Sea Surface Wind and Wave Parameter Estimation From X-Band Marine Radar Images with Rain Detection and Mitigation

by

© **Xinwei Chen, B.Eng.**

A thesis submitted to the School of
Graduate Studies in partial
fulfillment of the requirements for
the degree of Doctor of Philosophy.

Faculty of Engineering and Applied Science

Memorial University

October 2021

St. John's, Newfoundland and Labrador, Canada

Abstract

In this research, the application of X-band marine radar backscatter images for sea surface wind and wave parameter estimation with rain detection and mitigation is investigated.

In the presence of rain, the rain echoes in the radar image blur the wave signatures and negatively affect estimation accuracy. Hence, in order to improve estimation accuracy, it is meaningful to detect the presence of those rain echoes and mitigate their influence on estimation results. Since rain alters radar backscatter intensity distribution, features are extracted from the normalized histogram of each radar image. Then, a support vector machine (SVM)-based rain detection model is proposed to classify radar images obtained between rainless and rainy conditions. The classification accuracy shows significant improvement compared to the existing threshold-based method. By further observing images obtained under rainy conditions, it is found that many of them are only partially contaminated by rain echoes. Therefore, in order to segment between rain-contaminated regions and those that are less or unaffected by rain, two types of methods are developed based on unsupervised learning techniques and convolutional neural network (CNN), respectively. Specifically, for the unsupervised learning-based method, texture features are first extracted from each pixel and then trained using a self organizing map (SOM)-based clustering model, which is able to conduct pixel-based identification of rain-contaminated regions. As for the CNN-based method, a SegNet-based semantic segmentation CNN is first designed and then trained using images with manually annotated labels. Both shipborne and shore-based marine radar data are used to train and validate the proposed methods and high classification accuracies of around 90% are obtained.

Due to the similarities between how haze affects terrestrial images and how rain affects marine radar images, a type of CNN for image dehazing purposes, i.e., DehazeNet, is applied to rain-contaminated regions in radar images for correcting the influence of rain, which reduces the estimation error of wind direction significantly. Besides, after extracting

histogram and texture features from rain-corrected radar images, a support vector regression (SVR)-based model, which achieves high estimation accuracy, is trained for wind speed estimation. Finally, a convolutional gated recurrent unit (CGRU) network is designed and trained for significant wave height (SWH) estimation. As an end-to-end system, the proposed network is able to generate estimation results directly from radar image sequences by extracting multi-scale spatial and temporal features in radar image sequences automatically. Compared to the classic signal-to-noise (SNR)-based method, the CGRU-based model shows significant improvement in both estimation accuracy (under both rainless and rainy conditions) and computational efficiency.

Acknowledgements

Throughout my four years of PhD study I have received a great deal of support and assistance.

First, I would like to express my sincere gratitude to my supervisor, Dr. Weimin Huang, whose expertise was invaluable in formulating the research questions and methodology. His insightful feedback pushed me to sharpen my thinking and brought my work to a higher level. His recommendation and support also helped me obtain two internship opportunities and win several academic awards during my study. I wish to thank the Faculty of Engineering and Applied Science for offering me the opportunity of taking courses, serving as a teaching assistant, and conducting this research. I also appreciate my supervisory committee members, Dr. Lihong Zhang and Dr. Tariq Iqbal, for their suggestions and comments on my research. In addition, I am grateful to the financial support in the form of the Natural Sciences and Engineering Research Council of Canada (NSERC) Discovery Grants to Dr. Huang (NSERC RGPIN-2017-04508 and RGPAS-2017-507962), and the Mitacs through the Mitacs Accelerate Program (IT21291 and IT20702). I would also like to thank Dr. E. Thornhill at Defence Research and Development Canada (DRDC), as well as Dr. M. Haller and Mr. R. Pittman from Oregon State University for the provision of radar, anemometer, buoy, and rain gauge data. In addition, I would like to thank Compute Canada for providing advanced computing resources.

Next, I would like to thank all the members in our ocean remote sensing research group. The discussions and presentations in group meetings were always inspiring to my research. I also appreciate all my friends I met in St John's for their help in my daily life.

Finally, I would like to thank all my family members. I could not have completed this dissertation without their unconditional support.

Table of contents

List of tables

List of figures

List of symbols

Θ	the azimuth direction (p. 5)
σ_Θ	the average intensity over range for each azimuthal direction (p. 5)
a_0, a_1, a_2	the parameters in the cosine-squared function determined through least-squares fitting (p. 5)
$\bar{\sigma}$	the integration of fitted curve over azimuth (p. 5)
b_0, b_1, b_2, b_3	the parameters in the third-degree polynomial function (p. 5)
w_{spd}	the local wind speed (p. 5)
L_i	the i^{th} intensity level (p. 5)
$r_i(\Theta)$	the i^{th} range distance vector (p. 5)
α_i	the i^{th} conversion rate (p. 6)
β_0, β_1, β_2, β_3	the parameters that relate intensity level and conversion rate (p. 6)
$F_F^3(\vec{k},\omega)$	the 3-D wavenumber-frequency image spectrum (p. 9)
$F_W^3(\vec{k},\omega)$	the 3-D wavenumber-frequency wave spectrum (p. 9)
$T_M(\vec{k})$	the modulation transfer function (p. 9)
H_s	the significant wave height (p. 9)
c_0, c_1	the parameters that relate signal-to-noise ratio to significant wave height (p. 9)
SNR	the signal-to-noise ratio (p. 9)
P_t	the radar transmitter power (p. 17)
G_t	the gain of the transmitting antenna (p. 17)
A_r	the effective aperture of the receiving antenna (p. 17)

$\boldsymbol{\sigma}$	the scattering cross section of the target (p. 17)
R	the range from the radar to the target (p. 17)
$\boldsymbol{\sigma^0}$	the radar cross section per unit surface area (p. 18)
ϕ	the radar incidence angle (p. 18)
w_{dir}	the local upwind direction (p. 18)
B_0	the steady state term accounts for major wind speed effects (p. 18)
B_1	the fundamental term accounts for the difference between backscatter at upwind and downwind (p. 18)
B_2	the second harmonic that accounts for difference between backscatter at upwind and crosswind (p. 18)
$\mathbf{x_i}$	the feature vector (p. 20)
$\mathbf{y_i}$	the classification result of $\mathbf{x_i}$ (p. 20)
l	the number of training samples (p. 20)
$\mathbf{w}$	the vector that is normal to the hyperplane (p. 21)
b	the bias term (p. 21)
$\boldsymbol{\psi(\cdot)}$	the kernel function that projects training samples from the input space to the feature space (p. 21)
ε_i	the slack variable (p. 21)
$J(\mathbf{w})$	the risk function (p. 21)
C	the penalty term (p. 21)
$L(\mathbf{w}, b, \epsilon, \alpha, r)$	the Lagrange function (p. 22)
$\mathbf{r_i}$, $\boldsymbol{\alpha_i}$	the Lagrange multipliers (p. 22)
$f(\mathbf{x})$	the decision function (p. 22)
$K_d(\mathbf{x_i}, \mathbf{x_j})$	the RBF kernel function (p. 23)
γ	the scale of the kernel (p. 23)
$G_f(m, n)$	the Gabor filter function (p. 36)
(m, n)	the pixel location (p. 36)
σ_m, σ_n	the spatial extent and bandwidth of the filter (p. 36)

λ_g the wavelength of the filter (p. 36)

ϕ_g the phase of the filter (p. 36)

(m', n') the pixel location after orientation rotation (p. 36)

θ_g the orientation normal to the parallel stripes of a Gabor function (p. 36)

$I[m, n]$ the intensity of the pixel located at (m, n) in a polar radar image (p. 39)

$v_{1,L}$ the horizontal low frequency component (p. 40)

$v_{1,H}$ the high frequency component (p. 40)

$g_l[k]$ the low pass filter of 1-D DWT (p. 40)

$g_h[k]$ the high pass filter of 1-D DWT (p. 40)

$I_{1,LL}$ the low-low channel output (p. 40)

$I_{1,HL}$ the high-low channel output (p. 40)

$I_{1,LH}$ the low-high channel output (p. 40)

$I_{1,HH}$ the high-high channel output (p. 40)

u_1, u_2, u_3 the one-dimensional filter masks (p. 41)

$\mathbf{D}_l$ the l^{th} DCT mask (p. 41)

Y_l the filtering output generated by the l^{th} DCT mask (p. 41)

$Y_{var,l}$ the local variance of Y_l (p. 41)

W the size of the moving window (p. 42)

μ_l the Gaussian smoothing result of Y_l (p. 42)

X_T the texture feature vector input into the SOM-based model (p. 45)

D the total number of texture feature images (p. 45)

X_{Ti} the i^{th} element in the texture feature vector (p. 45)

w_j the weight vector of the j^{th} neuron (p. 45)

w_{ji} the i^{th} element of w_j (p. 45)

N the total number of neurons (p. 45)

$e_j(X_T)$ the Euclidean distance between X_T and the j^{th} neuron (p. 45)

$\Psi(X_T)$ the "winning" neuron of X_T (p. 45)

Δw_{ji} the value change of w_{ji} within an iteration (p. 45)

$\eta(t)$	the learning rate as a function of the current training iteration t (p. 46)
t	the current iteration (p. 46)
$T_{j,\Psi(X_T)}(t)$	the function determined by the distance between neuron j and $\Psi(X_T)$ (p. 46)
$S^2_{j,\Psi(X_T)}$	the distance between neuron j and $\Psi(X_T)$ (p. 46)
$\sigma(t)$	the function that determines the value of $T_{j,\Psi(X_T)}(t)$ (p. 46)
η_0, τ_η, σ_0, τ_σ	the constants in the which are tuned via validation (p. 46)
$z_d(r,s)$	the distance between two clusters (p. 48)
n_r	the number of neurons in cluster r (p. 48)
n_s	the number of neurons in cluster s (p. 48)
$\bar{x}_r$	the centroid of cluster r (p. 48)
$\bar{x}_s$	the centroid of cluster s (p. 48)
$Kappa$	the Kappa coefficient (p. 57)
TP_i	the true positives of the i^{th} class (p. 57)
TN_i	the true negatives of the i^{th} class (p. 57)
FP_i	the false positives of the i^{th} class (p. 57)
FN_i	the false negatives of the i^{th} class (p. 57)
F_i	a certain feature map (p. 65)
$\mathrm{BN}_R(F_i)$	the output of F_i after batch normalization (p. 65)
$E(F_i)$	the mean of F_i (p. 65)
$Var(F_i)$	the variance of F_i (p. 65)
ϵ_R	the arbitrarily small constant added in the denominator (p. 65)
γ_R	the scale parameter subsequently learnt in the optimization process (p. 65)
β_R	the shift parameter subsequently learnt in the optimization process (p. 65)
MA_i	the mean accuracy of the i^{th} class (p. 76)
IoU_i	the intersection over union of the i^{th} class (p. 77)

F_{1i} the boundary F_1 contour matching score of the i^{th} class (p. 77)

PR_i the positive rate of the i^{th} class (p. 77)

$Cont(I)$ the local contrast of radar image I (p. 90)

$Sat(I)(used)$ the saturation of radar image I (p. 91)

$\mathbf{W_1^i}$ the i^{th} filter convolved with the input radar image (p. 93)

$\mathbf{f_1^i}$ the filtering output generated by $\mathbf{W_1^i}$ (p. 94)

$\mathbf{B_1^i}$ the bias term of $\mathbf{W_1^i}$ (p. 94)

$\mathbf{F_1^i}$ the i^{th} feature map generated by the pixel-wise maximization operation for every 4 consecutive feature maps (p. 94)

$\mathbf{F_2^i}$ the i^{th} filter that generates multi-scale feature map from $\mathbf{F_1^i}$ (p. 95)

$\mathbf{F_3^i}$ the local extremum of $\mathbf{F_2^i}$ (p. 95)

$\Omega(m_0, n_0)$ the 7×7 neighborhood centered at the pixel located at (m_0, n_0) (p. 95)

$\mathbf{F_4}$ the network output (p. 95)

t_{max} the maximum limit of the regression output (p. 95)

t_{min} the minimum limit of the regression output (p. 95)

$\mathbf{W_4}$ the $6 \times 6 \times 48$ filter (p. 95)

$\mathbf{B_4}$ the bias terms for $\mathbf{W_4}$ (p. 95)

$\mathbf{t}(m, n)$ the pixel intensity of a pixel located at (m, n) in the transmission map (p. 96)

$\mathbf{J}(m, n)$ the output of $\mathbf{t}(m, n)$ generated by the atmospheric scattering model (p. 96)

$P(m, n)$ the rain-contaminated region identification result of $I(m, n)$ (p. 96)

c_f the centroid of the rain-free region (p. 96)

c_r the centroid of the rain-contaminated region (p. 96)

$d_f(m, n)$ the Euclidean distance between $I(m, n)$ and c_f (p. 96)

$d_r(m, n)$ the Euclidean distance between $I(m, n)$ and c_r (p. 96)

$R_c(m, n)$ the intensity of the pixel located at (m, n) in the rain-corrected radar image (p. 97)

y'	the radar-derived wind speed (p. 101)
y_i	the anemometer-measured wind speed for the i^{th} sample (p. 102)
y_i'	the radar-derived wind speed for the i^{th} sample (p. 102)
ξ_i, ξ_i^*	the values of estimation errors that exceed either of the two margins around the hyperplane (p. 102)
K_L, α_L, β_L	the hyperparameters in the normalization (p. 121)
$\mathbf{ss}$	the sum of squares of the elements in the normalization window (p. 121)
x_{in}	the input element of the local normalization response layer (p. 121)
x_{out}	the output element of x_{in} (p. 121)
H_s'	the estimated wave height (p. 123)
$loss$	the loss function (p. 123)
$\mathbf{z_i}$	the output of the update gate (p. 125)
$\mathbf{W^{(z)}}$, $\mathbf{U^{(z)}}$	the weight vectors of the update gate (p. 125)
$\mathbf{x_{in}}$	the input of the current cell (p. 125)
$\mathbf{y_{out}}$	the output of the current cell (p. 125)
h_i	the new hidden state generated by current cell (p. 123)
h_{i-1}	the hidden state of preceding cell (p. 125)
$\mathbf{re_i}$	the output of the reset gate (p. 125)
$\mathbf{W^{(r)}}$, $\mathbf{U^{(r)}}$	the weight vectors of the reset gate (p. 125)
$\mathbf{W^{(h)}}$, $\mathbf{U^{(h)}}$	the weight vectors to generate the final output (p. 126)
$h_i^{'}$	a vector used to store the relevant information from the past (p. 126)

List of abbreviations

SAR	Synthetic Aperture Radar (p. 2)
HF	High Frequency (p. 2)
HH	Horizontal Transmit and Horizontal Receive (p. 2)
RCS	Radar Cross Section (p. 3)
LGM	Local Gradient-based Method (p. 4)
ILS	Intensity Level Selection (p. 5)
SWH	Significant Wave Height (p. 8)
FFT	Fast Fourier Transform (p. 9)
EEMD	Ensemble Empirical Mode Decomposition (p. 8)
MTF	Modulation Transfer Function (p. 9)
SNR	Signal-to-Noise Ratio (p. 9)
CWT	Continuous Wavelet Transform (p. 9)
EOF	Empirical Orthogonal Function (p. 10)
ANN	Artificial Neural Network (p. 10)
SVM	Support Vector Machine (p. 10)
CNN	Convolutional Neural Network (p. 10)
ML	Machine Learning (p. 12)
ZPP	Zero Pixel Percentage (p. 15)
MSG	Meteosat Second Generation (p. 16)
DRDC	Defence Research and Development Canada (p. 19)
RBF	Radial Basis Function (p. 23)

SMO	Sequential Minimal Optimization (p. 23)
PDF	Probability Density Function (p. 25)
DCT	Discrete Cosine Transform (p. 34)
DWT	Discrete Wavelet Transform (p. 34)
SOM	Self Organizing Map (p. 34)
ADCP	Acoustic Doppler Current Profiler (p. 34)
NRCS	Normalized Radar Cross Section (p. 50)
TP	True Positives (p. 55)
FN	False Negatives (p. 55)
FP	False Positives (p. 55)
FCN	Fully Convolutional Network (p. 56)
BN	Batch Normalization (p. 61)
ReLU	Rectified Linear Unit (p. 61)
SGDM	Stochastic Gradient Descent with Momentum (p. 63)
HCD	High Clutter Direction (p. 66)
HPP	High Pixel Percentage (p. 66)
HPC	High-performance Computing (p. 68)
GPU	Graphics Processing Unit (p. 68)
CPU	Central Processing Unit (p. 68)
PC	Personal Computer (p. 68)
MA	Mean Accuracy (p. 71)
IoU	Intersection of Union (p. 72)
AI	Artificial Intelligence (p. 75)
ECMWF	European Centre for Medium-Range Weather Forecasts (p. 76)
RMSD	Root-Mean-Square Deviation (p. 79)
SVR	Support Vector Regression (p. 82)
RGB	Red, Green and Blue (p. 86)
CC	Correlation Coefficient (p. 107)

RNN	Recurrent Neural Network (p. 111)
GRU	Gated Recurrent Unit (p. 111)
CGRU	Convolution Gated Recurrent Unit (p. 111)
NOMAD	Navy Oceanographic Meteorological Automatic Device (p. 112)
FC	Fully Connected Layer (p. 118)
LSTM	Long Short Term Memory (p. 120)
ADAM	Adam Algorithm (p. 123)

Chapter 1

Introduction

1.1 Research Rationale

Real-time monitoring of sea surface wind and wave parameters facilitates the forecasting of local sea states, and therefore is crucial to the safety and efficiency of various nearshore and offshore activities, such as offshore drilling, cargo shipping, and the construction and maintenance of offshore infrastructure. The collected sea surface parameter data are also important for the study of oceanography. Traditionally, in-situ sensors, such as anemometers and wave buoys deployed nearby, are used to obtain those parameters. Although the estimation accuracy is satisfactory, they have certain limitations. For example, anemometer measurements might be negatively affected by movements of ships or platforms and local turbulences induced by a ship's hull or platform installations [1]. Buoys can only be deployed at fixed positions with limited spatial coverage. Besides, the cost associated with deployment and maintenance is relatively high. The swift currents in deep water could also make mooring a wave buoy enormously difficult. In addition, as extreme weather events on the ocean (e.g., extreme wave events) are becoming more frequent and unpredictable due to climate change, it is necessary to employ multiple approaches for sea surface parameter measurements, which will make sea state monitoring more accessible and accurate.

Over the past few decades, various types of remote sensing instruments, such as airborne and spaceborne synthetic aperture radar (SAR) [2, 3, 4], high frequency (HF) radar [5, 6, 7] and X-band marine radar [8, 9, 10] have been used more and more widely for sea surface parameter estimation. Each type of sensor has its own pros and cons. For example, although SAR provides images with a relatively wide range of spatial information, its demands in instruments (high power requirement and complex circuit designs) as well as the expenses of deployment result in high cost. Also, it is difficult to achieve high-sampling frequency (higher than once per hour). As for HF radar, while it is able to obtain the temporal information of the sea surface with wide spatial coverage (up to 200 km or more), installing the receiver requires significant tracts of coastline, and the cost associated with its deployment and maintenance is relatively high as well. On the other hand, thanks to the development of faster hardware with more memory in the past two decades, X-band marine radar, which can image the sea surface with high temporal (around once per 2 s) and spatial resolution (around 10 m), has become a popular sensor for a variety of ocean remote sensing tasks, such as target tracking [11], bathymetry mapping [12, 13], rip current detection [14], coastal upwelling estimation [15], internal wave analysis [16], tidal observations [17], sea ice drift measurement [18], and sea surface wind [19], wave [9, 20], and current [8, 21, 22, 23] parameter estimation. Moreover, as a "dry" sensor, it can be easily deployed in different locations (e.g., shore-based towers, offshore platforms and vessels) with relatively low installation and maintenance cost. In addition, as practically most ships are already equipped with X-band radars for surveillance and navigation purposes, those real-time sea surface data can be obtained at a small extra cost [24].

For conventional non-coherent X-band marine radars operating at a low grazing-incidence angle with horizontal transmit and horizontal receive (HH)-polarization, the backscatter signal is mostly generated by the Bragg resonance between radar-transmitted electromagnetic waves and centimetre-scale ripples due to their comparable wavelengths. Those short waves from different ranges and azimuths are then modulated by longer waves according to several mechanisms, such as hydrodynamic modulation (induced by interactions between capillary

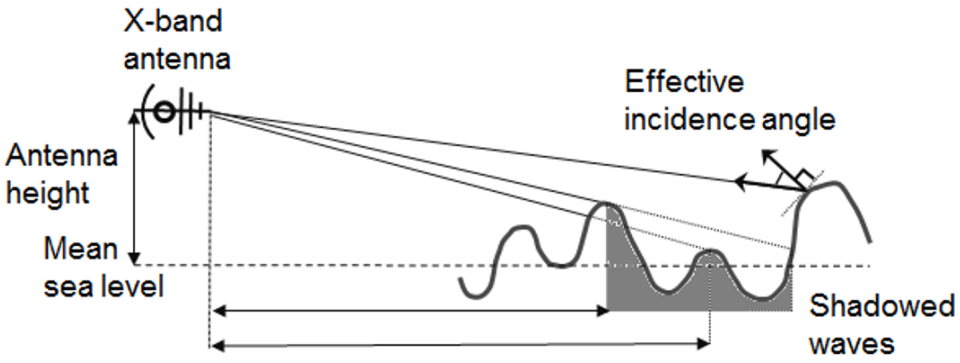

Figure 1.1: An illustration diagram of shadowing and tilt modulation.

waves and long waves), tilt modulation (due to the varying effective incidence angles of the radar signals along the slopes of long waves), and shadowing (due to the obstruction of radar signals caused by higher waves). This leads to the visibility of longer waves (stripe-like patterns) in the radar image [25]. A diagram illustrating the shadowing effect and tilt modulation is presented in Fig. 1.1. Since the radar backscatter is caused by the sea surface roughness mainly induced by local winds, wind information can be obtained by exploiting the dependence of the radar cross section (RCS) on wind speed [26] and the angle between radar look direction and wind direction [27]. On the other hand, the retrieval of wave parameters such as wave height, wave period, wave direction, and wavelength from marine X-band radar image sequences is mainly based on directional wave spectra derived from radar image sequences [28].

1.2 Literature Review

1.2.1 Wind Parameter Estimation Methods

Sea surface Wind stress is one of the main forces driving ocean dynamic processes and plays and important role in the energy exchange between the ocean and the atmosphere [29]. It has been found in [30] that the presence of wind-induced streaks can be observed in temporally-integrated X-band marine radar images (an example is shown in Fig. 1.2(b)). It can be observed that wave signatures are filtered out because the local variance of radar backscatter

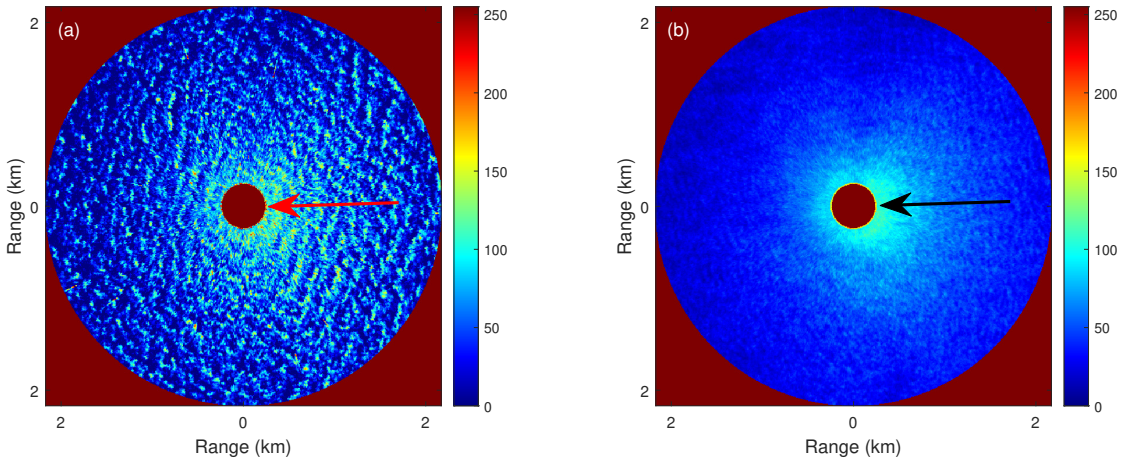

Figure 1.2: (a) An X-band marine radar backscatter image with surface wave signatures. The arrow indicates the direction from which the wind blows measured by an anemometer. The colorbar indicates the mapping of image pixel intensities into the jet colormap. (b) The temporally-integrated image of 32 consecutive radar images including (a). The arrow indicates the direction from which the wind blows measured by an anemometer.

is canceled out in a longer period. Based on the observation that those streaks are aligned in local wind direction with a typical spacing of the order of 100 m, a local gradient-based method (LGM) was proposed for wind direction estimation. However, wind streaks may be difficult to be observed from some marine radar data (such as Fig. 1.2(b)), especially those obtained from moving vessels. Hence, methods with higher robustness should be developed.

