee, L Jurkevich, P Dewaele, Pl Wambacq and A Oosterlinck. "Speckle filtering of synthetic aperture radar images: A review". In: *Remote Sensing Reviews* 8.4 (1994), pp. 313–340.

[118] Satoshi Kouya and Susumu Miwa. "Analysis of K-distributed clutter parameters for various geographical features". In: *Electronics and Communications in Japan (Part I: Communications)* 82.2 (1999), pp. 57–65.

[119] L James Marier. "Correlated K-distributed clutter generation for radar detection and track". In: *IEEE Transactions on aerospace and electronic systems* 31.2 (1995), pp. 568–580.

[120] Fulvio Gini, Maria S Greco and F Lombardini. *Statistical Analysis of High Resolution SAR Ground Clutter Data*. Tech. rep. Pisa University, Dipartimento di ingneria dell' informazione, 2005.

[121] Han Gao and Jingwen Li. "Detection and tracking of a moving target using SAR images with the particle filter-based track-before-detect algorithm". In: *Sensors* 14.6 (2014), pp. 10829–10845.

[122] Lance M Kaplan. “Improved SAR target detection via extended fractal features”. In: *IEEE Transactions on Aerospace and Electronic Systems* 37.2 (2001), pp. 436–451.

[123] Jin Min Kuo and K-S Chen. “The application of wavelets correlator for ship wake detection in SAR images”. In: *IEEE Transactions on Geoscience and Remote Sensing* 41.6 (2003), pp. 1506–1511.

[124] Bir Bhanu and Grinnell Jones. “Recognizing occluded MSTAR targets”. In: *Algorithms for Synthetic Aperture Radar Imagery VII*. Vol. 4053. International Society for Optics and Photonics. 2000, pp. 361–370.

[125] Baiyuan Ding, Gongjian Wen, Conghui Ma and Xiaoliang Yang. “An Efficient and Robust Framework for SAR Target Recognition by Hierarchically Fusing Global and Local Features”. In: *IEEE Transactions on Image Processing* 27.12 (2018), pp. 5983–5995.

[126] Alin Achim, Panagiotis Tsakalides and Anastasios Bezerianos. “SAR image denoising via Bayesian wavelet shrinkage based on heavy-tailed modeling”. In: *IEEE Transactions on Geoscience and Remote Sensing* 41.8 (2003), pp. 1773–1784.

[127] Langis Gagnon and Alexandre Jouan. “Speckle filtering of SAR images: a comparative study between complex-wavelet-based and standard filters”. In: *Wavelet Applications in Signal and Image Processing V*. Vol. 3169. International Society for Optics and Photonics. 1997, pp. 80–92.

[128] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio and Pierre-Antoine Manzagol. “Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion”. In: *Journal of machine learning research* 11.Dec (2010), pp. 3371–3408.

[129] Xue Feng, Yaodong Zhang and James Glass. “Speech feature denoising and dereverberation via deep autoencoders for noisy reverberant speech recognition”. In:

2014 IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE. 2014, pp. 1759–1763.

[130] Michael Wilmanski, Chris Kreucher and Jim Lauer. "Modern approaches in deep learning for SAR ATR". In: *Algorithms for synthetic aperture radar imagery XXIII*. Vol. 9843. International Society for Optics and Photonics. 2016, 98430N.

[131] LM Novak. "State-of-the-art of SAR automatic target recognition". In: *Radar Conference, 2000. The Record of the IEEE 2000 International*. IEEE. 2000, pp. 836–843.

[132] Yingying Jiang, Xiangyu Zhu, Xiaobing Wang, Shuli Yang, Wei Li, Hua Wang, Pei Fu and Zhenbo Luo. "R2CNN: rotational region CNN for orientation robust scene text detection". In: *arXiv preprint arXiv:1706.09579* (2017).

[133] Yicheng Jiang, Xiaohui Zhao, Yun Zhang, Bin Hu and Yuan Zhuang. "Pose estimation based on exploration of geometrical information in SAR images". In: *2016 IEEE Radar Conference (RadarConf)*. IEEE. 2016, pp. 1–4.

[134] David Malmgren-Hansen, Rasmus Engholm and Morten Ostergaard Pedersen. "Training convolutional neural networks for translational invariance on SAR ATR". In: *Proceedings of EUSAR 2016: 11th European Conference on Synthetic Aperture Radar*. VDE. 2016, pp. 1–4.