Previous studies [26, 27] have demonstrated that for X-band radar operating at grazing incidence with HH polarization, its radar backscatter only exhibits one peak, which lies in the upwind direction. Thus, wind direction can be derived from the dependence of backscatter intensity on the relative azimuth between antenna look direction and wind direction. As for wind speed, it can be retrieved by establishing an empirical model relating wind speed to backscatter intensity information extracted from the radar data. Based on these principles, a curve-fitting based method for wind parameter estimation from shipborne marine radar data in [24]. Given a radar image in range-azimuth coordinates, the average intensity over range for each azimuthal direction is calculated and denoted as σ_Θ. A cosine-squared function is

used to relate azimuth Θ to σ_Θ, which can be expressed as

$$\sigma_\Theta = a_0 + a_1 \cos^2(0.5(\Theta - a_2)), \tag{1.1}$$

where a_0, a_1 and a_2 are parameters determined through least-squares fitting. An example of the curve fitting result is presented in Fig. 1.3. The estimated wind direction corresponds to the azimuth located at the peak of the fitted function. It should also be noted that σ_Θs obtained from azimuths shadowed by ship structures were not used for curve fitting. since the wind speed is dependent on the average backscatter intensity, the cosine-squared function is first integrated over azimuth, which can be expressed as

$$\bar{\sigma} = \frac{1}{2\pi} \int_0^{2\pi} \left(a_0 + a_1 \cos^2(0.5(\Theta - a_2))\right) d\Theta. \tag{1.2}$$

Then, as shown in Fig. 1.4, a third-degree polynomial model between $\bar{\sigma}$ and wind speed (denoted as w_{spd}) is trained using images obtained under different wind speeds and simultaneous anemometer measurements. The function can be expressed as

$$\bar{\sigma} = b_3 w_{spd}^3 + b_2 w_{spd}^2 + b_1 w_{spd} + b_0. \tag{1.3}$$

Then, an intensity level selection (ILS)-based method is proposed in [19]. In particular, a set of intensity levels L_is are first determined based on the range of radar image pixel intensities. Each radar image sequence in range–azimuth coordinates is integrated over time. Then, in order to reduce the noise in range direction, the temporally-integrated image is smoothed in range direction using an averaging technique. The range distance vector $r_i(\Theta)$ for each L_i is computed, which consists of the maximum range distance where the intensity is equal to or greater than L_i in a every azimuth direction. $r_i(\Theta)$ is then smoothed in the azimuthal direction by conducting a 5° moving average. For the k^{th} integrated and smoothed image, its lowest L_i (denoted as $L_i(k)$) with all its associated smoothed maximum ranges being greater than an inner distance boundary (determined based on the

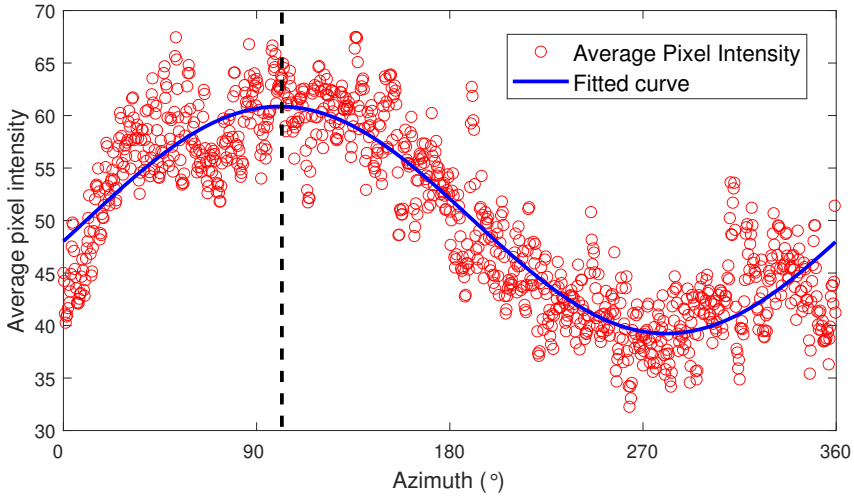

Figure 1.3: Average pixel intensity as a function of azimuth for an HH-polarized X-band marine radar backscatter image. The corresponding best-fit curve is shown in blue. The estimated wind direction corresponds to the azimuth located at the peak of the fitted function, which is indicated by the black dash line.

range coverage of the radar) is designated as the selected intensity level. After applying the above procedures to the first 16 integrated and smoothed images, rather than all the L_is, only the selected intensity level from the previous image (i.e., $L_i(k-1)$) and its upper and lower defined intensity levels (i.e., $L_{i-1}(k-1)$ and $L_{i+1}(k-1)$) are used for generating the range distance vectors. This makes the algorithm more efficient. Fig. 1.5 shows the smoothed $r_i(\Theta)$ (indicated by the red curve) for $L_i = 40$ in a temporally-integrated radar image. Then, wind speed (denoted as w_{spd}) can be determined by relating to the maximum range of the smoothed $r_i(\Theta)$ (i.e., $\max\{r_i(\Theta)\}$), which can be expressed as

$$w_{spd} = \alpha_i \times \max\{r_i(\Theta)\}, \tag{1.4}$$

where α_i refers to a conversion rate to be determined by least square fitting using ground truth wind speed. Specifically, its value can be obtained from the selected intensity level L_i by fitting a third-order polynomial continuous function. The function can be expressed as

$$\alpha = \beta_3 L^3 + \beta_2 L^2 + \beta_1 L + \beta_0, \tag{1.5}$$

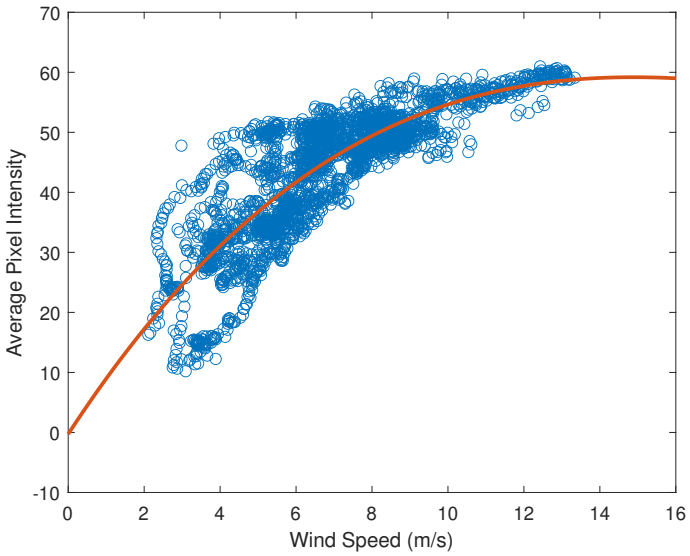

Figure 1.4: Scatter plot showing the wind speed from an anemometer, corresponding radar image average pixel intensity, and the best-fit curve based on a third-degree polynomial function in Eq. (1.3).

where $\beta_0, ..., \beta_3$ are parameters that can be found using the total least squares algorithm. Since the pixel intensities of the marine radar image are dependent not only on wind condition, but also on sea states, especially in nearshore regions where swell appears frequently, in [31], a linear function is used to fit between wind speed and the average pixel intensity of the radar image, and then considered the influence of sea states by incorporating significant wave height (SWH) into the wind speed estimation function.

Instead of directly extracting information from radar images in spatial and time domain, some researchers exploit the intrinsic characteristics of radar images by conducting some sorts of transform to radar images. After preprocessing, information extracted from transform domain analysis is fitted to empirical functions for wind parameter estimation. For example, in [32], after conducting 3D-fast Fourier transform (FFT), the spectral noise was separated from ocean wave signal and used to infer wind speed through a linear equation. In [33, 34], Huang and Wang proposed a novel spectral integration-based algorithm, which uses different functions to measure wind parameters from images obtained under rainless and rainy conditions. In [35, 36], three ensemble empirical mode decomposition

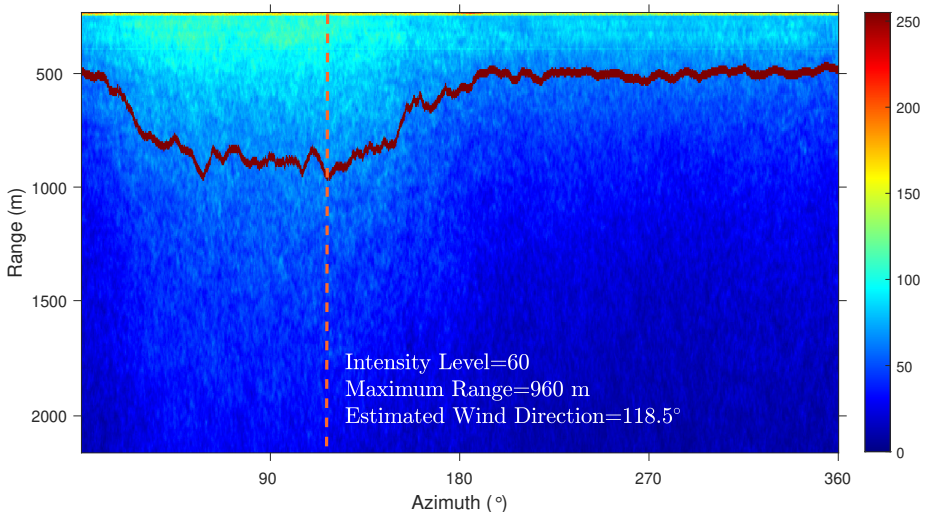

Figure 1.5: An example of a temporally-integrated radar image. The selected intensity level is 40. The red line corresponds to the smoothed maximum range distance in each azimuth. The dash line corresponds to the azimuth with the greatest distance, which is regarded as the radar-derived wind direction.

(EEMD)-based methods (i.e., 1D-EEMD, 2D-EEMD, normalization-incorpoarted EEMD) were proposed for wind parameter estimation.

1.2.2 Significant Wave Height Estimation

Significant wave height (SWH) is defined as the average wave height, from trough to crest, of the highest one-third of the waves. Under deep water wave assumption, it can be calculated as four times the standard deviation of the surface elevation, or four times the square root of the zeroth-order moment (area) of the wave spectrum [37]. Since non-coherent marine radars are not radiometrically calibrated, radar image intensities are not directly related to sea surface elevations. Thus, SWH cannot be determined directly and certain inversion schemes need to be derived. One of the most widely accepted inversion schemes is to first obtain the high-pass filtered image spectrum using the 3-D FFT. For linear ocean, the dispersion relationship for surface waves is given by

$$\omega = \sqrt{gk \tanh(kd)} + \mathbf{k} \cdot \mathbf{U}, \tag{1.6}$$

where ω is the frequency, $\mathbf{k}$ is the wavenumber vector, $k = |\mathbf{k}|$, d is the water depth, and $\mathbf{U} = (U_x, U_y)$ is the surface ocean current. Then, after obtaining the current speed (estimated by means of a least square fit using the dispersion relation [8, 38]) and water depth information, an empirical modulation transfer function (MTF) developed in [9] is applied to derive wave spectrum from the high-pass filtered image spectrum, which can be expressed as

$$F_W^3(\vec{k}, \omega) = F_F^3(\vec{k}, \omega) \cdot T_M(\vec{k}), \tag{1.7}$$

where $F_F^3(\vec{k}, \omega)$ and $F_W^3(\vec{k}, \omega)$ denote the 3-D wavenumber-frequency image spectrum and wave spectrum, respectively, and $T_M(\vec{k})$ is the modulation transfer function. As an extension of the technique developed for SAR images [39], SWH can be estimated by assuming a linear relation between SWH (denoted as H_s) and the square root of signal-to-noise ratio (SNR) [40], which can be expressed as

$$H_s = c_0 + c_1\sqrt{SNR}. \tag{1.8}$$

Specifically, the signal refers to the total energy of the estimated wave spectrum while the noise relates to the spectral energy of the speckle noise within the radar image [41]. It should be noted that the coefficients of the linear model (i.e., c_0 and c_1) need to be determined via a calibration process, which requires the measurements from a reference sensor, such as a wave-measuring buoy. Besides, later studies also found that the relationship is not completely linear due to factors such as different SNR calculation methods, environmental changes in the sea area, and radar system differences [42]. As a result, the SNR-based method might not always produce accurate estimation, especially when the radar is installed on a moving platform [43]. Other than the SNR-based algorithms, a couple of other methods have been proposed for SWH estimation by conducting spatial and/or temporal analysis on radar images, such as the 2-D continuous wavelet transform (CWT)-based algorithms [44], array beamforming-based algorithms [45], and ensemble empirical mode decomposition (EEMD)-based algorithms [43]. On the other hand, in order to estimate SWH directly from

radar images without an external reference, multiple types of calibration-free methods have been derived as well, such as shadowing analysis-based algorithms [46, 47, 48], empirical orthogonal function (EOF)-based algorithms [49], maximum correlation analysis-based algorithms [50], and shadowing mitigation-based algorithms [51]. Nevertheless, the design of those calibration-free algorithms is generally more complicated than the ones requiring ones. Also, many parameters and thresholds appeared in calibration-free algorithms need to be determined empirically. It should be noted that the computational time of both calibration-free and calibration-required algorithms are relatively long. Hence, in order to simplify the design of wave estimation model and improve computational efficiency, while providing more accurate results, machine learning-based nonparametric regression models have been incorporated in several works for SWH estimation, including artifical neural networks (ANNs) [52], support vector machines (SVMs) [53, 41], and convolutional neural networks (CNNs) [54]. In particular, as an end-to-end system, CNNs are able to take the radar image as input and regress SWH directly by learning to extract deep spatial features automatically. However, while the radar image sequence contains both spatial and temporal information, the CNNs used in [54] can only extract the spatial features from each individual image without conducting temporal analysis. Besides, as only simulated radar images are employed in [54] for model training and testing, whether those CNN-based models perform well on real radar data is unknown.

1.2.3 Rain Influence and Mitigation

Although the classic methods for wind and wave parameter estimation mentioned above have all proven to produce promising results using real marine radar data, it is found that the presence of rain will negatively affect the estimation accuracy of all those methods. That is because under rainy conditions, radar backscatter will be altered by rain via volume scattering, attenuation due to raindrops in the intervening atmosphere, and changes in sea surface roughness resulting from rain impinging the ocean [55, 56]. As shown in Fig.

1.6(b), the wave signatures in the radar image may also be blurred as a result. In previous studies, radar images obtained during rainy periods were usually discarded. In many weather-sensitive offshore activities (such as cargo shipping, offshore oil and gas drilling), the real-time monitoring of sea state condition requires the sensor to operate 24/7. Under rain condition (including stormy weather), sensors such as video camera may perform poorly due to a lack of clear vision. Recycling rainy X-band radar images collected for sea state monitoring can further improve the safety and efficiency of those activities. Recently, several novel or modified methods have been developed to obtain wind and wave parameters using rain-contaminated X-band marine radar images. For wind parameter estimation, Huang and Gill [10] developed two separate third-order polynomial functions to estimate wind speed from rainless and rainy X-band marine radar images, respectively. It has been observed in [57] that since rain may blur wave signatures, the textures of rain-contaminated echoes were different from those of less-contaminated or rain-free echoes. Therefore, texture analysis was incorporated into the ILS algorithm to detect the presence of rain and measure wind parameters from rain-contaminated radar images with higher accuracy. Also, as mentioned above, information extracted from the radar image using certain transform techniques is used for curve fitting to obtain wind parameters, including the spectral integration-based algorithm [33, 34] and EEMD-based methods (i.e., 1D-EEMD, 2D-EEMD, normalization-incorpoarted EEMD) [35, 36]. Compared to traditional methods, those methods obtain higher estimation accuracy from both rain-free and rain-contaminated images. As for SWH estimation under rainy conditions, so far only two preliminary studies have been conducted [58, 59]. In these methods, the influence of rain is first mitigated or corrected using image filtering techniques. Those rain-corrected images are then subjected to the previously proposed wave estimation methods. Although in [59], the estimation error has been reduced in some sense, the method is still relatively coarse and the accuracy can be further improved.

Among those novel or modified algorithms for wind and wave parameter estimation, a common and important step is to detect the presence of rain in radar images. However, since little is known about the dominant backscatter mechanism for X-band marine radar

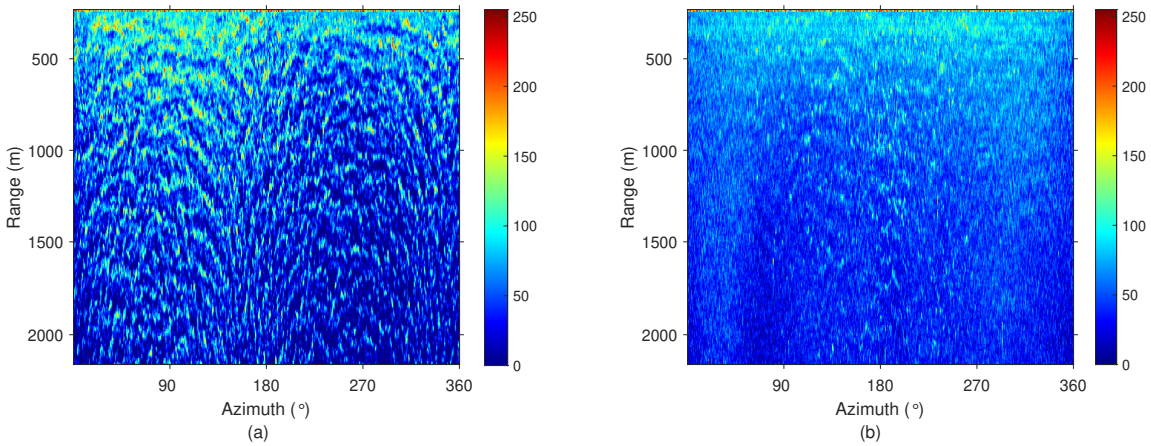

Figure 1.6: Comparison between two X-band marine radar backscatter images obtained under rainless (a) and rainy (b) conditions. The wave signatures in (b) are blurred by rain significantly

operating at grazing incidence in presence of rain [24], the determination of rainy images is based on empirical analysis. For example, as it has been observed that rain generally enhances the average pixel intensity of the radar image, some sorts of parameters (e.g., the zero pixel percentage proposed in [24]) can be extracted from the image and used as thresholds to determine whether an image or a certain azimuth is contaminated by rain. Although the threshold-based methods are quite straightforward, their detection accuracies have not been investigated. Besides, they cannot determine whether a certain pixel or region is rain-free or rain-contaminated. Furthermore, the proposed filtering techniques for correcting the influence of rain on radar images are still relatively preliminary and their effectiveness lacks further validation. In addition, as each of those methods was only validated using a single marine radar image dataset, whether they are effective for other datasets remains unknown.

As a state-of-the-art approach for data-driven problems, machine learning (ML)-based algorithms have been widely applied in various ocean remote sensing tasks, including the SWH estimation mentioned above. It is also reasonable to apply ML-based algorithms for rain-contaminated image/pixel detection and mitigation. Novel ML-based regression methods can also be developed to improve wind and wave estimation accuracy and robustness

under both rainless and rainy conditions. Thus, a ML-based scheme for determining wind and wave parameters incorporated with pixel-based rain detection and correction techniques is worth pursuing.

1.3 The Scope of the Thesis

In this thesis, multiple ML-based methods are first developed to detect X-band marine radar images containing rain clutter and identify rain-contaminated regions in these images. Then, image dehazing techniques are introduced to mitigate the influence of rain on wind direction estimation. ML-based models are designed and trained to estimate wind speed and SWH with high accuracy, efficiency, and robustness. The thesis is organized as follows:

In Chapter 2, a support vector machine (SVM)-based model is proposed to detect radar images collected in the presence of rain. In order to further identify pixels contaminated by rain in each radar image, two methods are developed based on unsupervised and supervised ML techniques, respectively in Chapter 3. Chapter 4 introduces the application of image dehazing techniques for rain mitigation in wind parameter estimation, as well as the support vector regression (SVR)-based wind speed estimation model. Chapter 5 presents two deep neural-based models for SWH estimation. Finally, a conclusion and outlook of this research are outlined in Chapter 6.

The research presented in this thesis has been published in one book chapter and five refereed journal papers:

1. X. Chen, W. Huang, and B. Lund, "Wind parameter measurement using X-band marine radar images," in Chapter 17 of Ocean Remote Sensing Technologies: High Frequency, Marine and GNSS-Based Radar, W. Huang and E. W. Gill, IET, 2021. (in press)

   This book chapter provides a comprehensive review of the methods proposed for wind parameter estimation from X-band marine radar images (Chapter 1).

2. X. Chen, W. Huang, C. Zhao, and Y. Tian, "Rain detection from X-band marine radar images: a support vector machine-based approach," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 3, pp. 2115-2123, 2020.

   This paper presents the SVM-based model to classify between radar images obtained under rainless and rainy conditions (Chapter 2).

3. X. Chen and W. Huang, "Identification of rain and low-backscatter regions in X-band marine radar images: an unsupervised approach," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 6, pp. 4225-4236, 2020.

   This paper introduced a method to identify rain-contaminated and low-backscatter regions in radar images using texture features and unsupervised learning techniques (Chapter 3).

4. X. Chen, W. Huang, M. Haller and R. Pittman, "Rain-contaminated region segmentation of X-band marine radar images with an ensemble of SegNets," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 141-154, 2021.

   This paper proposes an image segmentation CNN (SegNet)-based model to segment between rain-free and rain-contaminated regions in X-band marine radar images (Chapter 3).

5. X. Chen, W. Huang, and M. Haller, "A novel scheme for extracting sea surface wind information from rain-contaminated X-band marine radar images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 5220-5434, 2021.

   This paper first introduces a method to correct the influence of rain on pixel intensities based on an image dehazing CNN (DehazeNet), which significantly improves wind direction estimation accuracy under rainy conditions. Then, texture features are extracted for from rain-corrected radar data training the SVR-based wind speed estimation model (Chapter 4).

6. X. Chen and W. Huang, "Spatial-temporal convolutional gated recurrent unit network

for significant wave height estimation from shipborne marine radar data," *IEEE Trans. Geosci. Remote Sens.*, 2021. (in press, DOI: 10.1109/TGRS.2021.3074075)

This paper introduces two deep neural network-based models for SWH estimation that can obtain higher estimation accuracy and efficiency compared to the SNR-based method (Chapter 5).

Chapter 2

Rain Detection Using Support Vector Machines (SVMs)

In [24], zero pixel percentage (ZPP) was used as a threshold to detect rain-contaminated X-band marine radar images. The ZPP value of a certain image is the ratio of the number of image pixels with zero intensity to the total number of pixels. The idea behind the ZPP method is that due to the shadowing effect, many pixels in the radar image have intensities equal or close to zero. Under rain conditions, the rain echoes mask those shadowing areas, which causes a significant decrease in the percentage of pixels with very low intensities. In [24], the value of zero intensity is identically zero, while in [60, 33, 34], all gray scale intensities lower than 5 are defined as zero intensity. Although the ZPP threshold method is quite straightforward, the value of the threshold as well as the range of zero intensity are difficult to determine because of their dependence on the radar systems [59]. Also, the accuracy of the ZPP threshold method for rain-contaminated image detection has never been studied. Since the detection accuracy may further affect the performance of the above-mentioned ocean parameter measurement algorithms, it is necessary to propose a novel detection method, which is both accurate and easy to implement for different X-band marine radar systems.

In fact, rain detection using X-band marine radar images can be regarded as a classification task, which aims to classify radar images into rain-free and rain-contaminated types. In other words, if rain does exist, the backscatter intensity of simultaneous radar images will be altered and regarded as rain-contaminated. As a state-of-the-art technique for data classification and pattern recognition, support vector machines (SVMs) combines the error approximation to the data with the generalization of the model by applying the structural risk minimization principle introduced in [61, 62]. Also, compared to other pattern recognition techniques, SVMs have often been found to be able to achieve high accuracies even when only a few training samples are available [63]. SVMs have been widely applied in many research topics associated with remote sensing, such as hyperspectral image classification [63], land cover classifications from satellite images [64, 65] and retrieval of oceanic chlorophyll concentration [66]. In [67, 68], SVMs were applied for rainy areas detection using the Meteosat Second Generation (MSG) satellite and Kalpana-1 satellite data, respectively. On the other hand, in [69], a SVM-based method was proposed and used to detect raindrops from colour video imagery. However, so far SVMs have not been applied for rain detection in X-band marine radar images.

Here, the proposed SVM-based method and ZPP threshold algorithm are both applied to detect rain-contaminated X-band marine radar images. The chapter proceeds as follows. The SVM-based rain detection model is introduced and derived in Section 2.1. In Section 2.2, experimental results obtained from two shipborne radar systems using the proposed technique and the ZPP threshold method are analyzed and compared. A summary for this chapter is provided in Section 2.3.

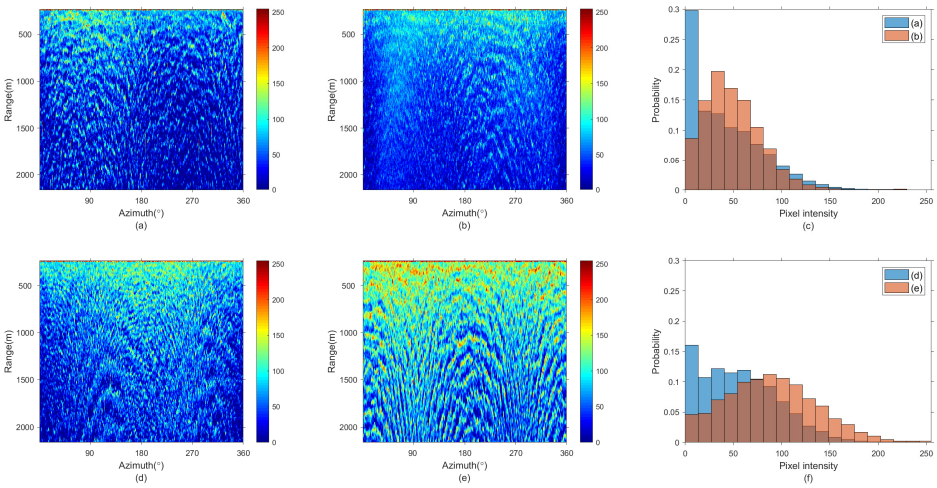

Figure 2.1: Rain-free and rain-contaminated Decca polar radar images with different wind speeds and their normalized pixel intensity histograms. (a) Rain-free image obtained under a wind speed of 6.1 m/s. (b) Rain-contaminated image obtained under a wind speed of 6.1 m/s. (c) The 19-bin normalized histogram of (a) and (b) in color blue and orange respectively. (d) Rain-free image obtained under a wind speed of 13.6 m/s. (e) Rain-contaminated image obtained under a wind speed of 13.5 m/s. (f) The 19-bin normalized histogram of (d) and (e) in color blue and orange respectively.

2.1 SVM-based Rain-Contaminated Image Detection

2.1.1 The Dependence of Radar Image Histogram on Wind and Rain

The marine radars used in this study are standard shipborne X-band non-coherent Decca and Furuno radars. One radar image is obtained after the antenna finishes each 360° scan of the sea surface. The radar data were collected from a moving ship during a sea trial.

For a given radar system, the radar backscatter power P_r from a point target is calculated by the Radar Equation [70]:

$$P_r = \frac{P_t G_t A_r \boldsymbol{\sigma}}{(4\pi R^2)^2} \tag{2.1}$$

where

P_t: the transmitter power (Watt)

G_t: the gain of the transmitting antenna (ratio)

A_r: the effective aperture of the receiving antenna (m^2)

$\boldsymbol{\sigma}$: the scattering cross section of the target (m^2)

R: the range from the radar to the target (metre).

Therefore, the radar backscatter is dependent on R and $\boldsymbol{\sigma}$. For the area of interest in this study, the range coverage is from 240-2160 m with a resolution of 7.5 m for both radars and the beam widths are 2° (Decca) and 1.9° (Furuno). A resampled B-scan image is obtained after a full rotation of the antenna. Each pixel in the B-scan image represents 7.5 m in range and around 0.36° in azimuth. As Eq. (2.1) manifests, backscatter intensity tends to decrease as range increases. For the scattering cross section, previous studies [71, 72] concluded that given a certain range distance and wave spectrum, the scattering cross section is dependent on the local wind speed as well as the angle between radar look direction and upwind direction, which can be expressed by a second-order harmonic empirical expression as:

$$\boldsymbol{\sigma^0}(w_{spd}, \phi, \Theta) = \sum_{n=0}^{2} B_n(w_{spd}, \phi) \cos\left[n(\Theta - w_{dir})\right] \tag{2.2}$$

where

$\boldsymbol{\sigma^0}$: the radar cross section per unit surface area

w_{spd}: the local wind speed

ϕ: the radar incidence angle

Θ: the radar look direction

w_{dir}: the local upwind direction

B_0: the steady state term accounts for major wind speed effects

B_1: the fundamental term accounts for the difference between backscatter at upwind and downwind

B_2: the second harmonic that accounts for difference between backscatter at upwind and crosswind.

Since B_0 increases with the increase of w_{spd}, backscatter intensities tend to increase with the increase of wind speed as well, which is also observed in the histogram distribution of Fig. 2.1(a) and (d). Different B_1 and B_2 account for different number of backscatter local maximums along the azimuth direction. For an X-band marine radar operating at grazing incidence with HH-polarization, there is only one maximum of backscatter which is located in the upwind direction, i.e., $B_1 \gg B_2$. However, Eq. (2.2) may not be valid in the presence of rain since the wave spectrum will become the sum of the rain-generated ring-wave spectrum and the wind-generated wave spectrum [73]. A histogram is an accurate representation of the distribution of numerical data. As in an image processing context, the histogram of an image refers to the probability distribution of the pixel intensity values. Since rain and wind speed will affect the radar backscatter, the histogram of the radar image is dependent on both wind speed and rain rate. It should be noted that the input feature vector contains the information of pixel intensity distribution. Since the pixel intensity distribution of the radar image is dependent on wind speed, the effect of wind speed is already implicitly included. Thus, it is not necessary to add the wind speed parameter explicitly into the model. Hence, it is feasible to first extract bin values from the normalized histogram and then build rain-contaminated image detection model based on SVMs.

2.1.2 Dataset Description

The radar data used for the present work were provided by Defence Research and Development Canada (DRDC). Data were collected during a sea trial approximately 220 km from the coast of Halifax (42°30′ N, 62°5′ W) in several time periods from November 25 to December 4, 2008. Since the water depths around the vessel location are around 200 m, a

deep-water assumption is valid here. Two marine radar systems, i.e., Decca and Furuno, were installed on the Canadian Navy research ship CFAV Quest [74]. Both radars were standard non-coherent X-band radar operating in short-pulse mode at 9.41 GHz with HH-polarization and grazing incidence angle. Other important parameters of the radar systems, including beam width, azimuth coverage and antenna rotation speed, are listed in Table 2.1. The received backscatter signal is amplified logarithmically and then imported to the Wave Monitoring System (WaMoS II) [75], which digitized radar backscatter into 8-bit binary numbers corresponding to image intensities from 0 to 255. In this work, the effect of motion on radar data and the rain signatures is not considered. During the radar data collection period, two anemometers mounted on the port and starboard of the ship, respectively, were used to measure simultaneous wind speed and direction. The average of the two measurement results in Fig. 2.3(a) and Fig. 2.4(a) show that the range of the wind speed is 2 to 16 m/s, which allows the proposed method to be implemented under typical wind conditions. The speeds of the ship during the data collection periods are shown in Fig. 2.3(b) and Fig. 2.4(b), respectively. The simultaneous rain rate data shown in Fig. 2.3(c) and Fig. 2.4(c) were obtained from the average measurements of two rain gauges which were deployed on the ship and used as ground truth to determine whether an image is contaminated by rain or not. Since the range coverage is up to just over 2 km, which is between microscale and mesoscale in meteorology, it is reasonable to employ single-point measurements as ground truth.

Table 2.1: Parameters of two radar systems

	Decca	Furuno
Polarization	Horizontal	Horizontal
Operating frequency	9.41 GHz	9.41 GHz
Range resolution	7.5 m	7.5 m
Pulse width	50 ns	50 ns
Range coverage	240 m - 2160 m	240 m - 2160 m
Beam width	2°	1.9°
Azimuth coverage	360°	Around 300°
Antenna rotation speed	28 rpm	40 rpm
Antenna height	21.9 m	16.5 m
Backscatter intensity level	0 - 255	0 - 255

2.1.3 SVM Algorithms for Rain Detection

A set of training samples are required in order to train the detection model, which can be expressed as $\{\mathbf{x_1}, \mathbf{y_1}\}$,$\{\mathbf{x_2}, \mathbf{y_2}\}$,...,$\{\mathbf{x_l}, \mathbf{y_l}\}$ ($\mathbf{x_i} \in R^n, \mathbf{y_i} = \pm 1$), where $\mathbf{x_i}$ represents an n-dimensional vector consisting of the bin values, $\mathbf{y_i}$ implies whether the image is labeled as rain-free (-1) or rain-contaminated ($+1$), and l is the number of samples. For the feature vector $\mathbf{x_i}$, each of its elements represents the probability value associated with a certain bin. For example, the first element of $\mathbf{x_i}$ equals to the probability value for the first bin. Hence, the dimension of $\mathbf{x_i}$, i.e, n, equals to the number of bins. In this study, various values of n (ranging from 10 to 32) are selected in the following experiments to test how the value of n affects the detection accuracy (see Subsection 2.2.2). Here, l represents the number of radar images used to train the SVM-based detection model. As shown in Table 2.2, for Decca radar data, l is 280 (140 for both rain-free and rain-contaminated images). While for Furuno radar data, l equals to 314 (157 for both rain-free and rain-contaminated images). In this work, the training data do not overlap with the testing data. Given the training samples, SVMs can find the optimal separating hyperplane by maximizing the margin between rain-free and rain-contaminated cases. This refers to finding the $\mathbf{w}$ and b that satisfy

$$\mathbf{y_i}(\mathbf{w} \cdot \boldsymbol{\psi}(\mathbf{x_i}) + b) > 0, i = 1, 2, ..., l, \tag{2.3}$$

where $\mathbf{w}$ is a vector that is normal to the hyperplane and $\frac{b}{\|\mathbf{w}\|}$ determines the offset of the hyperplane from the origin along the normal vector $\mathbf{w}$. $\boldsymbol{\psi}(\cdot)$ is the kernel function that projects training samples from the input space to the feature space. If there exists a hyperplane satisfying Eq. (2.3), the training set is said to be linearly separable. In that case, after rescaling, the closest samples lie on either side of the hyperplane, satisfying $\mathbf{w} \cdot \boldsymbol{\psi}(\mathbf{x_i}) + b = 1$ or $\mathbf{w} \cdot \boldsymbol{\psi}(\mathbf{x_i}) + b = -1$, which are also called support vectors. When the data contain outliers, slack variables $\varepsilon_1, ..., \varepsilon_l \geq 0$ (one for each sample) are then introduced

to allow for exceptions:

$$\mathbf{y_i}(\mathbf{w} \cdot \boldsymbol{\psi}(\mathbf{x_i}) + b) \geq 1 - \varepsilon_i, i = 1, 2, ..., l, \tag{2.4}$$

where $\varepsilon_i = 0$ for non-outliners (including support vectors), and $\varepsilon_i > 0$ for the outliers. In particular, if $0 < \varepsilon_i < 1$, points are still on correct side of hyperplane but within the margin while if $\varepsilon_i > 1$, they are on wrong side of hyperplane. As the margin in linearly separable cases is called the "hard margin", the margin with outliers is defined as soft margin. Since we want most of the points to be at ideal locations, the slack variables can be incorporated into the objective function (i.e., risk function) $\mathbf{J}(\mathbf{w})$ as [61]

$$\mathbf{J}(\mathbf{w}) \equiv \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{l} \varepsilon_i. \tag{2.5}$$

Therefore, the optimization problem can be written as

$$\begin{aligned} &\min_{w} \mathbf{J}(\mathbf{w}) \\ &\text{s.t. } \mathbf{y_i}(\mathbf{w} \cdot \boldsymbol{\psi}(\mathbf{x_i}) + b) \geq 1 - \varepsilon_i, \ i = 1, 2, ..., l \end{aligned} \tag{2.6}$$

where $C > 0$ is a penalty term for misclassification. As the margin is calculated as $\frac{1}{||\mathbf{w}||}$, the optimal (maximal margin) hyperplane can be obtained by minimizing $\frac{1}{2}||\mathbf{w}||^2$ under Eq. (2.4). Hence, the objective function now becomes a (linearly constrained) quadratic optimization problem. The optimization problem is usually tackled by solving its dual form, which is obtained through the minimization of the Lagrange function. The Lagrange function, which consists of the objective function and the problem constraints, can be expressed as

$$\begin{aligned} L(\mathbf{w}, b, \varepsilon, \alpha, \mathbf{r}) = &\frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{l} \varepsilon_i - \sum_{i=1}^{l} \mathbf{r_i}\varepsilon_i \\ &- \sum_{i=1}^{n} \boldsymbol{\alpha_i} \left[\mathbf{y_i}(\mathbf{w} \cdot \boldsymbol{\psi}(\mathbf{x_i}) + b) - 1 + \varepsilon_i\right]. \end{aligned} \tag{2.7}$$

where $\mathbf{r_i}$ ($1 \leq i \leq l$, $\mathbf{r_i} \geq 0$) and $\boldsymbol{\alpha_i}$ ($1 \leq i \leq l$, $0 \leq \boldsymbol{\alpha_i} \leq C$) are Lagrange multipliers. Since it is often easier to solve the dual, the original problem then turns into looking for the solutions of Lagrange multipliers that maximize the following objective function

$$W(\alpha) = \sum_{i=1}^{l} \boldsymbol{\alpha_i} - \frac{1}{2} \sum_{i,j=1}^{l} \boldsymbol{\alpha_i \alpha_j y_i y_j} \boldsymbol{\psi}(\mathbf{x_i})^T \boldsymbol{\psi}(\mathbf{x_j}) \tag{2.8}$$

subject to the constraints

$$\sum_{i=1}^{l} \boldsymbol{\alpha_i y_i} = 0. \tag{2.9}$$

$W(\alpha)$ can be maximized by applying sequential minimal optimization (SMO) algorithm proposed in [76]. The resultant decision function has the following form:

$$f(\mathbf{x}) = \text{sgn}\left(\sum_{i=1}^{l} \boldsymbol{\alpha_i y_i} \boldsymbol{\psi}(\mathbf{x_i}) \cdot \boldsymbol{\psi}(\mathbf{x}) + b \right). \tag{2.10}$$

As for the selection of the kernel function, in this study, multiple types of kernel functions, such as linear kernel, polynomial kernel, and radial basis function (RBF) kernel, have been applied to train the SVM-based model. The RBF kernel is selected because it yields the best detection accuracy of 99.1% for the training samples. Thus, the relative distance between two samples $K_d(\mathbf{x_i}, \mathbf{x_j})$ can be expressed as

$$K_d(\mathbf{x_i}, \mathbf{x_j}) = \boldsymbol{\psi}(\mathbf{x_i}) \cdot \boldsymbol{\psi}(\mathbf{x_j}) = \exp(-\gamma \|\mathbf{x_i} - \mathbf{x_j}\|^2) \tag{2.11}$$

where $\gamma > 0$ determines the scale of the kernel. In order to maximize the detection accuracy, the values of hyperparameters, i.e., C and γ, are also optimized using the sequential minimal optimization (SMO) algorithm proposed in [77] because they significantly affect the accuracy of the model. The objective function (2.6) will reach a minimum point when C and γ are optimized.

2.2 Experiments and Results

2.2.1 Model Training

Table 2.2: Number of data for training & testing

	Training		Testing	
Radar system	Rain-free	Rain data	Rain-free	Rain data
Decca	140	140	3117	505
Furuno	157	157	4906	234

Table 2.3: Time periods of data

Radar system	Time periods of data	Time periods for training
Decca	23:43 Nov. 26 - 04:04 Nov. 28 11:16 Nov. 28 - 12:06 Nov. 29 12:06 - 13:22 Nov. 30 01:27 - 12:02 Dec. 01 12:05 - 19:30 Dec. 04	23:43 Nov. 26 - 12:18 Nov. 27 06:33 - 09:25 Nov. 29
Furuno	13:24 Nov. 25 - 23:25 Nov. 26 12:11 Nov. 29 - 11:04 Nov. 30 12:09 Dec. 01 - 12:03 Dec. 04	12:09 - 21:25 Dec. 01

A pre-processing step should be undertaken before extracting features from the radar image. It is found that radial noise lines with high pixel intensities caused by interference of other marine radars are present in many raw radar images used in this study. An example is shown in Fig. 2.2(a) with the noise lines outlined in red. A closer look of a noise line is shown in Fig. 2.2(c). As they contaminate the original texture of the radar image, a method proposed in [35] is used to detect and eliminate those noise lines. First, a 3×3 pixel kernel is convolved with the radar image for radial noise line detection. The first and third columns of the kernel are set to -1 and the middle column is set to 2. The noise lines can be located in the convolved product, in which the intensity levels of at least five continuous pixels in range direction are equal to 255. Then, each such noise pixel is replaced by the average of its two azimuthal adjacent pixels in the original radar image. Result in Fig. 2.2(b) and (d) shows that those noise lines have been effectively removed.

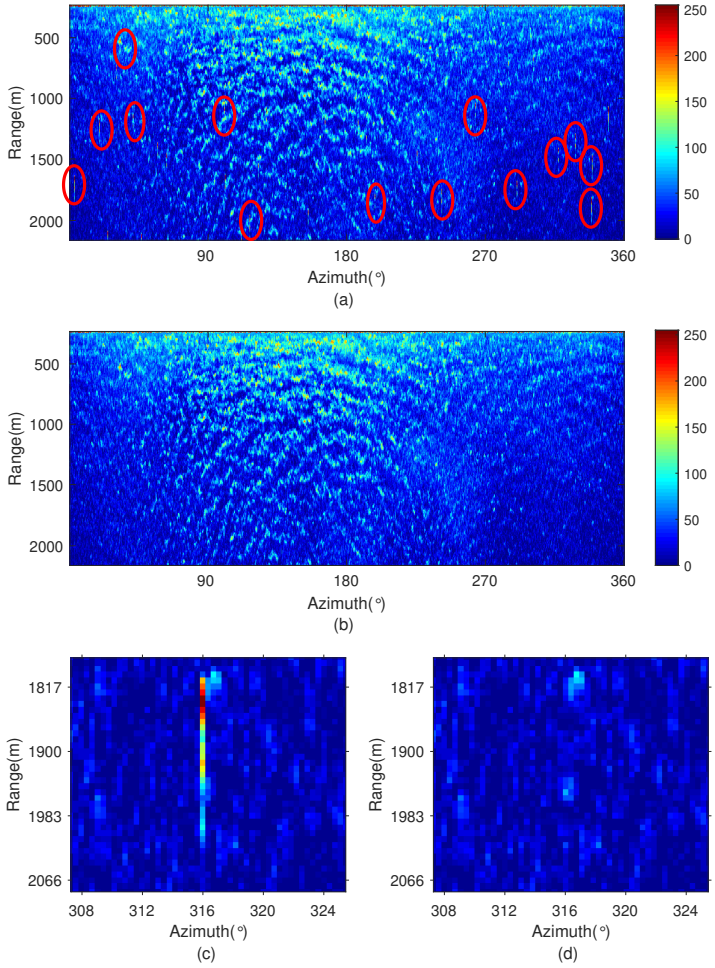

Figure 2.2: (a) A raw radar image with some examples of radial noise lines outlined in red. (b) Image in (a) after noise line removal. (c) A closer look of a noise line in (a). (d) The region in (c) after noise line removal.

As the size of radar images may vary due to the difference in the azimuth coverages of the two radar systems, the extracted feature vectors consisting of bin values are therefore normalized, representing the discrete probability density function (PDF) of the pixel intensities. To guarantee the applicability of the model under different wind and rain conditions, samples for training include a wide wind speed range with different rain rates. Table 2.2 shows the number of images for training and testing respectively. The acquisition time of the whole data as well as the periods chosen for training are listed in Table 2.3. The Decca and Furuno data used for training are also indicated by the gray shadow in Fig. 2.3(a) and Fig. 2.4(a) respectively. Whether an image is labeled as rain-free or rain-contaminated

is determined by the simultaneous rain rate, which is the moving-average of raw rain rate data over 5 minutes. An image will be labeled as rain-contaminated if simultaneous rain rate is nonzero. Otherwise, it belongs to the rain-free type. In this study, the Statistics and Machine Learning Toolbox in MATLAB is utilized to train and cross-validates the detection models for Decca and Furuno data respectively. The toolbox supports mapping the predictor data using kernel functions, and supports hyperparameter optimization via the sequential minimal optimization (SMO) algorithm for minimizing the objective function in Eq. (2.6).

2.2.2 Results and Discussions

Fig. 2.3(a) and Fig. 2.4(a) present the full detection results of the Decca and Furuno data respectively. Data that are not in the gray shadows of Fig. 2.3 and Fig. 2.4 are all used to calculate the detection accuracy of the model. Results show that for Decca data, the detection accuracy of all testing samples is 98.4%, with 98.7% and 96.4% for rain-free and rain-contaminated samples, respectively. As for Furuno data, the corresponding statistics are 99.5%, 100% and 97.8%. Therefore, in general, the proposed method is able to achieve high accuracy for both datasets, with the accuracy of Furuno data slightly higher than Decca data. In addition, for both datasets, the detection accuracy of rain-free data (i.e., specificity) is slightly higher than that of rain-contaminated data (i.e., sensitivity). It should be noted that the results in Fig. 2.3(a) and Fig. 2.4(a) are acquired using 19 bins ranging from 0 to 255. In order to investigate the effect of the number of bins on detection accuracy, detection models using different numbers of bins are trained and tested. Corresponding results with total numbers of bins ranging from 10 to 32 are shown in Fig. 2.5. According to Fig. 2.5(a), the total number of bins that reaches the highest detection accuracy (98.4%) among all Decca testing data is 19. While for Furuno data, the detection accuracy is even higher (99.7%), with the total number of bins as 25, as shown in Fig. 2.5(b). Although the total number

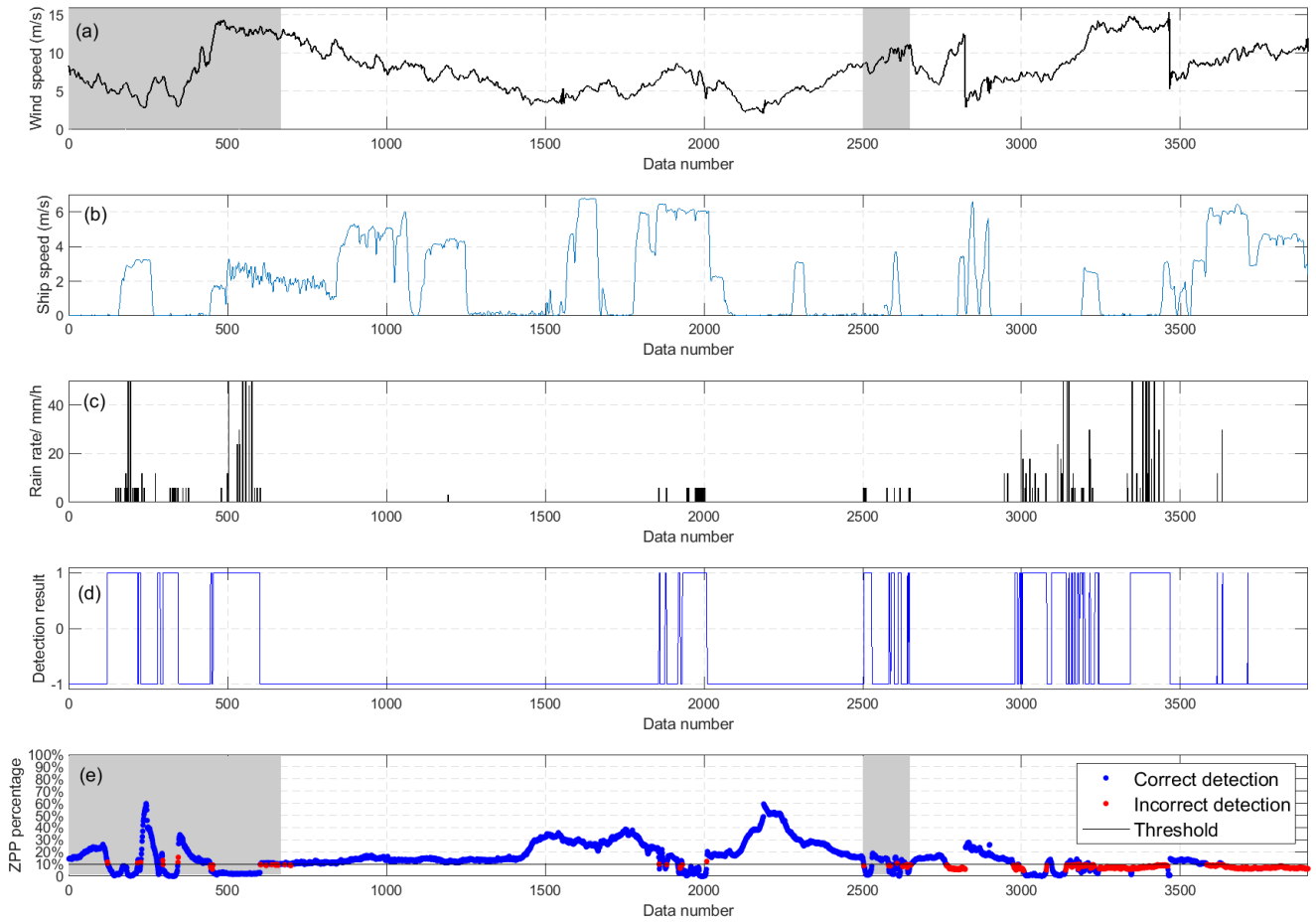

Figure 2.3: Decca data overview and experimental results. (a) Simultaneous wind speed data of radar images measured by anemometers. Training samples are in gray shadows. (b) Simultaneous ship speed data. (c) Simultaneous raw rain rate data of radar images measured by rain gauges employed on the ship. (d) Detection results using the SVM-based method. Rain free and rain-contaminated images are labeled as -1 and +1, respectively. (e) ZPP values of radar data, with a threshold of 9.5% determined by training samples in gray shadow. Data that are correctly or incorrectly detected using ZPP threshold are indicated in blue and red dots respectively.