[135] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, Dhruv Batra et al. "Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization." In: *ICCV*. 2017, pp. 618–626.

[136] Matthew D Zeiler and Rob Fergus. "Visualizing and understanding convolutional networks". In: *European conference on computer vision*. Springer. 2014, pp. 818–833.

[137] Ivet Rafegas and María Vanrell. "Understanding learned CNN features through Filter Decoding with Substitution". In: *CoRR* abs/1511.05084 (2015). arXiv: 1511.05084. URL: http://arxiv.org/abs/1511.05084.

[138] Dario Garcia-Gasulla, Ferran Parés, Armand Vilalta, Jonathan Moreno, Eduard Ayguadé, Jesús Labarta, Ulises Cortés and Toyotaro Suzumura. "On the Behavior of Convolutional Nets for Feature Extraction". In: *CoRR* abs/1703.01127 (2017). arXiv: 1703.01127. URL: http://arxiv.org/abs/1703.01127.

[139] Pang Wei Koh and Percy Liang. "Understanding black-box predictions via influence functions". In: *arXiv preprint arXiv:1703.04730* (2017).

[140] Scott Papson and Ram M Narayanan. "Classification via the shadow region in SAR imagery". In: *IEEE Transactions on Aerospace and Electronic Systems* 48.2 (2012), pp. 969–980.

[141] Jingjing Cui, Jon Gudnason and Mike Brookes. "Automatic recognition of MSTAR Targets using radar shadow and superresolution features". In: *Acoustics, Speech, and Signal Processing, 2005. Proceedings.(ICASSP'05). IEEE International Conference on*. Vol. 5. IEEE. 2005, pp. v–589.

[142] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox and Martin Riedmiller. "Striving for simplicity: The all convolutional net". In: *arXiv preprint arXiv:1412.6806* (2014).

[143] Aravindh Mahendran and Andrea Vedaldi. "Salient deconvolutional networks". In: *European Conference on Computer Vision*. Springer. 2016, pp. 120–135.

[144] Marco Tulio Ribeiro, Sameer Singh and Carlos Guestrin. "Why should i trust you?: Explaining the predictions of any classifier". In: *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. ACM. 2016, pp. 1135–1144.

[145] Xinrui Cui, Dan Wang and Z Jane Wang. "CHIP: Channel-wise Disentangled Interpretation of Deep Convolutional Neural Networks". In: *arXiv preprint arXiv:1902.02497* (2019).

[146] Pierfrancesco Lombardo, Massimo Sciotti and Lance M Kaplan. "SAR prescreening using both target and shadow information". In: *Radar Conference, 2001. Proceedings of the 2001 IEEE*. IEEE. 2001, pp. 147–152.

[147] T Sparr, RE Hansen, HJ Callow and J Groen. "Enhancing target shadows in SAR images". In: *Electronics letters* 43.5 (2007), pp. 69–70.

[148] Nicholas Sandirasegaram and Ryan English. "Comparative analysis of feature extraction (2D FFT and wavelet) and classification (L p metric distances, MLP NN, and HNeT) algorithms for SAR imagery". In: *Algorithms for Synthetic Aperture Radar Imagery XII*. Vol. 5808. International Society for Optics and Photonics. 2005, pp. 314–326.

[149] Huan Ruohong and Yang Ruliang. "SAR target recognition based on MRF and Gabor wavelet feature extraction". In: *IGARSS 2008-2008 IEEE International Geoscience and Remote Sensing Symposium*. Vol. 2. IEEE. 2008, pp. II–907.

[150] Corentin Henry, Seyed Majid Azimi and Nina Merkle. "Road Segmentation in SAR Satellite Images With Deep Fully Convolutional Neural Networks". In: *IEEE Geoscience and Remote Sensing Letters* 99 (2018), pp. 1–5.

Titre : Techniques de Classification par Deep Learning et Descripteurs pour l'Imagerie Radar

Mots clés : RSO, RAC, Deep learning, Classification, Segmentation, Computer vision.