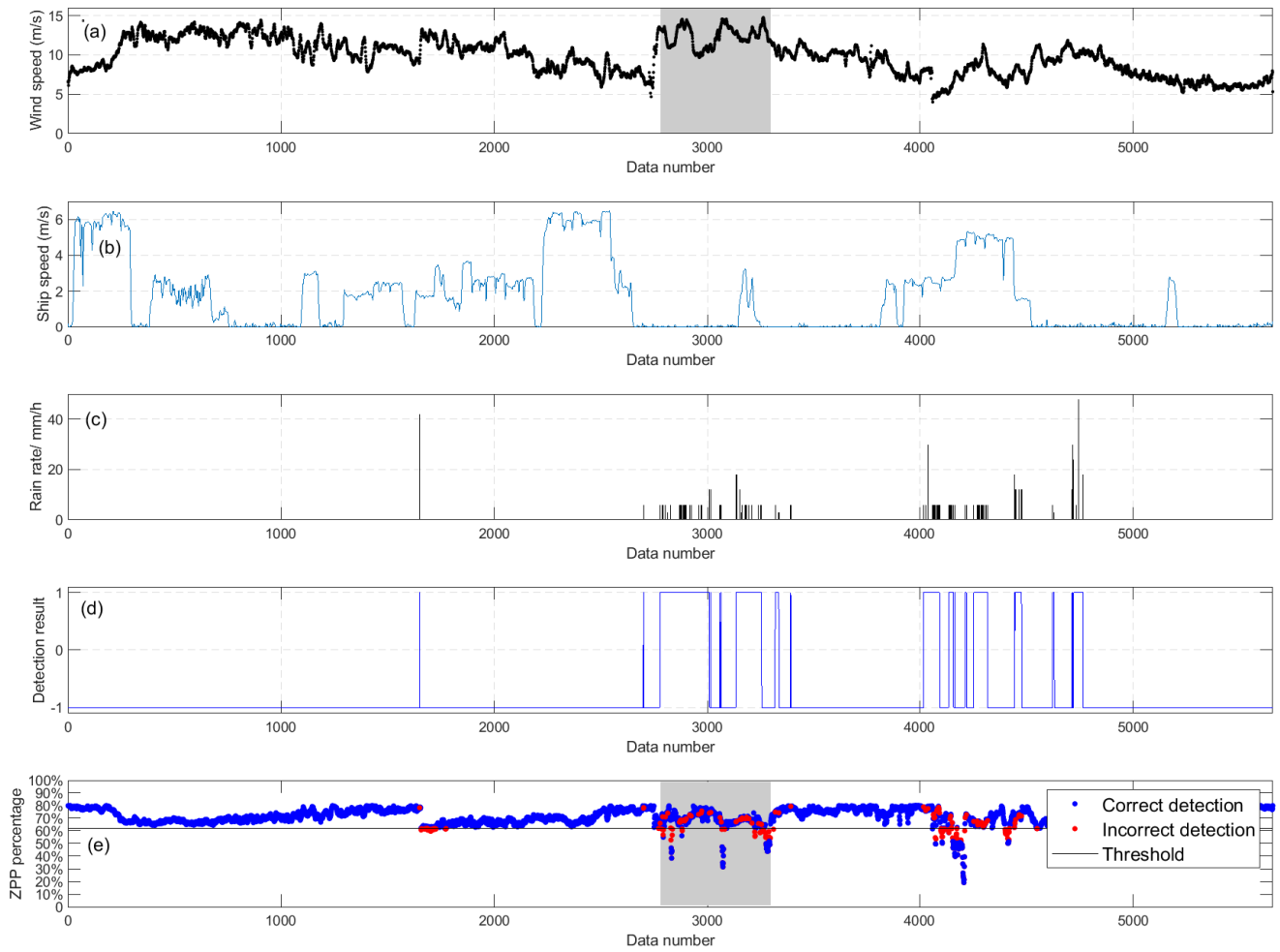

Figure 2.4: Furuno data overview and experimental results. (a) Simultaneous wind speed data of radar images measured by anemometers. Training samples are in gray shadow. Data that are correctly or incorrectly detected using the proposed method are indicated in blue and red dots respectively. (b) Simultaneous ship speed data. (c) Simultaneous raw rain rate data of radar images measured by rain gauges employed on the ship. (d) Detection results using the SVM-based method. Rain free and rain-contaminated images are labeled as -1 and +1, respectively. (e) ZPP values of radar data, with a threshold of 62.0% determined by training samples in gray shadow. Data that are correctly or incorrectly detected using ZPP threshold are indicated in blue and red dots respectively.

of bins that produce the highest detection accuracy are different for two radar systems, the variance of detection accuracy among all testing data using different total numbers of bins are only 0.2% and 0.03% for Decca and Furuno data. In addition, as presented by the red dots in Fig. 2.5, the detection accuracy of rain-contaminated data stays relatively stable, with variances of 0.1% and 1.0% for Decca and Furuno rain-contaminated data, respectively (also presented in Table 2.4). Hence, it can be concluded that the detection accuracy is not significantly affected by the total number of bins.

The ZPP threshold method is also applied to the data in order to compare with the proposed method. As the datasets used in this study are identical to those in [60], pixels with gray scale intensity lower than 5 are referred as zero intensity. The optimal ZPP thresholds are defined as the values that gain the highest detection accuracy among all training samples. Training results using various values for thresholds are shown in Fig. 2.6. According to Fig. 2.6, 0.095 and 0.62 are determined as the optimal ZPP thresholds for Decca and Furuno data, respectively. Detection results of both datasets are presented in Fig. 2.3(c) and Fig. 2.4(c), respectively. Although optimal thresholds reach high accuracy (97.5% for Decca data and 94.6% for Furuno data) among training samples, by applying the threshold values to the testing samples, it is found that the detection accuracy of Decca data decreases significantly. This is due to the fact that there are many fewer training samples than testing samples, and the optimal ZPP thresholds may no longer be optimal for the testing samples. As shown in Table 2.4, compared to the results obtained using the ZPP threshold method, the proposed method improves the detection accuracy by 16.5% and 4.2% for the Decca and Furuno radars among all testing data. It should also be noticed that significant difference exists between the detection accuracy of rain-free and rain-contaminated data. For Decca data, the rain-free data detection accuracy is 18.5% lower than rain-contaminated data, while it is 8.1% higher for Furuno data. This is due to the fact that when using the ZPP threshold method, a trade-off exists between the detection accuracy of rain-free and rain-contaminated image. As Fig. 2.6 shows, while a relatively low ZPP threshold value may classify all rain-free images correctly, many rain-contaminated images are misclassified, causing the failure

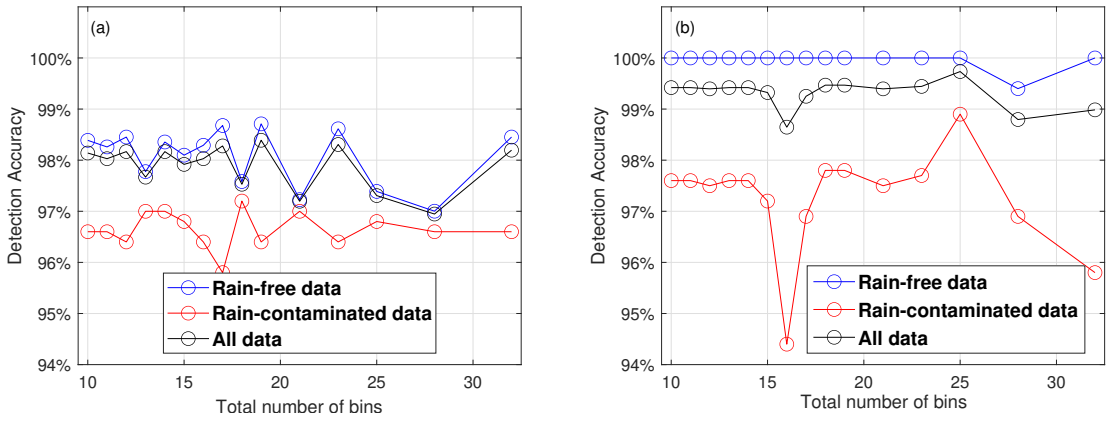

Figure 2.5: SVM-based method detection results using different total numbers of bins. The detection accuracy of all testing data are indicated using black lines. Accuracy of detecting rain-free and rain-contaminated images are shown in blue and red lines respectively. (a) Decca data results. (b) Furuno data results.

of detecting simultaneous rain. In contrast, a high ZPP tends to negatively affects the classification accuracy of rain-free images. Thus, the high detection accuracy of Decca rain-contaminated data using the ZPP threshold method (97.8%) is due to a ZPP value higher than 0.095, which also results in misclassification of many rain-free data. In contrast, for the proposed method, the difference between rain-free and rain-contaminated data classification accuracy is much smaller. As presented in Table 2.4, the differences between the detection accuracy of rain-free and rain-contaminated images are only 2.3% and 1.1% for the Decca and Furuno data. Therefore, compared to the ZPP threshold method, the proposed method is proved to be more accurate and robust, especially when the number of training samples is limited. In sum, the fundamental reason why the proposed method performs better is that it extracts more statistical information from the radar image and employs a more much superior classification scheme.

Although the proposed method is able to achieve high accuracy in general, the detection accuracy for some parts of the data needs further improvement. For example, as shown in Fig. 2.3(d), most of the incorrect detection happen after data number 3000. By observing the simultaneous non-directional wave spectrum generated by the wave buoy deployed around 10 km away from the radar, it is found that while the area was dominated by wind

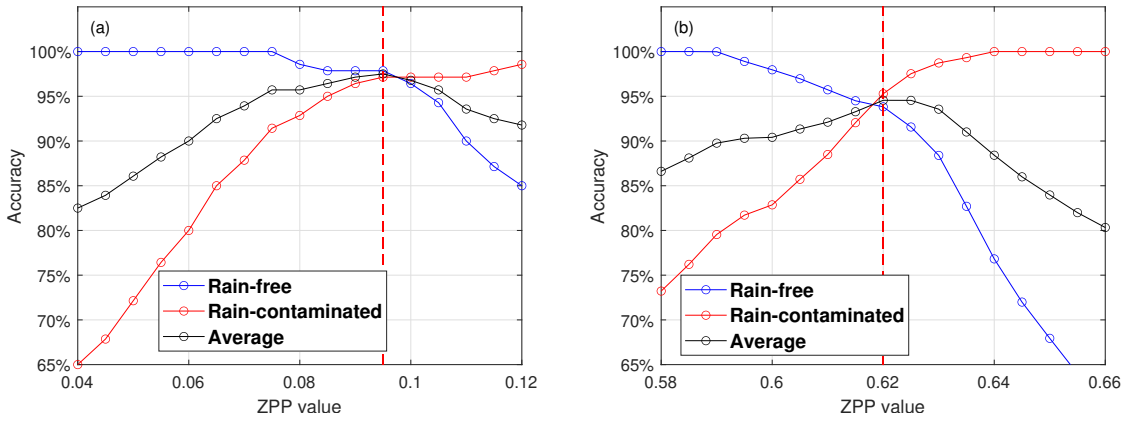

Figure 2.6: ZPP threshold determination using training samples. Accuracy of detecting rain-free and rain-contaminated data are shown in blue and red lines respectively. The optimal ZPP thresholds are defined as the values with the highest average accuracy (mean value of blue and red dots). (a) Decca training results, with optimal ZPP threshold as 0.095. (a) Furuno training results, with optimal ZPP threshold as 0.62.

Table 2.4: Comparison of the results using SVM-based and ZPP-based methods

	SVM						ZPP		
	All data		Rain-free data		Rain data		All data	Rain-free data	Rain data
Radar system	Accuracy	Variance	Specificity	Variance	Sensitivity	Variance	Accuracy	Specificity	Sensitivity
Decca radar	98.4%	0.2%	98.7%	0.3%	96.4%	0.1%	81.9%	79.3%	97.8%
Furuno radar	99.7%	0.03%	100%	0.02%	98.9%	1.0%	95.5%	95.9%	87.8%

waves in most time periods, swell was dominant from around 16:00 to 17:30, December 04. Since a number of images collected during this period are incorrectly detected, the accuracy of the detection model under swell-dominated sea should be further studied and improved. This might be associated with little training data being used with swell-dominated seas. Nevertheless, as the ZPP threshold method misclassifies most of the data collected during this period, the proposed method has already improved the detection accuracy significantly.

2.3 Chapter Summary

In this paper, a SVM-based method is presented to detect rain by accurately classifying X-band marine radar into rain-free and rain-contaminated types. The difference of the

histogram pattern between rain-free and rain-contaminated images is first observed and utilized for feature extraction. After data preprocessing, a feature vector consisting of normalized bin values of the histogram is extracted from each image. The detection model is then trained using SVMs, which can classify image data obtained under rainless and rainy conditions. The SMO algorithm is applied to optimize the hyperparameters in the SVM model.

The method is validated using images collected from two radars during a sea trial in the North Atlantic Ocean. The simultaneous rain rate data collected by rain gauges are used as ground truth to determine whether an image is contaminated by rain. Results demonstrate that compared to the ZPP threshold method, the proposed method further improves the detection accuracy for both radar systems. Also, the SVM-based detection model is proved to be more robust since the accuracy does not vary significantly when choosing different values of total number of bins. In addition, due to strong approximation and generalization ability of SVMs, the model works well even when a relatively small number of samples are used to train the SVM. In contrast, for the ZPP threshold method, a slight change in the ZPP value may significantly change the detection accuracy when limited training samples are available.

Chapter 3

Pixel-based Rain-contaminated Region Identification

Although the SVM-based method illustrated in [78] is more accurate and robust than the threshold-based methods for rain-contaminated image detection, it still cannot locate regions that are contaminated by rain echoes in a single image [79]. For rain-contaminated images, many of them are found to be only partially contaminated by rain echoes. In particular, some regions remain unaffected or less affected by rain echoes, especially in areas that are located in or close to upwind direction. It is evident that the image correction techniques should only be applied to those rain-contaminated regions. In addition, rain-contaminated region identification using marine radars could provide rain observations for atmospheric and oceanographic research activities, such as ocean salinity prediction in rainy regions [80]. In [81], artificial neural network is applied for rainfall estimation from both horizontally- and vertically-polarized meteorological radar data. In this chapter, two methods (SOM-based method and SegNet-based method) are presented to identify rain-contaminated regions in radar images based on unsupervised and supervised learning techniques in Sections 3.1 and 3.2, respectively.

3.1 SOM-based Model for Rain-contaminated Region Identification

It has been found in [57] that the texture of rain echoes is distinctly different from that of rain-free regions. Hence, it is possible to segment rain-free and rain-contaminated pixels using texture segmentation techniques. Also, due to radar system calibration errors or too-low wind speed, some images may contain large black regions with little or no wave signature. These are referred to as low-backscatter regions and should be recognized and discarded [60]. Therefore, a more sophisticated data quality control method that can recognize rain-contaminated and low-backscatter regions should be developed. In this way, the rain-free and rain-contaminated regions can be processed differently and measurement results could be further improved. Generally, for texture segmentation using signal processing approaches, the textured image is input to a linear transform, filter, or filter bank followed by some energy measure [82]. Numerous filtering techniques have been applied for texture segmentation, such as Gabor filtering [83], discrete cosine transform (DCT) [84], and discrete wavelet transform (DWT) [85]. The filtering results which consist of multiple feature images can be combined as a feature vector for each pixel. Since there is no ground truth data concerning whether a certain pixel is rain-contaminated, unsupervised clustering algorithms such as k-means and self organizing map (SOM) can be introduced to classify image pixels into rain-free, rain-contaminated or low-backscatter types automatically. In fact, the SOM has been successfully applied in ocean remote sensing. For example, in [86], SOMs were used to extract the patterns of ocean current variability from a joint HF radar and acoustic Doppler current profiler (ADCP) data set. A review of SOM applications in oceanography and meteorology was provided by Liu and Weisberg [87].

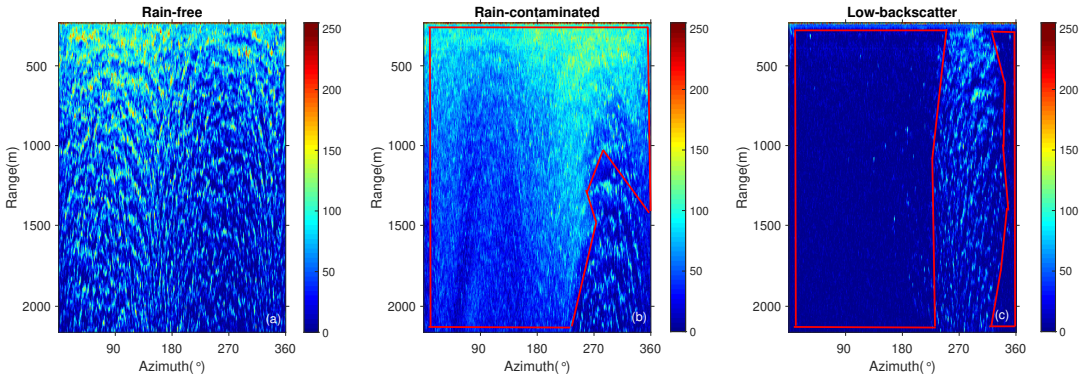

Figure 3.1: Examples of Decca radar images used in this study. (a) A rain-free Decca radar image collected at 23:43, November 26. Wave signatures can be clearly observed. (b) A rain-contaminated Decca radar image collected at 5:48, November 27, with rain-contaminated regions outlined in red. Wave signatures are blurred by rain in those regions. (c) A Decca radar image collected at 4:10, November 27, with large parts of low-backscatter regions outlined in red. Wave signatures can hardly be observed in those regions.

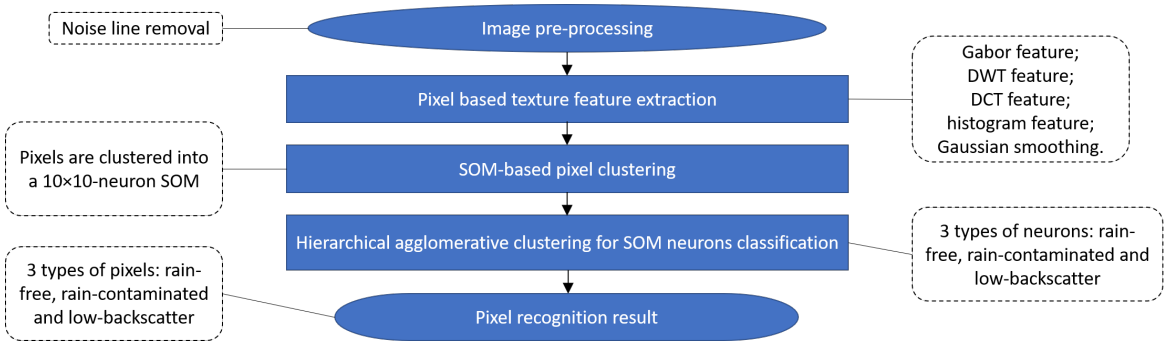

Figure 3.2: The proposed scheme to recognize rain-free, rain-contaminated, and low-backscatter pixels in marine radar images.

3.1.1 Framework of the Proposed Method

The Decca radar data introduced in Subsection 2.1.2 are used for the training and testing of the proposed model. In addition, the same data preprocessing step as described in Subsection 2.2.1 is undertaken. Fig. 3.1 shows examples of the polar or B-scan radar images obtained from the dataset. The distinct difference between rain-free, rain-contaminated, and low-backscatter regions in texture can be clearly observed. Fig. 3.2 shows the framework of the proposed method for rain and low-backscatter region recognition. After preprocessing,

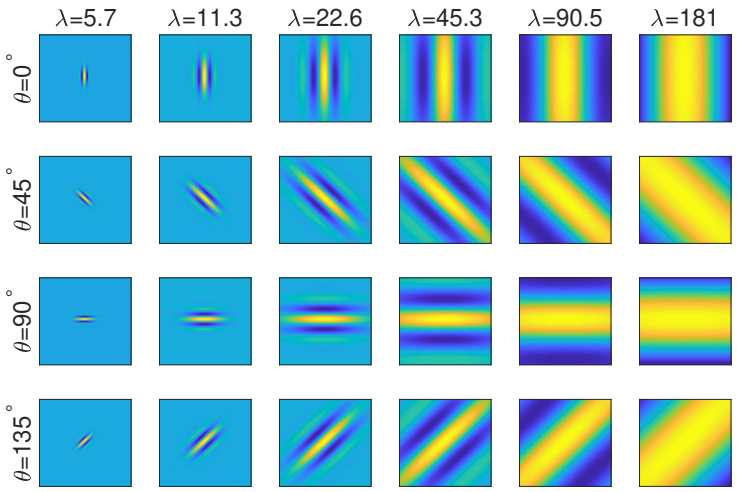

Figure 3.3: The designed Gabor filter bank for the study, including 4 orientations and 6 wavelengths.

texture features are extracted from radar images using multiple filtering techniques and combined into feature vectors. Feature vectors are fed into a 10×10-neuron SOM, which clusters pixels into 100 units (neurons). Those neurons are then grouped into three types (rain-free, rain-contaminated, and low-backscatter) using hierarchical agglomerative clustering.

3.1.2 Texture Feature Extraction

In previous studies numerous features have been extracted for texture analysis using different filtering techniques or different statistics of the image. Below are four types of features considered in this study.

Gabor Feature Extraction

In image processing, a Gabor filter is a Gaussian kernel function modulated by a sinusoidal plane wave, which accounts for the parallel stripes as shown in Fig. 3.3. It has been widely used for texture analysis due to its similarity with the human visual system in frequency and orientation representations. In the spatial domain, a Gabor

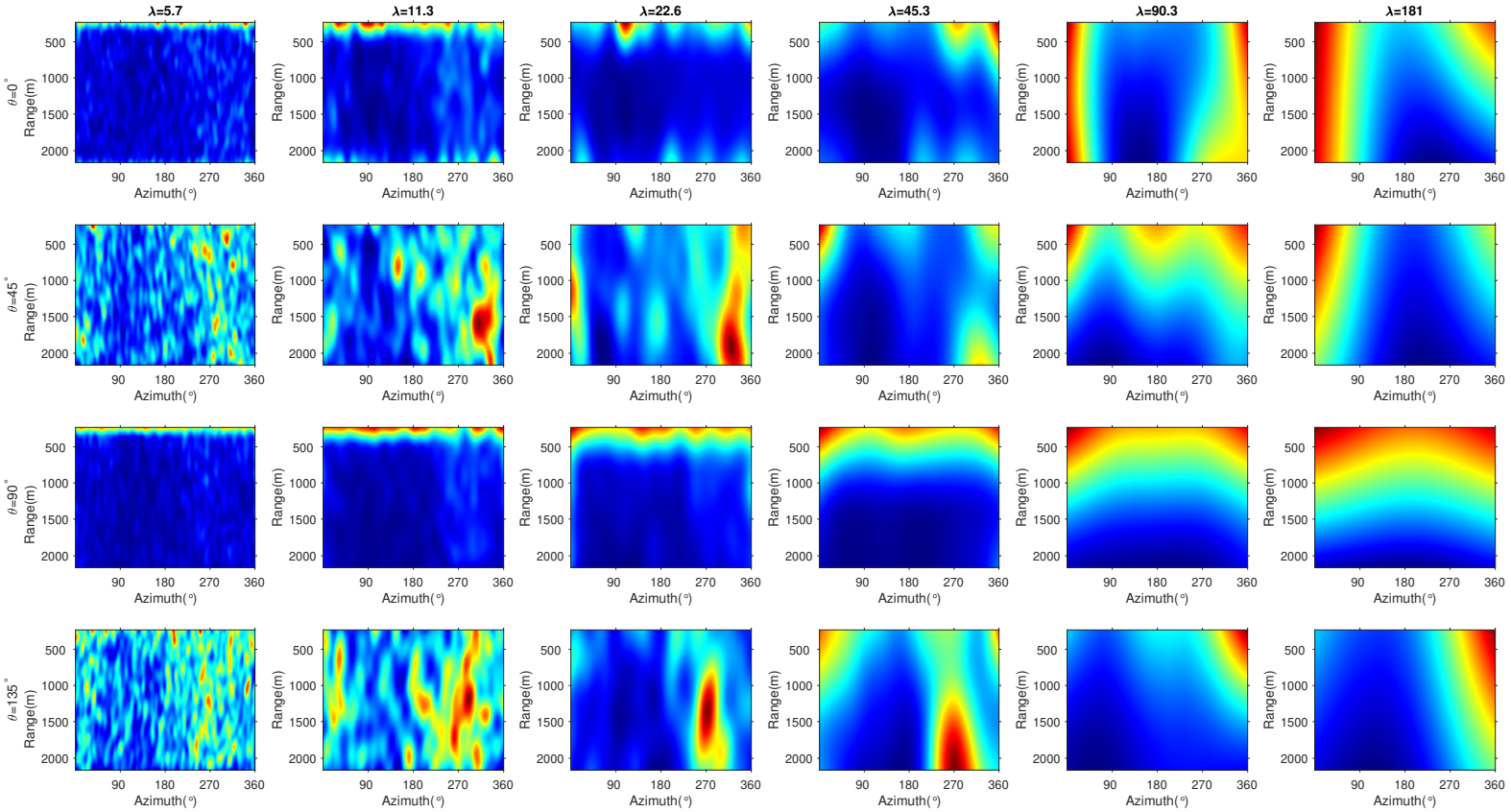

Figure 3.4: 24 Gabor feature images produced from Fig. 3.1(b). Each feature image corresponds to the filtering output of the Gabor filter in the same grid position in Fig. 3.3.

filter is given by [83]

$$G_f(m,n) = \exp\left\{-\frac{1}{2}\left[\frac{m'^2}{\sigma_m^2} + \frac{n'^2}{\sigma_n^2}\right]\right\}\cos(2\pi\frac{m}{\lambda_g} + \phi_g), \tag{3.1}$$

where σ_m and σ_n characterize the spatial extent and bandwidth of the filter, and λ_g and ϕ_g denote the wavelength and phase of the sinusoidal plane wave along the x-axis respectively. In addition, it should be noted in Eq. (3.1), $m' = m\cos\theta_g + n\sin\theta_g$ and $n' = -m\sin\theta_g + n\cos\theta_g$, where θ_g represents the orientation normal to the parallel stripes of a Gabor function. In this study, a bank of Gabor filters with different wavelengths (λ_g) and orientations (θ_g) are designed to extract features from radar images. The selection of wavelengths is dependent on the size of the radar image. Since all radar images have been resized into 256×1024, six wavelengths are selected using the criterion proposed by Jain and Farrokhnia [83], the values of which are shown in Fig. 3.3. Meanwhile, four values of orientations (i.e., $\theta_g = 0, \pi/4, \pi/2, 3\pi/4$) are used. Therefore, a total of 24 filters are generated for filtering the radar image. Fig. 3.4 shows the Gabor filtering results of Fig. 3.1(b).

Wavelet Feature Extraction Using DWT

DWT captures both frequency and location information of a signal, and it has been widely used for various applications in image processing, such as denoising, compression and segmentation. In this work, the DWT filtering output image is further processed and utilized to extract texture features. Here, $I[m,n]$ and (m,n) represent the intensity of the pixel and its coordinate in the radar image, respectively. First, the 1-D DWT is conducted on each row of the radar image, which can be expressed as [88]

$$v_{1,L}[m,n] = \sum_{k=0}^{K-1} I[m, 2n-k]\, g_l[k], \tag{3.2}$$

$$v_{1,H}[m,n] = \sum_{k=0}^{K-1} I[m, 2n-k]\, g_h[k], \tag{3.3}$$

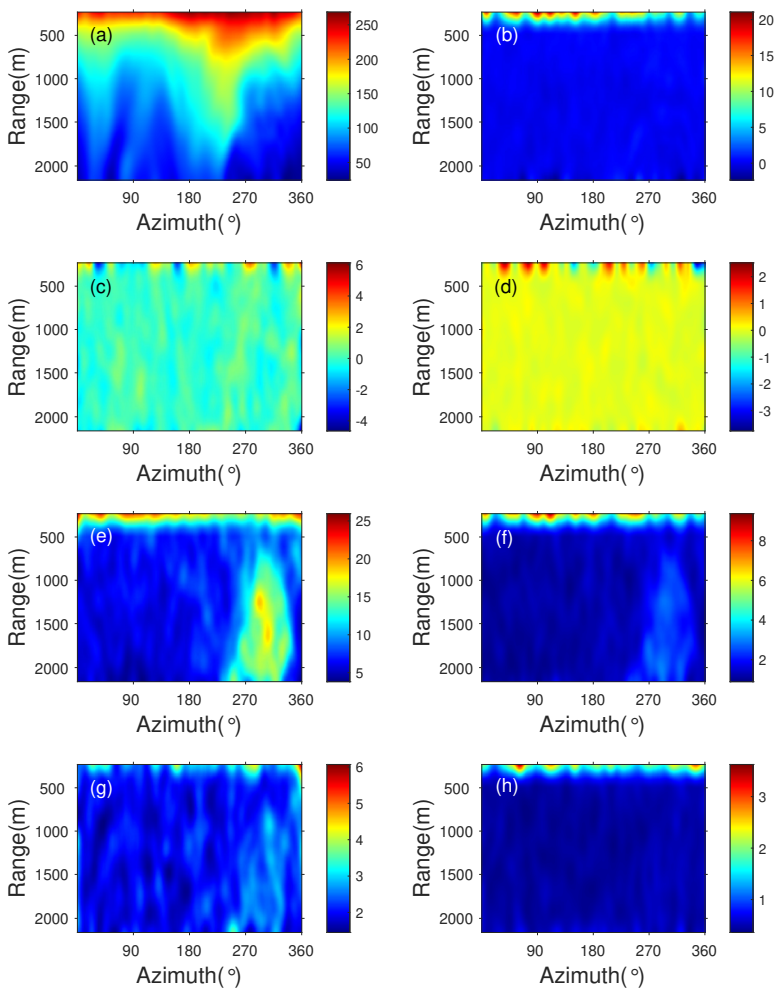

Figure 3.5: DWT feature images of Fig. 3.1(b). (a), (b), (c) and (d) correspond to the LL, LH, HL and HH channel ,respectively. (e), (f), (g) and (h) correspond to the local standard deviation images of (a), (b), (c) and (d), respectively.

where $g_l[k]$ and $g_h[k]$ are a low pass and high pass filter, respectively. Hence, $v_{1,L}$ and $v_{1,H}$ are the horizontal low frequency component (L) and high frequency component (H), respectively. In this study, the Haar wavelet [89] is selected to construct $g_l[k]$ and $g_h[k]$. Next, the 1-D DWT is applied to each column of the previous filtering

outputs (i.e., L and H) as

$$I_{1,LL}[m,n] = \sum_{k=0}^{K-1} v_{1,L}[2m-k,n]\, g_l[k], \tag{3.4}$$

$$I_{1,HL}[m,n] = \sum_{k=0}^{K-1} v_{1,L}[2m-k,n]\, g_h[k], \tag{3.5}$$

$$I_{1,LH}[m,n] = \sum_{k=0}^{K-1} v_{1,H}[2m-k,n]\, g_l[k], \tag{3.6}$$

$$I_{1,HH}[m,n] = \sum_{k=0}^{K-1} v_{1,H}[2m-k,n]\, g_h[k]. \tag{3.7}$$

As a result, the original radar image is decomposed into four different channels, namely the low-low (LL), low-high (LH), high-low (HL) and high-high (HH) channels. The two-step 1-D DWT mentioned above is actually equivalent to a 2-D DWT. In this work, the Haar wavelet is used for image decomposition due to its simplicity. Further decomposition of the image can be conducted on the LL ouput. Nevertheless, in this work only a level-1 decomposition is conducted in order to reduce calculation time.

In order to generate feature images, the four channels are first upsampled to the same size as the original radar image through bilinear interpolation. Then, the standard deviation filtering is performed on each output channel image, producing the local standard deviation image. Each pixel of a local standard deviation image contains the standard deviation value of the 15-by-15 neighborhood around the corresponding pixel in an output channel image. Hence, as shown in Fig. 3.5, for each radar image, its DWT features contain 8 output images (4 channels and their corresponding local standard deviation images).

DCT-based Image Feature Extraction

The DCT is often used in image compression due to its strong "energy compaction" property [90]. In [84], it was found that texture features can be extracted using the

DCT. The detailed steps are as follows. First, given three one-dimensional filter masks $u_1 = \{1, 1, 1\}^T$, $u_2 = \{1, 0, -1\}^T$, $u_3 = \{1, -2, 1\}^T$, a set of nine 3×3 orthogonal DCT masks can be produced with

$$\mathbf{D}_l(i, j) = u_a(i)\, u_b(j), \tag{3.8}$$

where $l = a + 3(b - 1)$ for $1 \leq a, b \leq 3$ and $\mathbf{D}_l(i, j)$ stands for the value of a certain point in the l^{th} DCT mask. The mask with low-pass property is excluded and the radar image is filtered by the eight masks left, which can be expressed as:

$$Y_l(m, n) = \sum_{i,j=0}^{2} \mathbf{D}_l(i + 1, j + 1)\, I(m - i, n - j), \tag{3.9}$$

The DCT texture feature images can be generated from the local variance of $Y_l(\cdot, \cdot)$, which is calculated as:

$$Y_{var,l}(m, n) = \frac{1}{W^2} \sum_{p,q=-\frac{(W-1)}{2}}^{(W-1)/2} \left[Y_l(m - p, n - q) - \mu_l(m, n)\right]^2, \tag{3.10}$$

where W is an odd number denoting the length of a $W \times W$ (31×31 in this work) moving window. $\mu_l(m, n)$ represent the local Gaussian smoothing of $Y_l(m, n)$ over the sliding window. The DCT feature images of Fig. 3.1(b) are shown in Fig. 3.6.

Local Histogram Feature Extraction

It has been observed in [91] that under similar wind speeds, the pixel intensity distribution between rain-free and rain-contaminated image are distinctly different. Fig. 3.7 shows that for a radar image partially contaminated by rain, the histogram patterns of rain-free and rain-contaminated regions are different as well. Therefore, in order to extract regional histogram feature, a 31×31 moving window is applied. For a certain pixel, its histogram features are calculated from the pixels within the window

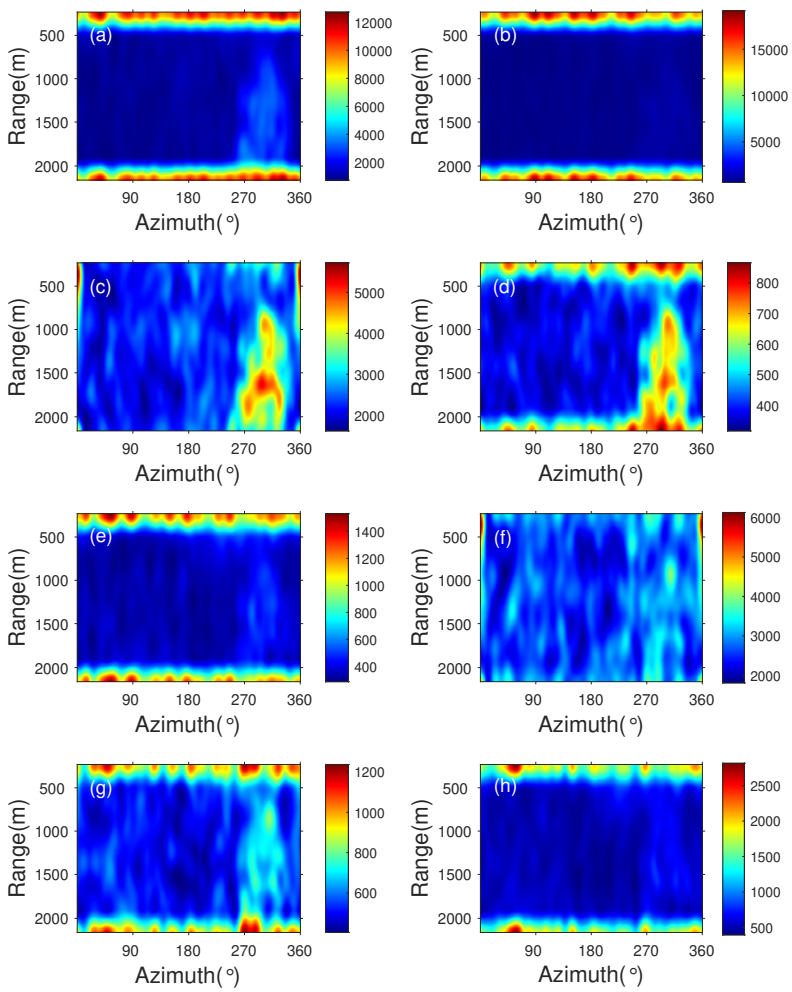

Figure 3.6: DCT feature images of Fig. 3.1(b). (a), (b), (c), (d), (e), (f), (g) and (h) correspond to $y_{var,1}, y_{var,2}, ..., y_{var,8}$ in Eq. (3.10), respectively.

that centers at this pixel. The histogram features for each pixel consist of the normalized bin values ranging from grayscale intensity 0 to 150. Since each bin contains 10 pixel intensities, a total of 15 feature images will be generated from each radar image.

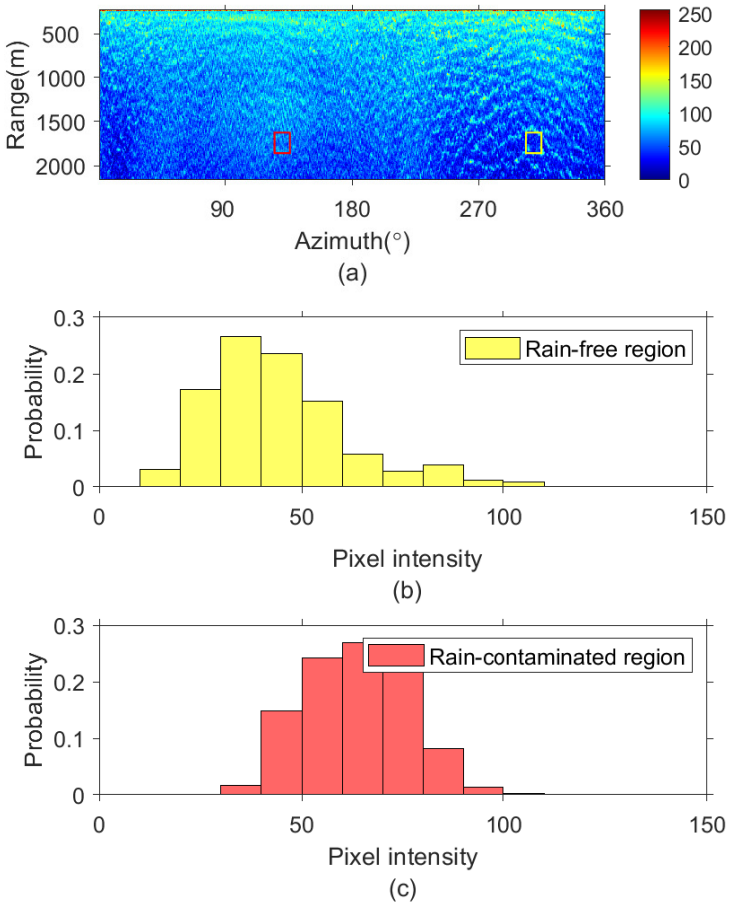

Figure 3.7: (a) A partially rain-contaminated image. (b) Histogram distribution of a rain-free region (i.e., pixels within the yellow box) in (a). (c) Histogram distribution of a rain-contaminated region (i.e., pixels within the red box) in (a).

3.1.3 SOM for Pixel Mapping

An SOM is a type of ANN that is trained using an unsupervised learning algorithm to produce a low-dimensional (especially two-dimensional), discretized representation of the input space of the training samples, called a map. Hence, it can also be regarded as a method to reduce dimensionality. The SOM differs from other ANNs since it applies competitive learning instead of error-correction learning (e.g. backpropagation with gradient descent). In addition, it uses a neighborhood function to preserve the

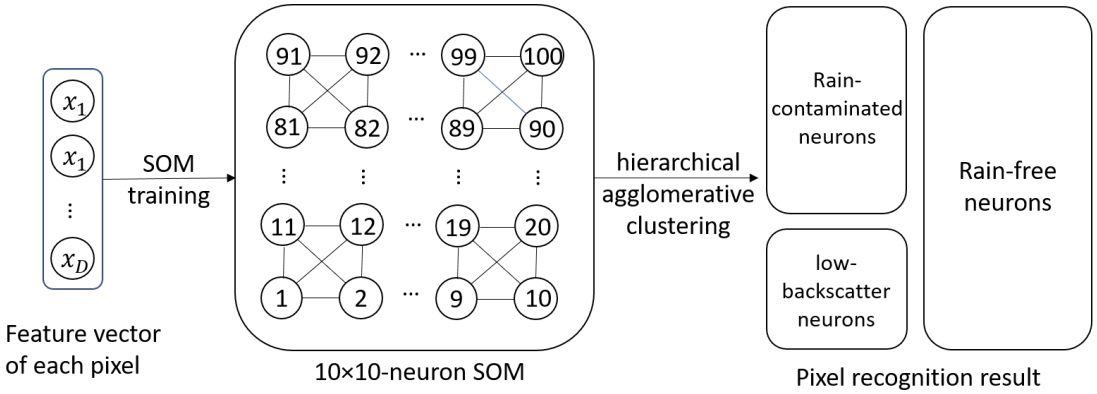

Figure 3.8: The proposed two-level clustering scheme to classify rain-free, rain-contaminated, and low-backscatter pixels based on SOM and hierarchical agglomerative clustering.

topological properties of the input space. The main advantage of using an SOM is that the reduction of dimensionality and grid clustering make it easy to observe similarities in the feature vectors. Compared to k-means clustering, the SOM is less sensitive to initialization and thus provides a more robust learning.

In order to design an SOM suitable for the study, the size of the map (i.e., the number of the neurons) should be determined. However, there is no theoretical principle to determine the optimum number of neurons in the output layer of an SOM [92]. In this study, although all the pixels should be finally clustered into 3 types (rain-free, rain-contaminated, and low-backscatter), the number of neurons should be much greater than 3. This is because the texture of radar images is inhomogeneous in space. In addition, the texture of rain-contaminated regions in different images may look different from each other due to different rain rates or wind speeds. Therefore, in this study a two-level clustering scheme is proposed, as shown in Fig. 3.8. First, pixels are mapped into a relatively large number of prototypes using a 10×10-neuron SOM. Those prototypes (neurons) are then grouped into 3 clusters using the hierarchical agglomerative clustering introduced in [93]. The structure of an SOM consists of three parts: input vector, weight vectors, and neurons. The steps of training an SOM are described in the following paragraphs.

Initialization: First, each feature image extracted from the four types of texture features mentioned above is smoothed using a 15×15 Gaussian filter. Then, data standardization is conducted by shifting features in different dimensions onto a uniform standard scale so that they are of the same measurement unit. For a certain pixel, its texture feature vector is denoted as $X_T = \{x_{Ti}, i = 1, ..., D\}$, where D is the total number of feature images equal to 55. In other words, the feature vector corresponds to the combination of each pixel value (after filtering) generated by the 4 types of filters introduced above.

Suppose the weight value between the i^{th} component of X_T (i.e., x_{Ti}) and the j^{th} neuron is w_{ji}. Therefore, the weight vector of the j^{th} neuron is $w_j = \{w_{ji} : j = 1, ..., N; i = 1, ..., D\}$, where N is the total number of neurons equal to 100. Small random values (ranging from around -0.3 to 0.3) are assigned to w_j to initialize the training process.

Sampling and Matching: For each time of sampling, a new random training sample X_T is selected and its Euclidean distance with the j^{th} neuron can be calculated as:

$$e_j(X_T) = \sum_{i=1}^{D} (x_{Ti} - w_{ji})^2. \tag{3.11}$$

The neuron with the smallest distance will be regarded as the "winning" neuron and denoted as $\Psi(X_T)$. In this way, feature vectors in the continuous input space can be easily mapped into the discrete output space consisting of neurons.

Weight Vector Updating: The winning neuron $\Psi(X_T)$ and its topological neighbors are moved closer to the input vector in the input space by updating their weight vectors. If Δw_{ji} is denoted as the value change of a certain weight vector w_j associated with the neuron j, then

$$\Delta w_{ji} = \eta(t) \cdot T_{j,\Psi(X_T)}(t) \cdot (x_{Ti} - w_{ji}), \tag{3.12}$$

where $\eta(t)$ is the learning rate as a function of the current training iteration t:

$$\eta(t) = \eta_0 \exp(-t/\tau_\eta); \tag{3.13}$$

$T_{j,\Psi(X_T)}(t)$ is a parameter determined by $S^2_{j,\Psi(X_T)}$ (the distance between neuron j and $\Psi(X_T)$) as well as t:

$$T_{j,\Psi(X_T)}(t) = \exp(-S^2_{j,\Psi(X_T)}/2[\sigma(t)]^2); \tag{3.14}$$

$\sigma(t)$ decreases exponentially with t as:

$$\sigma(t) = \sigma_0 \exp(-t/\tau_\sigma). \tag{3.15}$$

It should be noted that η_0, τ_η, σ_0 and τ_σ are all constants determined empirically prior to training. In this study, the SOM is trained with 200 iterations of batch learning in order to ensure that the map is well distributed through the input space. For each batch learning, the entire training dataset is applied to train the SOM. The specifications of some tunable SOM parameters are listed in Table 3.1. All the SOM parameters are identical to those used in [94] except the map size. In this study, a larger map size (10×10) which can obtain more detailed patterns from the radar image and further improve segmentation accuracy is selected.

Table 3.1: Tunable parameters of the designed SOM

Training algorithms	Batch
Activation function	Euclidean distance
Map size	10×10
Neighborhood function	Gaussian

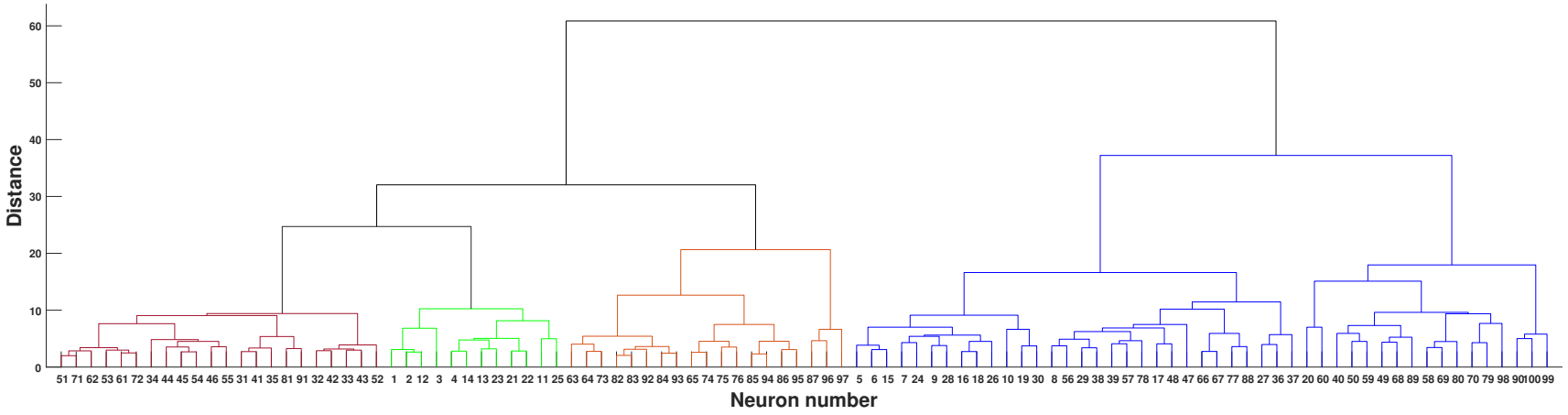

Figure 3.9: The dendrogram showing the clustering results of neurons using hierarchical agglomerative clustering. Branches containing rain-free and low-backscatter neurons are presented in blue and green colors, respectively. Both dark red and orange branches contain rain-contaminated neurons.

3.1.4 Hierarchical Agglomerative Clustering of SOM Neurons

Although the number of neurons is much greater than the types of pixels (rain-free, rain-contaminated, and low-backscatter), the distances between neurons with same type of pixels are smaller than those with different types of pixels. Therefore, a hierarchical clustering tree can be built in order to cluster those neurons using hierarchical agglomerative clustering. According to [93], the steps of agglomerative clustering algorithms are as follows:

1) Initialization: Assign the weight vector of each neuron to its own cluster.

2) Distances computation: Determine distances between all clusters.

3) Merging: Merge the two clusters that are closest to each other.

4) Iteration: Return to step 2 until there is only one cluster left.

There are many metrics for measuring the distance between two clusters. In this study, the Ward's criterion proposed in [95] is used for distance computation. Suppose r and s are two clusters, the distance between r and s is calculated as

$$z_d(r,s) = \sqrt{\frac{2n_r n_s}{n_r + n_s}} \parallel \bar{x}_r - \bar{x}_s \parallel_2, \tag{3.16}$$

where n_r and n_s are the number of neurons in cluster r and s, $\bar{x}_r$ and $\bar{x}_s$ are the centroids of r and s, $\| \cdot \|_2$ denotes the computation of Euclidean distance.

Fig. 3.9 shows the clustering result in a tree structure (dendrogram). By observation, it is found that neurons containing rain-free pixels are grouped into a single large cluster highlighted in blue. Meanwhile, rain-contaminated and low-backscatter neurons are grouped into another large cluster. Specifically, rain-contaminated neurons are highlighted in both orange and red, while low-backscatter neurons are highlighted in green branches. It can also be observed that if two neurons are topological neighbors in the SOM, the distance between them shown in Fig. 3.9 tends to be small as

well.

3.1.5 Result Analysis

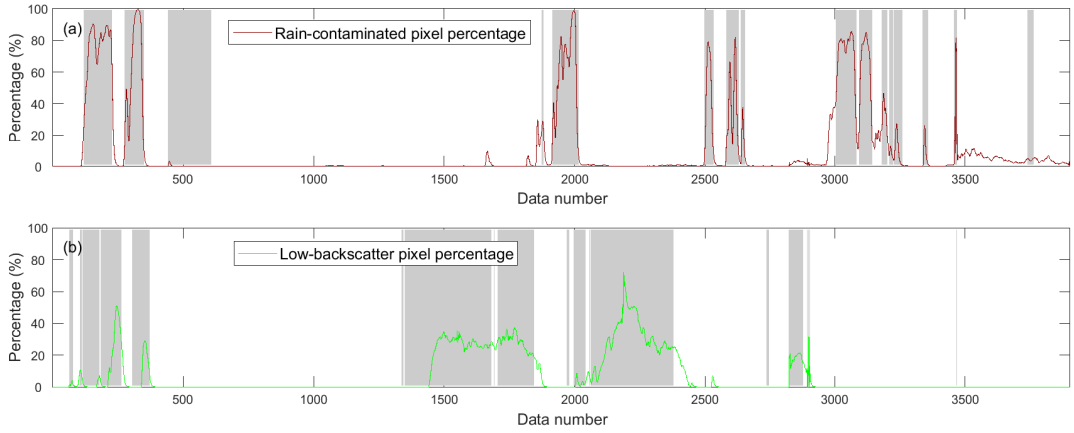

Figure 3.10: (a) Red line: The percentage of rain-contaminated pixels in each image. Images with ZPP lower than 10% are indicated by the grey shadow. (b) Green line: The percentage of low-backscatter pixels in each image. Images with simultaneous wind speed lower than 6 m/s are also indicated by the grey shadow.

In this study, a total of 3902 Decca radar images are available and they are numbered from 1 to 3902 chronologically. Images collected from 23:43, November 26 to 6:51, November 27 (corresponding to data number 1-400) are used to train the SOM-based recognition model while data obtained from 6:52, November 27 to 19:30, December 04 (corresponding to data number 401-3902) are used for validation. It should be noted that pixels located at the first 50 rows corresponding to the range from 240 to 607.5 m in the radar image are eliminated from result analysis because of the strong radar backscatter in the near range, which makes the wave signatures difficult to be observed. Therefore, region recognition results are generated in a range between 615 to 2140 m. The overall identification results are shown in Fig. 3.10. The red and green lines represent the percentage of rain-contaminated and low-backscatter pixels in each radar image, respectively. The gray shadow in Fig. 3.10 corresponds

to images with ZPP lower than 10%, which indicated the presence of rain in previous studies [33, 34] using the same dataset. It can be observed that most of red points with values larger than 10% lie in the gray shadow. On the other hand, images with simultaneous wind speeds lower than 6 m/s are indicated by the gray shadow in Fig. 3.10. Thus, it can be concluded that most low-backscatter regions appear in low-wind-speed radar images.