Résumé : Une plateforme autonome en mouvement dotée d'un système radar peut générer des images Radar à Synthèse d'Ouverture (RSO ou SAR). Ces images fournissent des informations stratégiques pour des applications civiles et militaires. Elles peuvent être acquises de jour comme de nuit dans des conditions météorologiques variées. Des algorithmes visant à la Reconnaissance Automatique de Cible (RAC ou ATR) sont alors utiles pour assister voire automatiser la prise de décision. En effet, l'interprétation de ces images peut être complexe, y compris pour un opérateur expérimenté.

La classification d'images du domaine visible génère un intérêt important des chercheurs, en partie grâce à la profusion des données. Par conséquent, des méthodes robustes de classification par descripteurs et deep learning ont été développées pour les images visibles. A l'inverse, une problématique essentielle rencontrée lors du développement d'algorithmes pour la RAC RSO est la rareté des données accessibles au public. Une difficulté supplémentaire est la variabilité des phénomènes physiques lors de l'acquisition radar. Les méthodes de classification des images optiques pourraient être adaptées pour les images RSO.

Une nouvelle base de données d'images RSO Inverse (RSOI ou ISAR) est proposée dans cette thèse. Elle contient des images d'entraînement et de test obtenues dans des configurations variées. Une technique visant à générer des images artificielles supplémentaires est aussi développée. L'objectif est d'améliorer l'efficacité de l'apprentissage des algorithmes de classification nécessitant de nombreuses images d'entraînement, tels que les réseaux de neurones. Cette technique consiste à simuler un bruit SAR réaliste sur les images initiales.

Une segmentation basée sur des Modèles de Mélange de Gaussiennes (MMG ou GMM) est adaptée à des images RSO à polarisation simple. Des descripteurs conçus pour caractériser des images optiques sont utilisés dans le domaine RSO afin de classifier des cibles après segmentation et leurs performances respectives sont comparées.

Une nouvelle architecture de réseau de neurones, appelée pose-informed, est développée. Elle prend en compte les effets de l'orientation de la cible sur son apparence dans les images RSO. Les résultats présentés montrent que cette architecture permet une amélioration significative de la classification par rapport à une architecture standard. Au-delà des performances, un enjeu clé réside dans l'explicativité des méthodes issues du deep learning. Un ensemble d'outils analytiques sont présentés afin faciliter la compréhension du processus de décision du réseau de neurones. Ils permettent, entre autres, l'identification des zones vues comme essentielles à la classification par le réseau de neurones.

UNIVERSITE BRETAGNE LOIRE / MATHSTIC

IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

Title : Deep Learning and Feature-Based Classification Techniques for Radar Imagery

Keywords : SAR, ATR, Deep learning, Classification, Segmentation, Computer vision.

Abstract : Autonomous moving platforms carrying radar systems can synthesise long antenna apertures and generate Synthetic Aperture Radar (SAR) images. SAR images provide strategic information for military and civilian applications and they can be acquired day and night under a wide range of weather conditions. Because the interpretation of SAR images is a common challenge, Automatic Target Recognition (ATR) algorithms can help assist with decision-making when the operator is in the loop or when the platforms are fully autonomous.

One of the main limitations of developing SAR ATR algorithms is the lack of suitable and publicly available data. Optical images classification, instead, has recently attracted significantly more research interest because of the number of potential applications and the profusion of data. As a result, robust feature-based and deep learning classification methods have been developed for optical imaging that could be applied to the SAR domain.

In this thesis, a new Inverse SAR (ISAR) dataset consisting of test and training images acquired under a range of geometrical conditions is presented. In addition, a method is proposed to generate extra synthetic images, by simulating realistic SAR noise on the original images, and increase the training efficiency of classification algorithms that require a wealth of data, such as deep neural networks.

A Gaussian Mixture Model (GMM) segmentation approach is adapted to segment single-polarised SAR images of targets. Features proposed to characterise optical images are transferred to the SAR domain to carry out target classification after segmentation and their respective performance is compared.

A new pose-informed deep learning network architecture, that takes into account the effects of target orientation on target appearance in a SAR image, is proposed. The results presented in this thesis show that the use of this architecture provides a significant performance improvement for almost all datasets used in this work over a baseline network. Understanding the decision-making process of deep networks is another key challenge of deep learning. To address this issue, a new set of analytical tools is proposed that enables the identification, amongst other things, of the location of the algorithm focus points that lead to high level classification performance.