Identification results of individual images are presented in Figs. 3.11, 3.12 and 3.13. Fig. 3.11 shows identification examples of images obtained under rainy conditions with different rain-contaminated pixel percentages. Fig. 3.11(a) shows an example of one radar image completely contaminated by rain. As shown in Fig. 3.10, images with high rain-contaminated pixel percentages mainly appeared in the middle of the rain events. On the other hand, partially rain-contaminated images such as Fig. 3.11(b) were usually obtained at the beginning or the end of the rain events. It is interesting to see that no rain-contaminated pixels are identified from images with data number ranging from 450 to 600, during which high wind speeds and heavy rainfall were recorded. An example of those figures is displayed in Fig. 3.11(c). As can be seen in Fig. 3.11(c), although rain could cause additional radar backscatter, surface wave signatures were not blurred by rain echoes. Previous studies [33] have found that under high wind speeds (larger than 10 m/s), the dependence of the normalized radar cross section (NRCS) on wind direction may not be significantly affected by rain because the wind effect dominates. Besides, since recent studies concerning wind parameter measurements [33, 34, 36] have applied the same algorithm to high-wind-speed rain cases and rain-free cases, the clustering results in Fig. 3.11(f) are reasonable. Fig. 3.12 shows the rain-contaminated pixel recognition results in partially contaminated images under different wind speeds. It can be observed that the region segmentation results are in good accordance with human perception. As can be observed in Figs. 3.11(e) and 3.12(f), rain-contaminated pixels in orange color are generally bright with

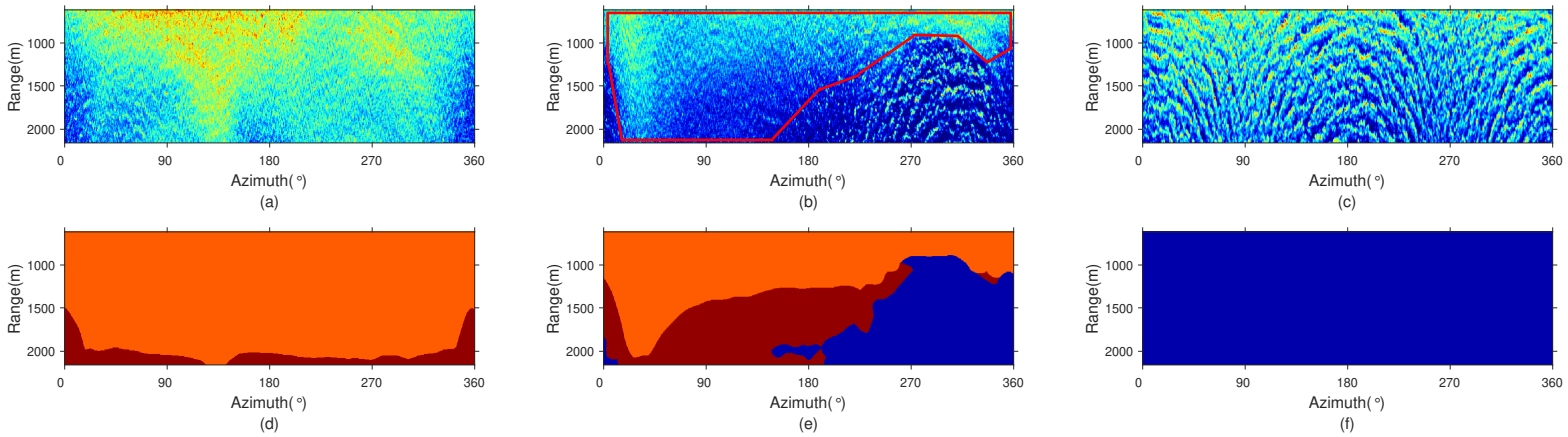

Figure 3.11: Pixel-based recognition results of rain-contaminated regions under different rain conditions. (a) Rain-contaminated radar image collected at 19:13, November 28 (corresponding to data number 1998). Anemometer-measured wind speed is 5.4 m/s. It can be observed that rain echoes contaminate the wave signatures of image completely. (b) A partially rain-contaminated radar image collected at 5:50, November 27 (corresponding to data number 343). Anemometer-measured wind speed is 2.3 m/s. Rain-contaminated regions are outlined in red manually. (c) Radar image collected at 9:36, November 27 (corresponding to data number 540). Anemometer-measured wind speed is 10.9 m/s. Rain gauge-measured rain rate is 24 mm/h. However, rain did not blur wave signatures in the image. (d) Recognition results of (a) using the proposed method. Blue color corresponds to rain-free pixels. Red and orange color represent rain-contaminated pixels in red and orange branches in Fig. 3.9, respectively. (e) Recognition results of (b). (f) Recognition results of (c).

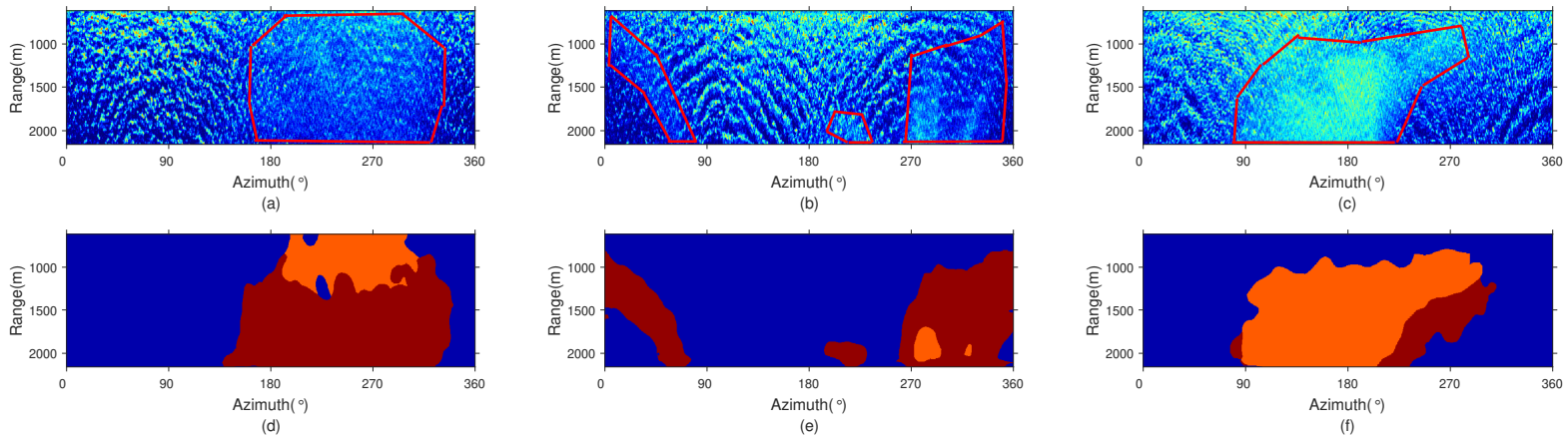

Figure 3.12: Pixel-based recognition results of partially rain-contaminated images under different wind speeds. (a) Rain-contaminated radar image collected at 1:47, November 27 (corresponding to data number 125). Anemometer-measured wind speed is 5.0 m/s. Rain-contaminated regions are outlined in red manually. (b) Rain-contaminated radar image collected at 8:05, November 29. Anemometer-measured wind speed is 10.8 m/s. Rain-contaminated regions are outlined in red manually (corresponding to data number 2591). (c) Rain-contaminated radar image collected at 9:53, December 01 (corresponding to data number 3345). Anemometer-measured wind speed is 15.6 m/s. Rain-contaminated regions are outlined in red manually. (d) Recognition results of (a) using the proposed method. Blue color corresponds to rain-free pixels. Red and orange color represent rain-contaminated pixels in red and orange branches in Fig. 3.9, respectively. (e) Recognition results of (b). (f) Recognition results of (c).

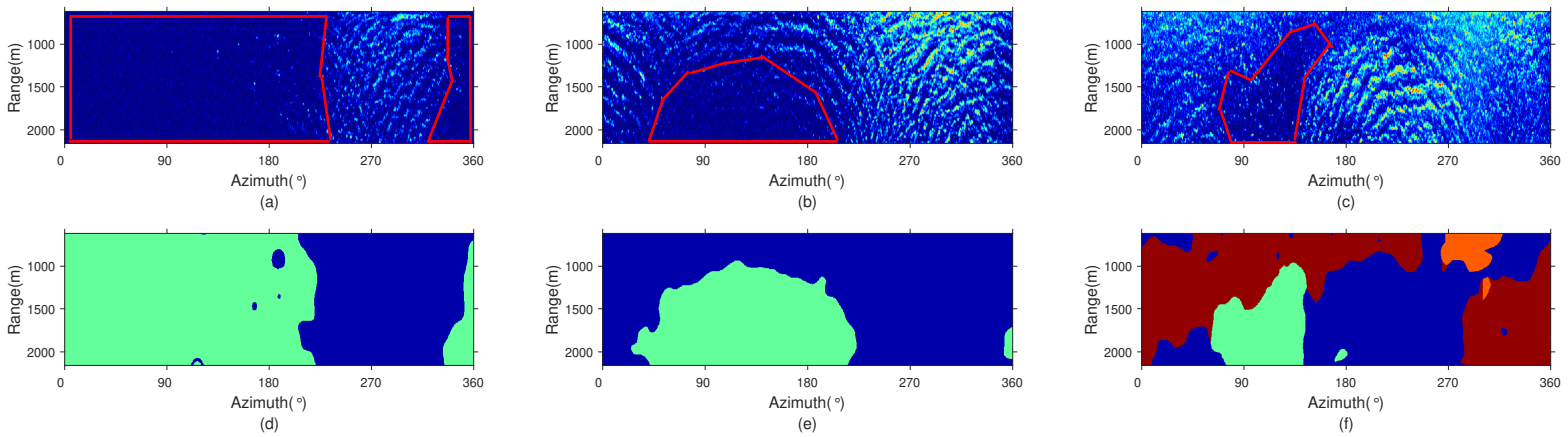

Figure 3.13: Pixel-based recognition results of several images with low-backscatter regions. (a) Radar image collected at 4:10, November 27 with low-backscatter regions outlined in red. Anemometer-measured wind speed is 2.8 m/s (corresponding to data number 245). (b) Radar image collected at 4:24, November 27 with low-backscatter regions outlined in red (corresponding to data number 259). Anemometer-measured wind speed is 5.0 m/s. (c) A rain-contaminated radar image collected at 19:26, November 28 with low-backscatter regions outlined in red (corresponding to data number 2008). Anemometer-measured wind speed is 2.4 m/s. (d) Recognition results of (a) using the proposed method. Blue and green color corresponds to rain-free and low-backscatter pixels, respectively. Red and orange color represent rain-contaminated pixels in red and orange branches in Fig. 3.9, respectively. (e) Recognition results of (b). (f) Recognition results of (c).

high pixel intensities, while rain-contaminated pixels in red color look more like low-backscatter pixels with much lower pixel intensities. That explains why rain-contaminated neurons in red branches have a smaller distance with low-backscatter neurons compared to rain-contaminated neurons in orange branches, as shown in Fig. 3.9. Finally, Fig. 3.13 shows three examples of low-backscatter region identification results, which look satisfactory. It should be noted that under low-wind-speed rain cases, both rain-contaminated and low-backscatter regions may appear in the radar image, as presented in Fig. 3.13(f).

In order to analyze the recognition accuracy quantitatively, the empirical discrepancy method proposed in [96] is adopted, which evaluate segmentation algorithms by comparing the resulting recognized image against a manually-labeled reference image. Due to the large data volume, only images containing rain-contaminated or low-backscatter regions are labeled manually using the Image Labeler toolbox on MATLAB. Those manually-segmented reference images (198 images in total) are regarded as ground-truths. The degree of similarity between the human and machine recognized images determines the performance of the proposed method. A confusion matrix is generated and presented in Fig. 3.14. Each row of the matrix represents the pixels in an actual class (manually-labeled results) while each column represents the pixels in a predicted class (SOM-based clustering results). A row-normalized row summary displays the percentages of correctly and incorrectly clustered pixel for each true class, which is also defined as producer accuracy. As shown in Fig. 3.14, the producer accuracies of rain-free, rain-contaminated, and low-backscatter pixels are 89.5%, 94.2% and 90.6%, respectively. On the other hand, a column-normalized column summary displays the percentages of correctly and incorrectly clustered pixels for each predicted class, which is referred to as user accuracy. Hence, in this study the producer accuracies of rain-free, rain-contaminated, and low-backscatter pixels are 91.6%, 92.5% and 88.7%, respectively. The overall accuracy of the proposed method

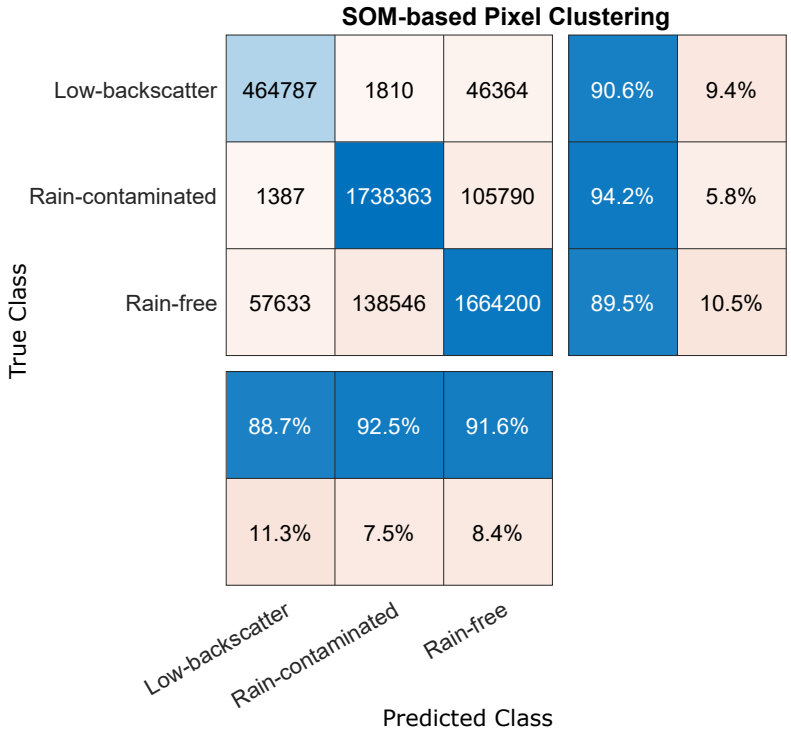

Figure 3.14: Confusion matrix of the clustering results. The true class and predicted class correspond to manually-labeled results and SOM-based clustering results, respectively.

is 92.8%, which is calculated by summing the number of correctly clustered pixel numbers (i.e., pixels located along the diagonal of the confusion matrix) and dividing by the total number of pixels. In order to further evaluate the performance of the proposed method, the Kappa coefficient which measures the agreement between classification and truth values is calculated. A Kappa value of 1 represents perfect agreement, while a value of 0 means no agreement. The Kappa coefficient is computed as follows:

$$\text{Kappa} = \frac{N \cdot \sum_{i=1}^{n} \text{TP}_i - \sum_{i=1}^{n} [(\text{TP}_i + \text{FN}_i)(\text{TP}_i + \text{FP}_i)]}{N^2 - \sum_{i=1}^{n} [(\text{TP}_i + \text{FN}_i)(\text{TP}_i + \text{FP}_i)]}, \tag{3.17}$$

where N is the total number of classified pixels in all testing images. TP_i, FP_i, and FN_i are, respectively, the number of true positives, false positives, and false negatives of the i^{th} class. By calculation, it is found that the Kappa coefficient is 0.86, which

indicates that the clustering results are in good agreement with human perception.

3.2 Rain-contaminated Region Segmentation Using SegNets

Although the unsupervised learning-based methods presented in [97, 98] is able to segment between rain-free and rain-contaminated regions by extracting multiple texture features from each pixel, they do not perform very well in the near range because the wave signature textures can be distorted under very strong backscatter intensity. In addition, the feature selection procedure is iterative, time-consuming, and resource-intensive. On the other hand, it has been observed in previous studies [36, 33] that under high wind speeds, rain may not blur the wave signatures in radar images because wind force dominates the generation of surface roughness. In consequence, rain only causes additional radar backscatter and does not affect the accuracy of wave and wind directions estimation significantly. In this work, images with this characteristic are named as wind-dominated rain cases. Thus, it is worthwhile to develop a more accurate and automatic method for classifying between rain-free, rain-contaminated, and wind-dominated rain pixels in X-band radar images.

As the state-of-the-art models for pixel-wise semantic segmentation, fully convolutional networks (FCNs) have been widely applied in research concerning remote sensing, such as sea-land segmentation [99], cloud recognition [100], and the semantic segmentation of high-resolution remote sensing images of land [101, 102]. Compared to typical CNNs, a stack of deconvolution layers are added after the convolution layers in FCNs, which upsamples the coarse outputs to pixel-wise outputs. Also, the fully connected layers appearing in CNNs are removed. This prevents the SegNet from losing spatial information. As a result, FCNs are able to take input of arbitrary size and produce correspondingly-sized output with efficient inference and learning [103].

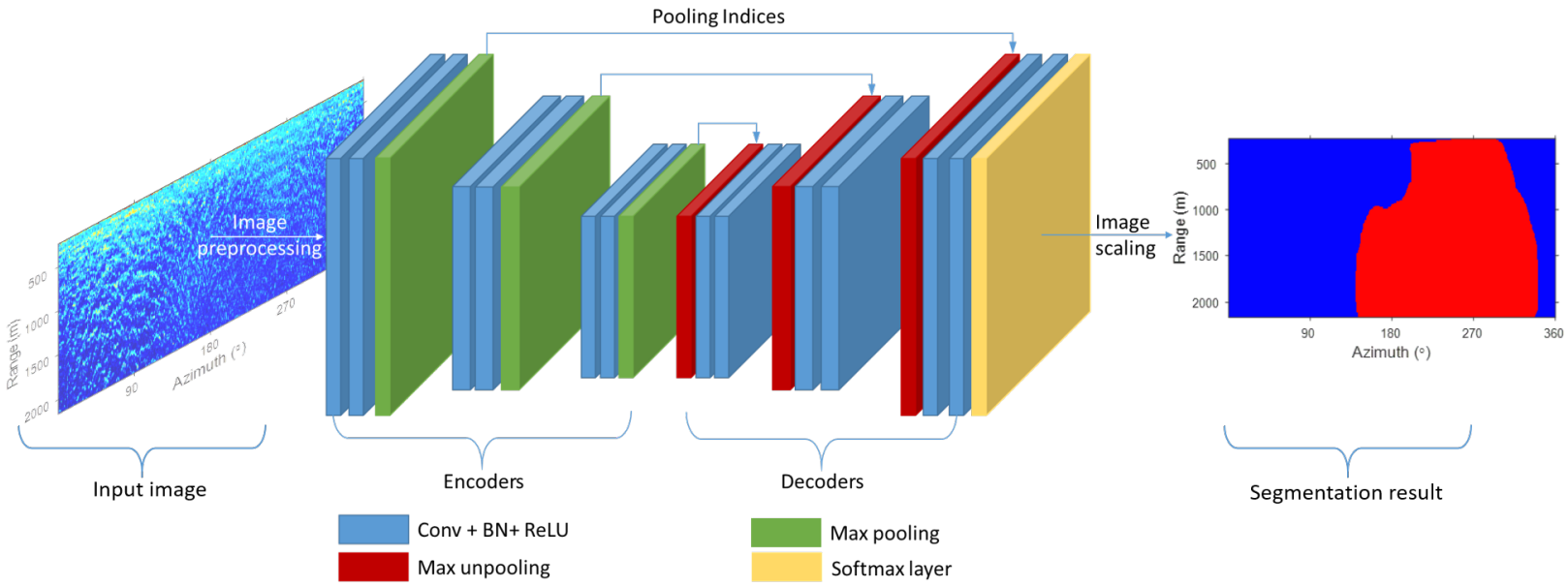

Figure 3.15: An illustration of the end-to-end SegNet-based segmentation model. The blue and red region in the segmentation result correspond to the rain-free and rain-contaminated pixels in the input image, respectively.

Since the whole network is trained end-to-end and pixels-to-pixel, those rigidly designed artificial methods for feature extraction are no longer required. Among various types of FCNs with different architectures, the SegNet proposed in [104] provides good performance with competitive inference time and most efficient inference memory-wise as compared to other FCNs.

Fig. 3.15 shows the whole structure of the proposed SegNet-based segmentation model. In order to generate the segmentation result, a radar image needs to undergo the following procedures: image preprocessing, encoding, decoding, pixel-based classification (softmaxing) and image scaling. Each procedure is illustrated below.

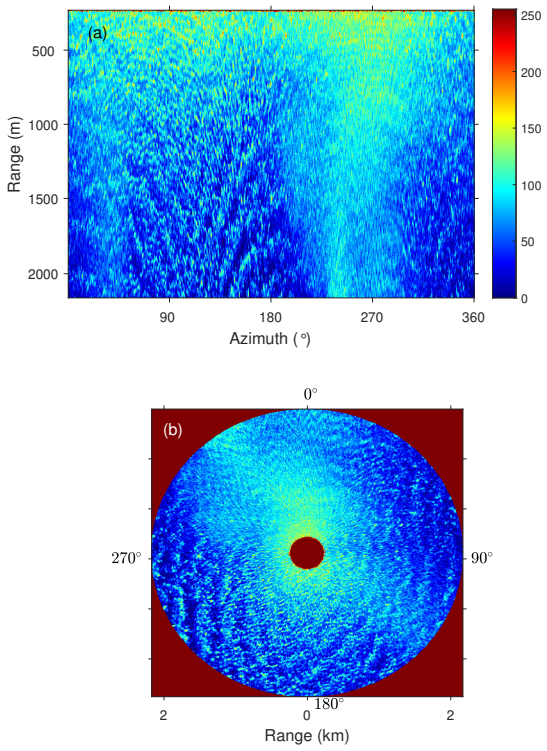

Figure 3.16: (a) A rain-contaminated X-band marine radar image in polar coordinates. (b) The X-band marine radar image in (a) converted into Cartesian coordinate.

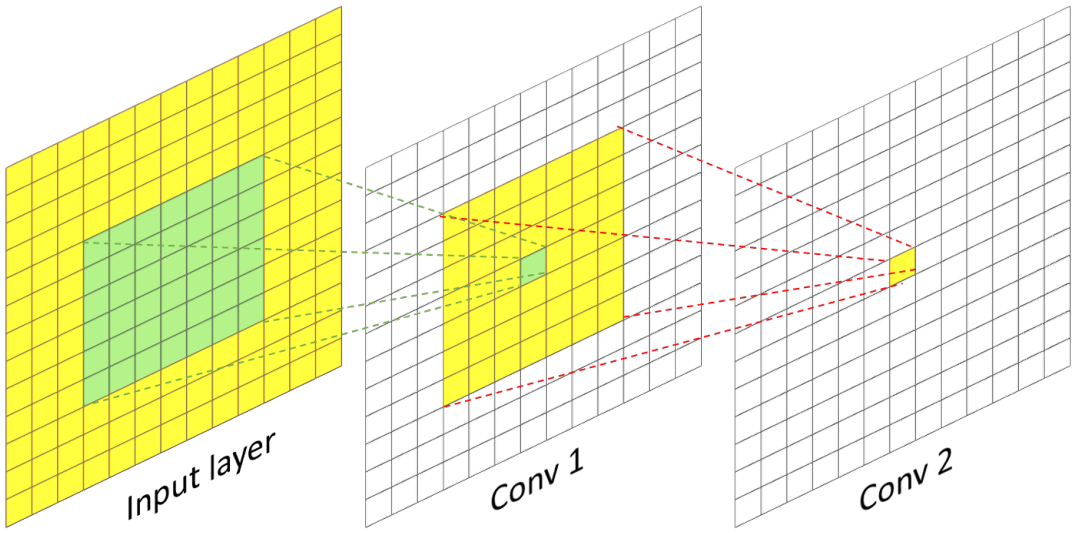

Figure 3.17: The receptive field of each layer with a 7×7 kernel used in the proposed network. The green area in the input layer corresponds to the receptive field of the pixel outlined in green in Conv 1, while the yellow area in the input layer indicates the receptive field of the yellow pixel in Conv 2.

3.2.1 Preprocessing of Radar Images

Two types of coordinate systems can be used to generate marine radar images, i.e., polar and Cartesian. In Fig. 3.16, an X-band marine radar image is depicted in both polar and Cartesian coordinates. In this study, the polar coordinate is adopted due to the following reasons: 1) the whole image only consists of radar backscatter, which makes it more convenient to process using image processing techniques. In contrast, the corners and centers of Cartesian images are blank backgrounds which should be excluded from result analysis; 2) scan conversion from polar to Cartesian coordinates may bring extra artificial distortion [105].

After generating polar images from radar data files, the noise line removal operation introduced in Subsection 2.2.1 is then implemented. Since the images fed into the encoders of the network are supposed to be fixed in size, all the radar images are first resized into a uniform size of 256 (rows) × 1024 (columns) using bilinear interpolation. Then, they are all downsampled by 2, indicating that the input data size of

the designed network is 128 (rows) × 512 (columns). The purpose of downsampling is to facilitate the designed encoders in obtaining a large sized receptive field. Although downsampling may introduce aliasing of shorter length scales. However, since this procedure is only used in the rain-contaminated region segmentation instead of wave parameter estimation, it does not impact the retrieval of wave statistics. As shown in Fig. 3.17, the receptive field refers to the part of the image that is visible to one filter at a time. A large sized receptive field is suitable for large area segmentation with obscure boundaries, such as the segmentation between rain-free and rain-contaminated regions in this context. In contrast, a small sized receptive field is favorable for small object detection with clear boundaries.

3.2.2 SegNet-based Network Architecture

Like other types of FCNs, a SegNet consists of the following parts: an input layer, encoding sections (encoders), decoding sections (decoders), softmax layers, and an output layer in the end. While texture features used in [57, 97] can be analyzed and extracted from the radar image for segmentation purpose, deep neural networks are able to exploit feature representations learned exclusively from data instead of handcrafting features that are designed based mainly on domain-specific knowledge [106]. In this study, the radar image is input into the SegNet-based end-to-end neural networks, which can generate various feature maps automatically through training and do not require feature engineering by analyzing the functions of different components or layers. Specifically, the encoder layers are used to extract high-level abstract feature information, while the decoder layers map the low resolution encoder features at the original resolution. The designed SegNet in this work incorporates three encoding and decoding sections. Each encoder/decoder section consists of two convolutional layers (represented by two consecutive blue blocks in Fig. 3.15) and one max pooling/unpooling layer (represented by one green/red block in Fig. 3.15). The

configuration of each layer in encoder/decoder sections is specified in Table 3.2. For one convolutional layer, it first performs convolution with a filter bank composed of various filters, which produce a set of feature maps.

Similar to other studies concerning the application of deep neural network to remote sensing, the setting and hyperparameters (e.g., number of layers, filter size, fiter number) of the network are determined through fine-tuning from the original SegNet proposed in [104]. Specifically, various combinations of hyperparameters values selected randomly (but also based on empirical experience) have been used to train the network here. The combination of hyperparameters with the highest training accuracy and relatively short training time is selected to construct the proposed networks. In a FCN, the filter size determines the area of the input data required to calculate one single element in the output. As presented in Fig. 3.17, a relatively large-sized filter with a 7×7 window is selected, which is helpful for incorporating information with large receptive fields. It can also be perceived that increasing the number of convolution layers (abbreviated as Conv in Fig. 3.15) will increase the receptive field of the output layer as well. The feature maps are then input into a batch normalization (BN) layer followed by a rectified linear unit (ReLU). The goal of batch normalization is to improve the speed, performance, and stability of the network by reducing internal covariate shift [107]. The ReLU contains a piece-wise linear activation function that will output the input directly if it is positive, otherwise, it will output zero. Therefore, given a feature map F_i, the output of a batch normalization with ReLU activation, i.e., $\mathrm{BN_R}(F_i)$, can be expressed as

$$\mathrm{BN_R}(F_i) = \max\left(0, \frac{F_i - \mathrm{E}(F_i)}{\sqrt{\mathrm{Var}(F_i) + \epsilon_R}} \cdot \gamma_R + \beta_R\right), \tag{3.18}$$

where $\mathrm{E}(F_i)$ and $\mathrm{Var}(F_i)$ are the mean and variance of F_i, respectively. ϵ_R is added in the denominator for numerical stability and is an arbitrarily small positive constant.

Table 3.2: Setup of the encoders and decoders

Layer		Filter size	Filter number	Stride
Encoders	Conv 1/2	7×7	64	1
	Max pooling 1	2×2	-	2
	Conv 3/4	7×7	64	1
	Max pooling 2	2×2	-	2
	Conv 5/6	7×7	64	1
	Max pooling 3	2×2	-	2
Decoders	Max unpooling 1	2×2	-	2
	Deconv 7/8	7×7	64	1
	Max unpooling 2	2×2	-	2
	Deconv 9/10	7×7	64	1
	Max unpooling 2	2×2	-	2
	Deconv 11/12	7×7	64	1

γ_R and β_R are the scale and shift parameters subsequently learnt in the optimization process. For each two convolutional layers, a max pooling layer with a 2×2 window and stride 2 (non-overlaping window) is applied and the maximum value within that window is the output. This is done to: 1) reduce the number of parameters and computational cost of the network; 2) generalize the results from a convolutional filter by making the detection of features invariant to scale or orientation changes. Meanwhile, the location information of the maximum value in each window, i.e., max pooling indices, is saved by the network for decoding purposes.

The main difference between SegNet and other types of FCNs lies in the structure of decoders. As Fig. 3.15 shows, each layer in the decoder is symmetrically same to the corresponding layer in encoders, except that the max unpooling layers replace max pooling layers. For each max unpooling layer, it produces sparse feature maps by utilizing the max pooling indices obtained from its corresponding max pooling layer in symmetrical position, which is indicated by the arrow lines in Fig. 3.15. The upsampled feature maps are then convolved with trainable multi-channel decoder filters to densify its sparse inputs [104]. The full-sized feature maps at the output of the final decoder layer are fed into a softmax layer, which generates the probability of each

pixel belonging to a certain class. The output layer produces pixel-wise classification by selecting the class with maximum probability at each pixel. Nevertheless, since the size of the output layer is 128 (rows) $\times$ 512 (columns), the output image is upsampled by 2 using nearest-neighbor interpolation to generate the full-size segmentation result, as indicated by the "Image scaling" step in Fig. 3.15.

3.2.3 Model Training Using an Ensemble of SegNets

In order to provide ground truth for training, the radar images used for training are labeled manually. Specifically, every pixel of the image is labeled into three classes, i.e., rain-free, rain-contaminated (rain blurs wave signatures), and wind-dominated rain cases (rain does not blur wave signatures due to a high wind speed). The reason why the learning set is labeled based on a human expert is that it is impossible to obtain the rain information at each location within the coverage area of an X-band radar. Although simultaneous rain rate data provided by the rain gauge installed on the ship and within the radar coverage facilitates us to select images that might contain rain, it does not provide the spatial distribution information about the rain. Therefore, labeling the learning set based on a human expert is the only practical choice for developing the proposed method. In fact, in many studies of ocean remote sensing, it is common practice to label images manually based on expert knowledge due to a lack of ground truth reference [108]. Although a human expert may not segment the boundary between rain-free and rain-contaminated regions with 100% accuracy, the input is reliable because the images in the training set are selected carefully. Here, only the images with distinctive difference between clear and blurry (i.e., rain-contaminated) wave signatures are selected for learning set labeling. This distinct difference can be better manifested by observing some of the texture feature maps generated from the rain-contaminated radar image. For example, Fig. 3.5(e) is one of the DWT texture feature maps calculated from the rain-contaminated radar image

presented in Fig. 3.1(b). A distinct difference in the pixel intensities between rain-free and rain-contaminated region can be observed, which facilitates the labeling process. Since manual labeling inevitably introduces human-induced errors, selecting training samples with clear differences between clear and blurry wave signatures can minimize those errors and further improve the accuracy and robustness of the model. Although the proposed algorithm is trained only using samples with distinct boundary, it is able to handle images without such clear boundary since most of those images are either fully rain-free or rain-contaminated. In addition, the learning set consists of radar images collected under various wind and wave conditions, which further improves the robustness of the proposed model.

The training of the proposed network proceeds as follows. First, all learnable parameters (i.e., filters and their bias term) in the network are initialized with the Glorot initializer [109]. Then, mini-batched stochastic gradient descent with momentum (SGDM) optimizer is introduced as the solver to update those parameters in each iteration. Details about the SGDM algorithm can be found in [110]. A mini-batch is defined as a subset of the training set. In this study, the mini-batch size is set to 16, indicating that 16 images are used to evaluate the gradient of the loss function and update the learnable parameters during each training iteration. The base learning rate of the proposed network is 0.01. When the entire training set has been fed into the network, one epoch is finished. Previous studies indicate the full dataset needs to be passed into the network multiple times in order to avoid underfitting. Thus, the maximum number of epochs for training is set to 100. In addition, in order to prevent the network from overfitting, a validation set is used to test the accuracy and loss of the network every 50 iterations. It should be noted that the network training will stop if the number of times when the loss on the validation set is larger than or equal to the previously smallest loss (i.e., defined as validation patience) reaches 8. To sum up, the training options are listed in Table 3.3.

Table 3.3: Training options of the designed SegNet-based network

Solver for network training	SGDM
Maximum number of epochs	100
Size of mini-batch	16
Validation frequency	50 iterations
Patience of validation stopping	8
Initial learning rate	0.01

Although a single network is able to generate segmentation results after training, it has been found in previous studies that averaging multiple deep networks instances further boosts performance [111]. That is because if the parameters of a network are initialized with different values, its loss function will probably converge to different local minima even using the same training data. In consequence, the classification results may vary slightly between different networks, especially for pixels located near the boundaries between different regions. Therefore, in order to increase the robustness of the proposed method, an ensemble of SegNets consisting of 10 individual networks are trained using the same structure and training options mentioned above. Instead of selecting one particular network, the classification results are produced from all of the 10 networks based on a majority voting scheme. In other words, for each pixel, its classification result would be the class chosen by most individual networks.

3.2.4 Data Overview

Image data collected by two different X-band marine radar systems (i.e., Decca and Koden) at two different locations are used to train the SegNet-based segmentation model and evaluate its performance. The basic information of those two radars is given in Table 3.4. The parameters in Table 3.4 determine the characteristics of the radar image. For example, the range resolution and azimuthal resolution of the radar image are related to the pulse width and beam width, respectively. As for the antenna height, according to the operating manual of the WaMoS II [75], in order to set up

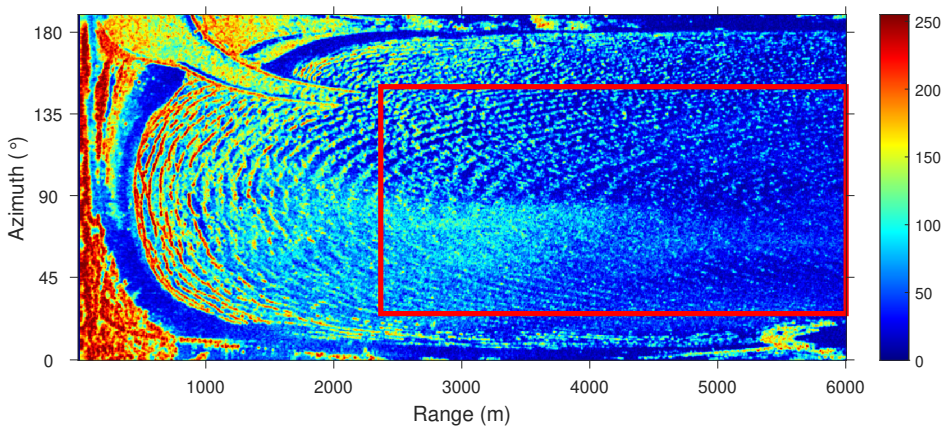

Figure 3.18: An example of a rain-contaminated Koden radar image. The red bin is the study region.

the system correctly, the minimum antenna height is 15 m. Both radars satisfy this requirement. However, the effect of antenna height on the results is beyond the scope of this thesis and should be addressed in future extended work. The Decca radar images as well as simultaneous environmental data have already been described in Subsection 2.1.2. After image resizing and downsampling, the physical resolution of the image input into the network is around 15 m and 1.4° in the range and azimuth directions, respectively. As for the Koden radar owned by the Nearshore Remote Sensing Group of Oregon State University, it was installed on a United States Coast Guard watchtower near the North Pacific Ocean. The tower was located in the north of the Yaquina Bay Inlet [44.624125, -124.062888]. A detailed description of the location information, radar and tower equipment can be found in [112]. The Koden image data used in this study were collected from 00:00, December 18 to 23:00, December 23 of 2019, at a rate of 45 images per minute. However, as only one image is selected for analysis per hour, a total of 144 images are available. That is because the owner of the dataset only provided one image per hour and other images were not saved. As depicted in Fig. 3.18, in order to avoid the influence of the shore on sea clutter, only the region outlined in red is selected and input into the proposed model for analysis. Since the Koden images sampled a much greater number of range pixels (2029) than

Table 3.4: Radar information

	Decca	Koden
Installation site	Ship	Shore-based Tower
Polarization	Horizontal	Horizontal
Operating frequency	9.41 GHz	9.45 GHz
Pulse width	50 ns	80 ns
Range resolution	7.5 m	12 m
Range coverage	240 m - 2160 m	3 m - 6087 m
Beam width	2°	0.8°
Azimuth coverage	360°	190°
Antenna rotation speed	28 rpm	44 rpm
Antenna height	21.9 m	63 m
Backscatter intensity level	0 - 255	0 - 255

azimuth (381), the selected region in Fig. 3.18 is resized and downsampled into 128 (azimuth) $\times$ 512 (range) instead of 128 (range) $\times$ 512 (azimuth) in order to alleviate the distortion of wave signatures caused by resizing operation. Thus, the physical resolution of the image input into the network is around 7.2 m and 0.97° in the range and azimuth directions, respectively. Meanwhile, simultaneous wind speed and rain rate data collected from two APRSWXNET stations in the Newport residential area North of Yaquina Bay were averaged as reference and presented in Fig. 3.19(a) and (b), respectively. It should be noted that some parts of the wind speed data are missing and shown as 0 in the figure.

3.2.5 Segmentation Results and Analysis

The rain-free, rain-contaminated, and wind-dominated rain classes in radar datasets are defined and labeled using a two-step procedure. First, the high clutter direction (HCD) percentage proposed in [36] is extracted from each radar image and used as a threshold to determine the presence of rain. Images with HCD percentage higher than 5% are assumed to be collected under rain conditions. The high pixel percentage (HPP) proposed in [33] is then introduced to classify between rain-contaminated and wind-dominated rain images. If the HPP of a rain-contaminated image is lower than

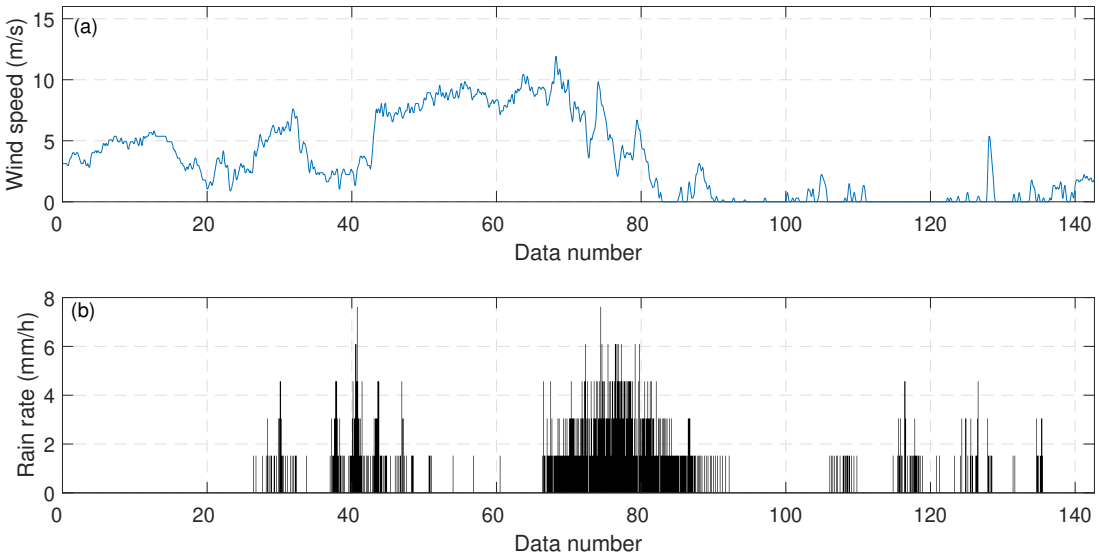

Figure 3.19: Simultaneous wind speed & rain rate collected from APRSWXNET stations in the Newport residential area North of Yaquina Bay during Koden radar data collection periods.

15%, the image will be identified as rain-contaminated; otherwise, it will be classified as wind-dominated rain cases. Since the identified rain-contaminated images may still contain regions that are not significantly affected by rain, the next step is to label the rain-contaminated regions with blurry wave signatures manually in each rain-contaminated image. Since the volume of rain-contaminated Decca data (around 600 images in total) is much more than rain-contaminated Koden data (around 40 images in total), a total of 26 Decca images are first labeled manually using the Image Labeler toolbox in MATLAB, and then augmented to 208 samples (images) using circular array rotation. Specifically, for each original radar image consisting of 1024 azimuthal directions, a total of 8 training samples (images) can be generated by shifting 128 directions circularly for 8 times. In this way, a sufficient number of samples can be obtained for network training and validation. Also, compared to other operations such as horizontal flipping, circular array rotation ensures that the training samples are obtained under a wide range of upwind directions. Consequently, the robustness of the network can be further improved. Those samples are utilized as

training (144 samples) and validation (64 samples) datasets to train the SegNet-based networks. The Koden data, as well as the Decca data that are not included for training and validation, can all be utilized to test the performance of the proposed model. The SegNet-based segmentation model is built and trained with MATLAB R2019a programming on a high-performance computing (HPC) cluster named Cedar provided by Compute Canada, an organization responsible for digital research infrastructure in Canada. Located at Simon Fraser University, Cedar has a total of 58416 CPU cores and 584 GPU devices for computation. In this study, four GPUs (NVIDIA P100-PCIE-12GB) were assigned to train the model using parallel computing techniques. Among the 10 individual networks, it took an average of less than 4 minutes to train each of them. The average validation accuracy of each individual network reached 95.82%.

The testing of the model was conducted in MATLAB R2019b installed on a local PC running Windows 10 with a 3.0 GHz Intel Core i5-2320 CPU, 8 GB memory, and a 64-bit operating system. Since for each radar image it only took an average of around 1.5 s to generate the segmentation result, the proposed model can be completely implemented on a normal PC in real-time. Figs. 3.20 and 3.21 show some examples of the segmentation results using Decca and Koden testing data, respectively. It can be observed that the segmentation results (i.e., images in the third column) obtained from the proposed model are in good agreement with the manual labeled images (i.e., images in the second column) in most of the regions under different wind speeds and rain-contaminated coverages for both datasets. In addition, the proposed model is able to well separate between rain-contaminated (i.e., red regions) and wind-dominated rain cases (i.e., yellow regions), by determining whether wave signatures are blurred by rain. It should be noted that for the Koden images, as the simultaneous wind

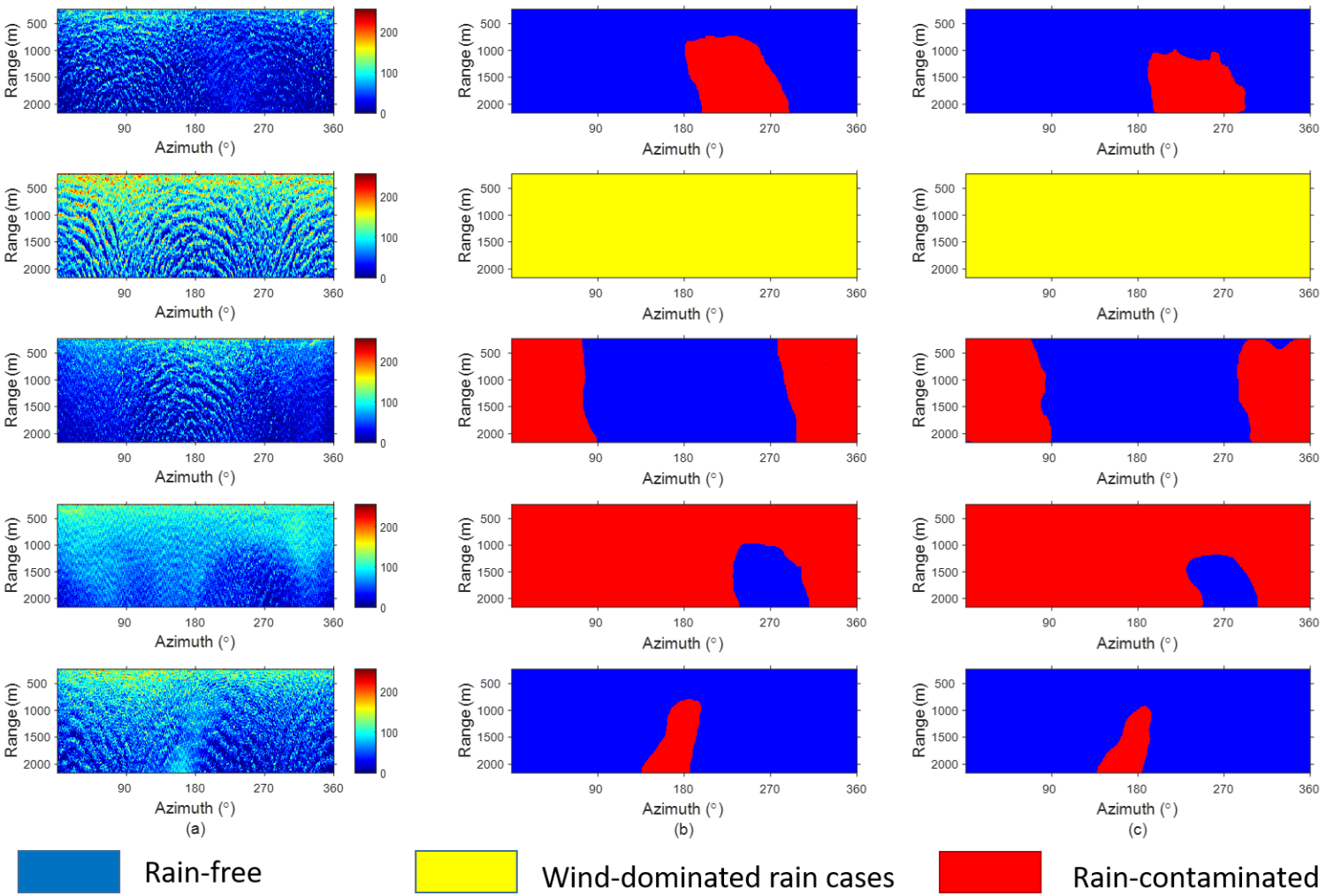

Figure 3.20: (a) Examples of Decca images for testing. (b) Manual labels of images in (a). Blue, red and yellow color indicate rain-free, rain-contaminated, and wind-dominated rain cases, respectively. (c) Segmentation results of images in (a) obtained from the proposed model.

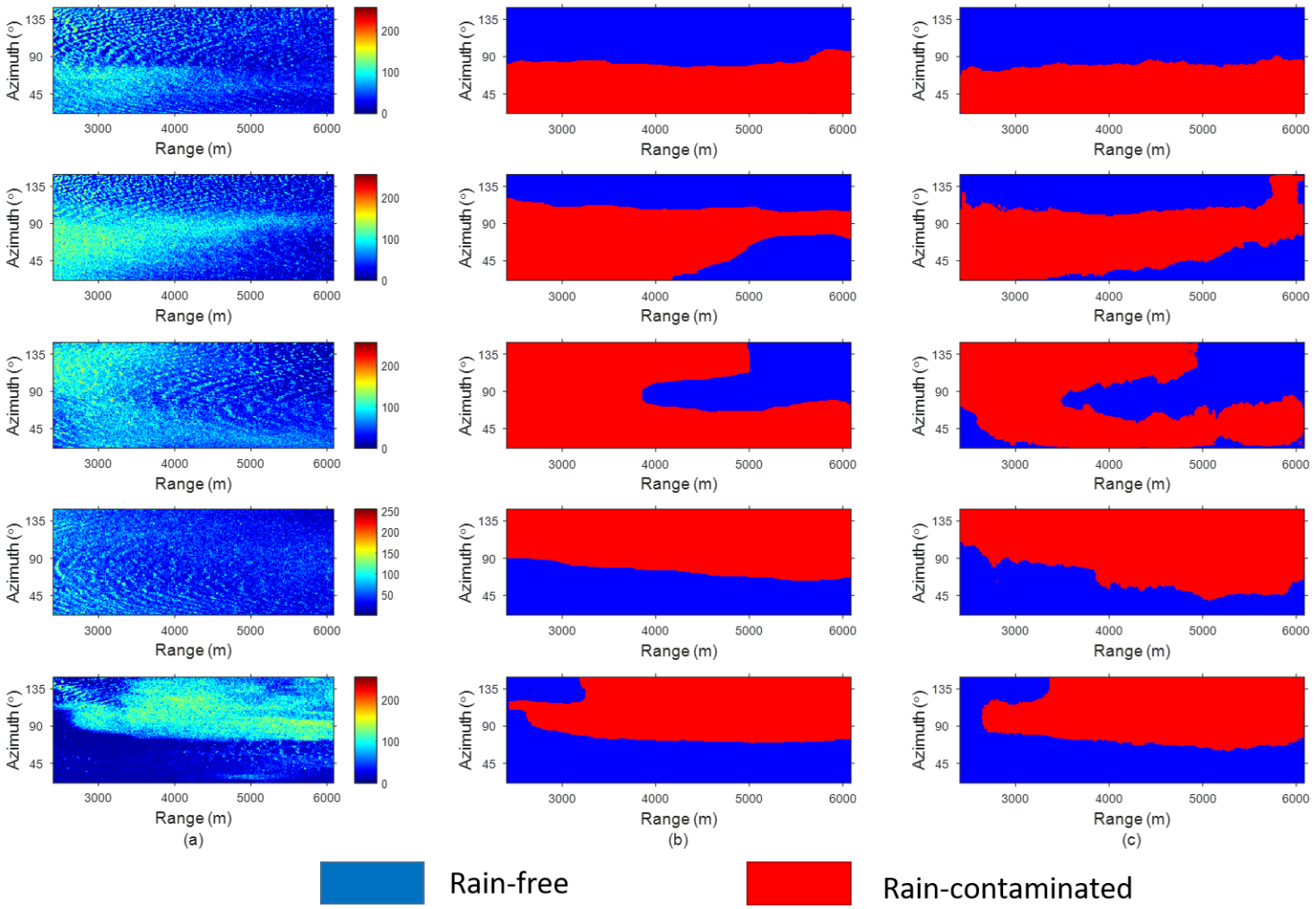

Figure 3.21: (a) Examples of Koden images for testing. (b) Manual labels of images in (a). Blue and red color indicate rain-free and rain-contaminated pixels, respectively. (c) Segmentation results of images in (a) obtained from the proposed model.

speeds during data collection periods were mostly below 10 m/s, no wind-dominated rain images were detected.

Despite the satisfactory results observed from testing samples, quantitative analysis is required to further evaluate the effectiveness of the proposed model. Therefore, a part of rain-contaminated testing data (i.e., 40 Decca and 30 Koden images) are labeled manually and used as ground truth to compare with their corresponding segmentation results. Confusion matrices generated from Decca and Koden images are presented in Fig. 3.22 and Fig. 3.23, respectively. Each column of the matrix represents the number and percentage of pixels in an actual class (manually-labeled results). Each row represents the number and percentage of pixels in a predicted class (segmentation results using the proposed model). Several metrics can be acquired from the confusion matrix to assess the segmentation accuracy of the proposed model, which are calculated as follows:

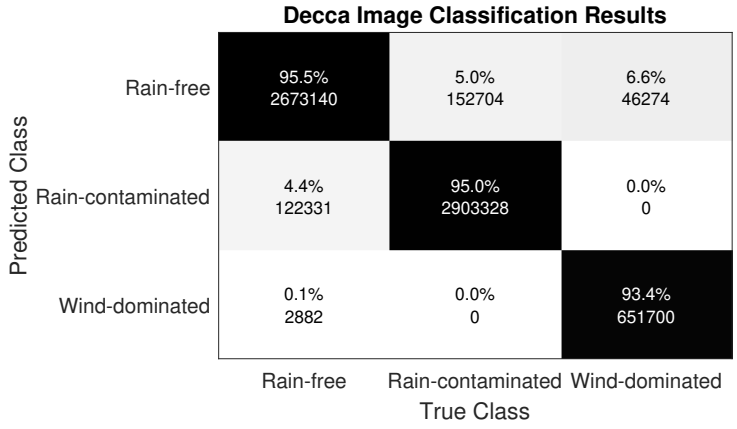

Figure 3.22: Confusion matrix of the pixel classification results using Decca images. The true class and predicted class correspond to manually-labeled results and segmentation results using the proposed model, respectively. The class "wind-dominated" stands for wind-dominated rain cases.

Mean accuracy (MA): For each class, its mean accuracy is the ratio of correctly classified pixels to the total number of pixels in that class in all testing images,

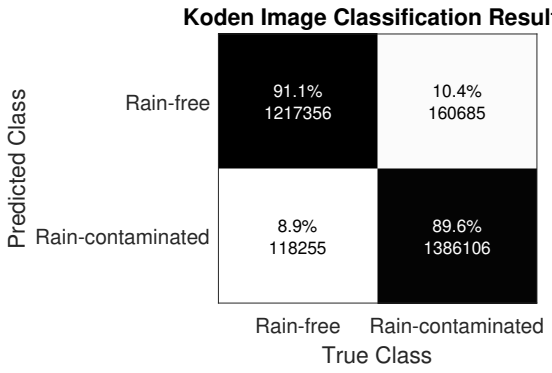

Figure 3.23: Confusion matrix of the pixel classification results using Koden images. The true class and predicted class correspond to manually-labeled results and segmentation results using the proposed model, respectively.

according to the ground truth. In other words,

$$\mathrm{MA}_i = \frac{\mathrm{TP}_i}{\mathrm{TP}_i + \mathrm{FN}_i}, \tag{3.19}$$

where MA_i is the mean accuracy of the i^{th} class. For the whole dataset, its MA can be expressed as

$$\mathrm{MA} = \frac{\sum_{i=1}^{n} \mathrm{TP}_i}{\sum_{i=1}^{n} (\mathrm{TP}_i + \mathrm{FN}_i)}, \tag{3.20}$$

where n is the total number of classes.

Intersection over union (IoU): Intersection over union (IoU), also known as the Jaccard similarity coefficient, is the most commonly used metric for semantic segmentation. For the i^{th} class, its IoU equals the ratio of correctly classified pixels to the total number of ground truth and predicted pixels in that class in all testing images, i.e.,

$$\mathrm{IoU}_i = \frac{\mathrm{TP}_i}{\mathrm{TP}_i + \mathrm{FP}_i + \mathrm{FN}_i}. \tag{3.21}$$

For the whole testing dataset, its IoU is the average IoU score of all classes in all

images, which can be expressed as

$$\text{IoU} = \frac{\sum_{i=1}^{n} \text{TP}_i}{\sum_{i=1}^{n} (\text{TP}_i + \text{FP}_i + \text{FN}_i)}. \tag{3.22}$$

Generally, an IoU greater than 0.5 is considered as a good prediction.

Boundary F1 contour matching score (F_1 score): The F_1 score indicates how well the predicted boundary of each class aligns with the ground truth boundary. For each class, its F_1 score can be calculated as

$$\text{F}_{1i} = 2 \cdot \frac{\text{PR}_i \cdot \text{MA}_i}{\text{PR}_i + \text{MA}_i}, \tag{3.23}$$

where $\text{PR}_i = \text{TP}_i/(\text{TP}_i + \text{FP}_i)$. Similarly, for the whole dataset, its F_1 score is the average F_1 score of all classes in all testing images, which can be expressed as

$$\text{F}_1 = 2 \cdot \frac{\text{PR} \cdot \text{MA}}{\text{PR} + \text{MA}}, \tag{3.24}$$

where $\text{PR} = \sum_{i=1}^{n} \text{PR}_i$. Similar to IoU, F_1 score reaches its best value at 1 (perfect segmentation) and worst at 0.

Kappa coefficient: The Kappa coefficient is a statistic that is used to measure the agreement between classification and truth values. In broad terms, a Kappa value above 0.8 indicates very good agreement beyond chance. Its expression is shown in Eq. (3.17)

The accuracy assessment of the Decca and Koden datasets using the metrics introduced above is presented in Table 3.5. It can be observed that both datasets obtain a mean accuracy of above 90%, indicating that more than 90% of the pixels are classified correctly. The classification accuracy of each class is also relatively high and very close to each other as well. The IoUs as well as Kappa values for both datasets are

Table 3.5: Accuracy assessment using multiple metrics

Dataset		MA	IoU	F_1 score	Kappa
Decca	All	94.6%	0.912	0.718	0.916
	Rain-free	95.5%	0.892	0.664	-
	Rain-contaminated	95.0%	0.914	0.705	-
	Wind-dominated rain cases	93.4%	0.930	0.730	-
Koden	All	90.4%	0.824	0.673	0.806
	Rain-free	91.2%	0.814	0.667	-
	Rain-contaminated	89.6%	0.833	0.607	-

Table 3.6: The accuracy assessment of SVM-based method as a comparison

Dataset		MA	IoU	F_1 score	Kappa
Decca	All	89.5%	0.818	0.635	0.810
	Rain-free	90.2%	0.826	0.623	-
	Rain-contaminated	89.9%	0.807	0.642	-
	Wind-dominated rain cases	88.0%	0.815	0.625	-
Koden	All	82.6%	0.754	0.581	0.723
	Rain-free	84.2%	0.739	0.574	-
	Rain-contaminated	80.7%	0.756	0.543	-

both above 0.8, which indicates that good segmentation and classification results have been obtained from both datasets using the proposed model. Nevertheless, it is found that the segmentation results of Decca data have better agreement with labels than Koden data with higher values in all metrics (e.g., 4.2% higher in MA, 0.088 higher in IoU, 0.045 higher in F_1 score and 0.11 higher in Kappa value). By having a closer look at the testing examples presented in Fig. 3.20 and Fig. 3.21, it is found that the proposed model segment the boundaries between rain-free and rain-contaminated regions more accurately using Decca images than Koden images. That might be due to the fact that the networks were trained only using Decca images and the differences in many aspects (including the overall sampling regime of the data acquisition system, the physical resolution of the image after downsampling, the antenna elevations, and the nearshore vs offshore environment) between two radar datasets may negatively affect the performance of the proposed model. Thus, in order to further

Table 3.7: Training and testing time consumed for both methods

	SegNet	SVM
Model training time	∼4 min (per network)	∼5 min
Testing time for each image	∼1.5 s	∼45 s

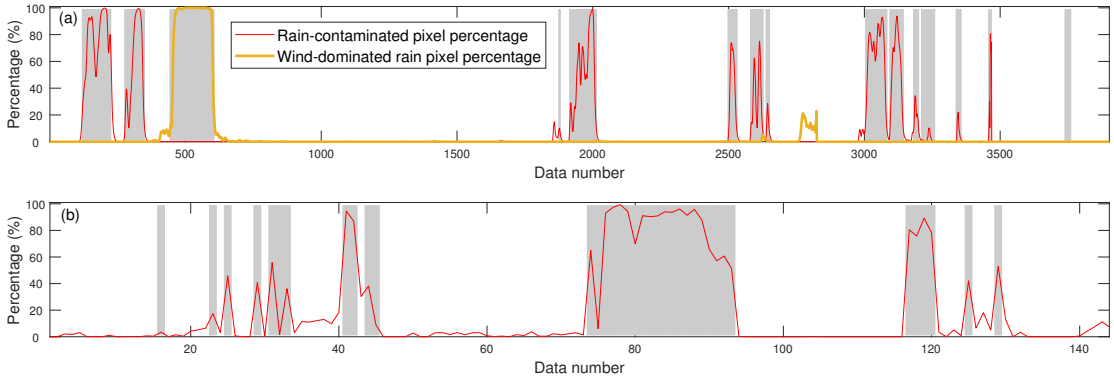

Figure 3.24: (a) Red line: The percentage of rain-contaminated pixels in each Decca image. Yellow line: The percentage of wind-dominated rain pixels in each Decca image. Images with HCD percentage higher than 5% are indicated by the grey shadow. (b) Red line: The percentage of rain-contaminated pixels in each Koden image. Images with HCD percentage higher than 5% are indicated by the grey shadow.

validate the robustness of the proposed method, the same network is also trained by using the same number of Koden images. The Decca data, as well as the Koden data that are not included for training and validation are utilized to test the performance of the proposed model. It is found that similar classification accuracies can be obtained (93.3% and 92.1% for Decca and Koden data, respectively). Also, in order to demonstrate the necessity of applying SegNet for rain-contaminated region segmentation, results generated using a simpler AI method (here SVM) are also analyzed for comparison. Specifically, four types of texture features illustrated in [97] are first extracted from each pixel. Similar to [78] (i.e., Section 2.1), the SVM-based model is implemented within the SVM solver function (fitcsvm) provided by the Statistics and Machine Learning Toolbox of MATLAB. The RBF kernel is selected for the solver function because it generates the best training accuracy compared to other kernels. The RBF kernel is selected for the SVM-based model because it generates the best

training accuracy compared to other kernels. Besides, the 10-fold cross-validation is introduced to avoid model overfitting. In addition, the SMO algorithm is adopted to optimize the hyperparameters (e.g., box constraint and kernel scale) during the training process. The accuracy assessment of the SVM-based segmentation model using both Decca and Koden testing samples with different types of metrics (i.e., MA, IoU, F1 score and Kappa coefficient) is presented in Table 3.6. It can be observed that the performance of the SVM-based model using Decca testing data is also good, with a mean pixel classification accuracy (denoted as MA in the table above) of 89.5%. However, there is an obvious drop in all the metrics above when using the Koden radar data. For example, the mean pixel classification accuracy is only 82.6%. By comparing Table 3.5 and Table 3.6, it can be clearly seen that the segmentation accuracies of the SegNet-based method are higher in all the metrics for both datasets (e.g., 5.1% and 7.8% higher in MA for the Decca and Koden testing data, respectively). In the end, the training and testing time consumed for both methods are listed in Table 3.7. It can be observed that, although the training time of the SegNet-based and the SVM-based model are similar, the time consumed for the SegNet-based model to generate segmentation results from one radar image is much shorter than that of the SVM-based model. That is because multiple types of texture features need to be computed and extracted when using the SVM-based model, which is time-consuming. In contrast, the convolution/deconvolution operation in the SegNet-based model can work much more efficiently.

In order to further evaluate the effectiveness and accuracy of the proposed method for rain detection, simultaneous rain rate data covering the radar site provided by the European Centre for Medium-Range Weather Forecasts (ECMWF) is used. The data are derived using the advanced precipitation hindcasting techniques, and each hourly recorded point measurement value in the dataset refers to the measured rainfall in the previous hour (i.e., average rainfall per hour, mm/h) covering an area of 0.25° (around

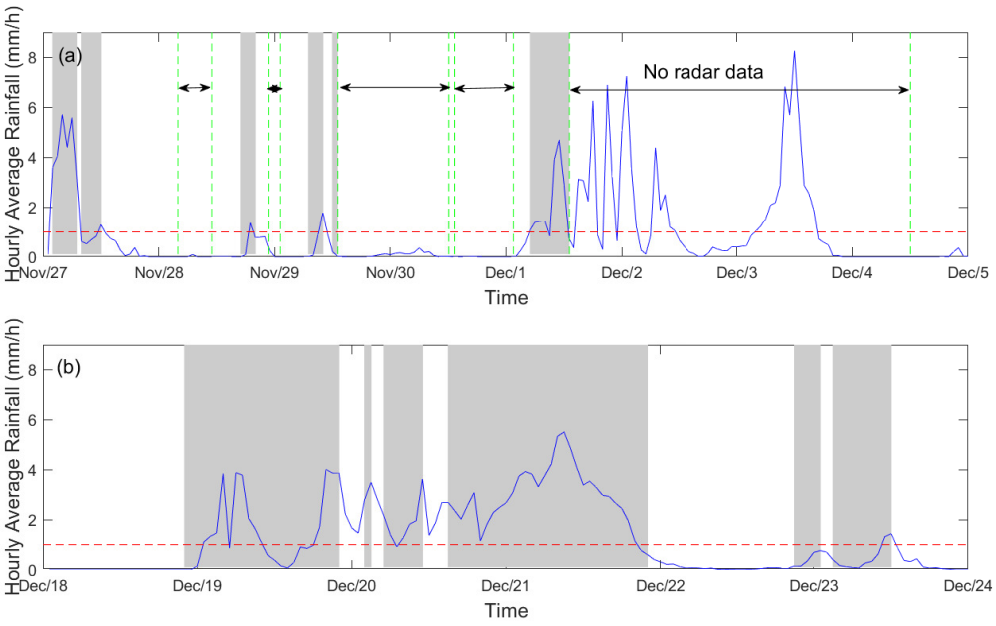

Figure 3.25: Hourly average rainfall (dots connected by blue lines) values provided by ECMWF during Decca (a) and Koden radar data (b) collection periods. Simultaneous radar images with rain-affected regions are indicated by gray shadows.

27 km) in both latitude and longitude, which is much larger than the radar coverage. Therefore, it can only be used to validate the presence of rain instead of indicating the spatial distribution of rain in the radar image. Since the Decca radar data were collected on a moving vessel, the average rain rate of each hour during data collection period is provided by the ECMWF measurement that covered the ship location during that time. As for the images collected by the shore-based Koden radar, the ECMWF measurements that cover the location of the radar site (44.62°N, 124.06°W) are utilized for providing rain rate data. Figs. 3.25(a) and (b) are presented to show whether the proposed method can detect the presence of rain accurately using Decca and Koden radar data, respectively. Specifically, the hourly average rainfall values are depicted as the blue lines. The gray shadows refer to the presence of rain-affected (including both rain-contaminated and wind-dominated rain cases) regions identified using the proposed method for simultaneous radar images. If more than 20% of the pixels in the radar image are identified as rain-contaminated, the image would be considered

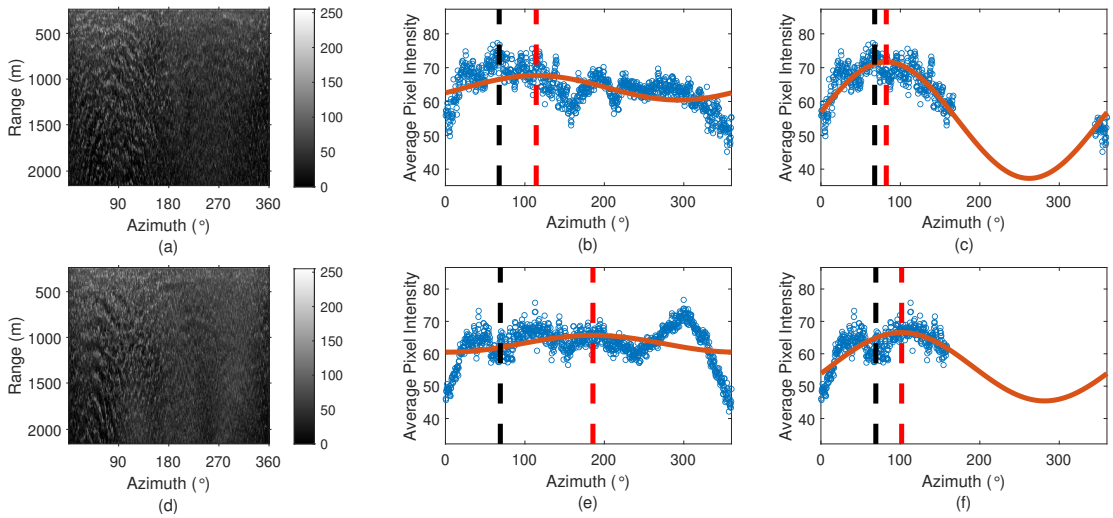

Figure 3.26: (a) An example of a rain-contaminated Decca radar image. (b) Wind direction estimation result (a) without excluding rain-contaminated regions. Red and black dash lines indicate radar-derived wind direction and anemometer-measured wind direction, respectively. (c) Wind direction estimation result (a) after excluding rain-contaminated regions. Red and black dash lines indicate radar-derived wind direction and anemometer-measured wind direction, respectively. (c) An example of a rain-contaminated Decca radar image. (d) Wind direction estimation result (c) without excluding rain-contaminated regions. Red and black dash lines indicate radar-derived wind direction and anemometer-measured wind direction, respectively. (e) Wind direction estimation result (c) after excluding rain-contaminated regions. Red and black dash lines indicate radar-derived wind direction and anemometer-measured wind direction, respectively.

as rainy. It should be noted that in Fig. 3.25(a), radar data were not collected during several time periods, which are indicated by the black arrow lines between the green dash lines. For the Decca radar data, it can be observed that when the hourly average rainfall is not lower than 1 mm/h, rain-affected regions can always be detected (i.e., a detection success rate of 100%) for the 60 radar images obtained in each hour using the proposed method. On the other hand, when the hourly average rainfall is nonzero but less than 1mm/h, rain-affected regions might not be identified, which is probably due to the sparse distribution of rainfall in time or space. As for the Koden radar data, rain-affected regions can only be detected in 87.5% of the images obtained under rain conditions with hourly average rainfall greater or equal to 1 mm/h. The main

reason is that compared to the Decca radar images, only one image was provided per hour in the Koden radar data. Due to the fact that then rainfall distribution can be uneven in both time and space, as well as the fact that the radar coverage is much smaller than the measured region, it is expected that the rain detection accuracy can be improved significantly if more image samples are available.

In order to provide an overview concerning the segmentation results of all images, the percentage of rain-contaminated and wind-dominated rain pixels for each image is presented as red and yellow lines, respectively in Fig. 3.24. As Fig. 3.24 shows, gray shadows indicate images that are detected with the presence of rain using HCD percentage. As the gray shadows and those high-value points (i.e., points greater than 20%) in red and yellow lines are in good agreement, the proposed model is able to detect the presence of rain accurately. It can also be observed that wind-dominated rain pixel percentage is almost 100% from Decca data number 450 to 600, during which simultaneous wind speeds were mostly above 13 m/s.

Finally, in order to demonstrate effectiveness of the proposed method in application, the network is applied for wind direction estimation from rain-contaminated Decca radar images. Specifically, regions that are identified as rain-contaminated using the proposed network are excluded from analysis. For each rain-contaminated image, the average pixel intensities of azimuths that are unaffected or less affected by rain are curve-fitted using Eq. (1.1). The estimated wind direction is the azimuth corresponding to the peak of the fitted function. Simultaneous wind directions measured by anemometers are used as ground truth to evaluate the estimation accuracy. Estimation results show that compared to curve fitting directly on original radar images without rain-contaminated region identification, the estimation accuracy is improved with a reduction of 13.2° in root-mean-square deviation (RMSD) which manifests that the proposed method has the potential to further improve the estimation accuracy of other ocean parameters (e.g., wind speed, wave height) under rain conditions. Two

examples of wind direction estimation from rain-contaminated images are presented in Fig. 3.26 to demonstrate the improvement of estimation accuracy after incorporating the proposed method.

3.3 Chapter summary

In this chapter, two methods are presented to identify pixels dominated by rain echoes in X-band marine radar images. In Section 3.1, unsupervised clustering techniques are employed to identify rain-contaminated and low-backscatter regions in X-band marine radar images. Based on the observation that rain blurs wave signatures and alters the texture of the radar image, four types of texture features (i.e., Gabor, DWT, DCT and histogram) are extracted from each pixel of the radar image and combined into a feature vector. The feature vectors are trained using a 10×10-neuron SOM, which maps pixels into 100 neurons. Those neurons are then further clustered into three classes (i.e., rain-free, rain-contaminated, and low-backscatter types) using hierarchical agglomerative clustering technique. The type of a pixel is therefore determined by the type of the neuron it is clustered into. Radar data collected during a sea trial off the east coast of Canada are used to validate the proposed method. It is observed that images with low or moderate coverage of rain-contaminated regions appear at the beginning or end of rain events. Images largely contaminated by rain mainly are present in the middle of rain events. Besides, it is found that rain-contaminated pixels that are clustered into two groups of neurons differ significantly in texture and pixel intensities, which may be caused by different rain rates in different regions. Also, for some rain-contaminated images obtained under high wind speed, the wave signatures are not blurred by rain since wind-induced effects dominate radar backscatter. On the other hand, images containing low-backscatter regions were mostly obtained under a wind speed of 6 m/s. Examples of radar images presented in the paper show

that rain-contaminated and low-backscatter regions are effectively identified and well segmented from rain-free regions. Comparison between manually labeled reference images and clustering results show that 92.8% of pixels are clustered correctly, which proves the good agreement between identification results and human observation.

On the other hand, a supervised learning-based model is developed in Section 3.2 to classify each pixel into three classes: rain-free, rain-contaminated (rain echoes blur wave signatures), and wind-dominated rain cases (rain echoes do not blur wave signatures). After preprocessing, polar radar images are input into 10 end-to-end SegNet-based semantic segmentation networks. The classification result of each pixel will be the class chosen by most individual networks. The full-size segmentation result can then be generated after image scaling. Data collected from two different marine radars in two different areas are used to train the designed SegNets and test the performance of the proposed model. By comparing with manual labeling results, it is found that 94.6% and 90.4% of pixels are correctly classified using Decca and Koden testing images, respectively. Several metrics are also computed to further evaluate the segmentation accuracy, which indicate that the proposed model is able to segment between rain-free and rain-contaminated regions accurately. Also, since for each radar image it only takes around 1.5 s to generate the segmentation result, the proposed model can be directly applied in real-time. Each method has its own pros and cons. For example, for the SOM-based method, it does not require manual labeling of the training samples, which saves much time and effort in model training. Although the SegNet-based method requires manual labeling, it performs better in the near range with higher computation efficiency. It can also be concluded that the use of a wide range of systems, wind speeds, and wave conditions (wind only and mixed seas) would improve the accuracy of models.

Chapter 4

Wind Parameter Estimation with Rain Mitigation and Support Vector Regression (SVR)

In recent years, a series of works [10, 33, 34, 57, 36, 91] managed to recycle the rain-contaminated radar images and applied novel or modified algorithms for wind retrieval, which improved the estimation accuracy under rainy conditions. Nevertheless, those methods still have certain limitations. For example, no strategy has been undertaken to conduct pixel-based rain detection or to correct for the influence of rain on radar images. In addition, as those methods were only validated using the same Decca radar image dataset introduced in Section 2.1, whether they can perform generally well on other datasets remains unknown. Thus, a novel scheme for determining wind parameters from rain-contaminated data with pixel-based rain correction techniques incorporated is worth pursuing.

Following the common practice in radar data processing, the proposed scheme should consist of three major procedures: detecting the presence of rain, mitigating the

influence of rain, and estimating wind parameters from rain-corrected radar images. It was found in [57] that the influence of rain on different portions of the radar image may differ. Specifically, in a rain-contaminated radar image, some regions may still remain unaffected by rain. Hence, rain-contaminated regions need to be identified first before introducing rain mitigation techniques. In Section 3.1 (also [97]), an SOM-based model was proposed to classify between rain-free and rain-contaminated regions in a single radar image with relatively high accuracy. Now this can be incorporated into the first step of the proposed scheme directly. While the method for detecting rain-contaminated regions can be applied directly, no mature rain correction techniques are available for our application. That is because little is known about the dominant backscatter mechanism for X-band marine radars operating at grazing incidence in presence of rain [24], and this hinders the studies concerning correcting the influence of rain on radar images. Although in recent years, many other image correction problems have been successfully tackled using the latest machine learning techniques, a lack of corresponding "ground truth" rain-free images for model training makes it impossible to build the rain correction model from scratch [113].

In contrast to rain mitigation, analyzing and mitigating the influence of haze on terrestrial images is a very active research topic with numerous papers being published in the past decade. In this study, many similarities have been observed between the influence of rain on marine radar images and the influence of haze on terrestrial photography. This enables us to apply image dehazing techniques on rain correction with certain modifications. In particular, a type of CNN-based image dehazing model, named as DehazeNet, proposed in [114] is not only superior to other previously proposed image dehazing methods in performance, but also efficient and easy to use, and this can be considered as the basis of the rain correction model. As for the third step, i.e., wind measurements, in the past few years, machine-learning-based regression algorithms have been introduced to estimate wind speed and significant wave height

from marine radar data [41, 91, 115] with higher accuracy and robustness in comparison with traditional methods. These also can be incorporated into the proposed scheme.

In this chapter, a novel scheme for wind parameter estimation using rain-contaminated X-band marine radar data based on the techniques mentioned above is presented. The remainder of the chapter is organized as follows. A detailed illustration of all the models and algorithms employed in each step of the proposed scheme is given in Section 4.1. Section 4.2 presents an overview of the radar datasets used in this study followed by experimental results obtained from these data. A summary with discussions appears in Section 4.3.

4.1 The Proposed Scheme for Wind Estimation

4.1.1 Framework of the Proposed Scheme

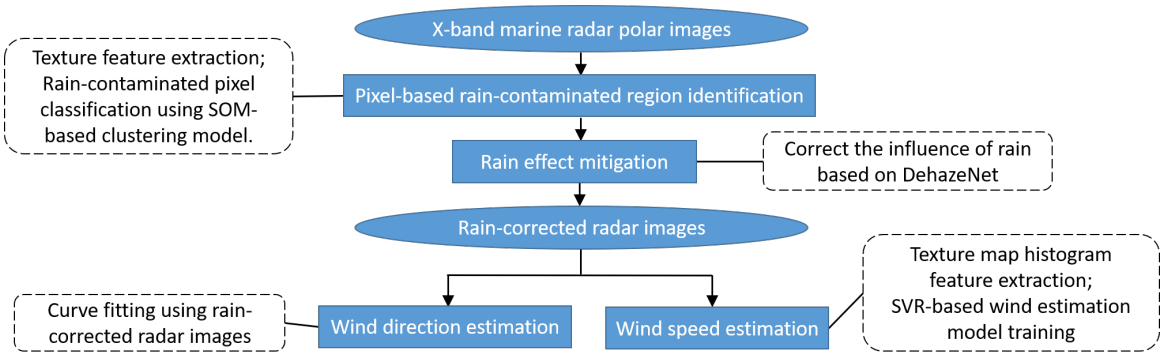

Figure 4.1: A flowchart of the proposed scheme for sea surface wind measurements from rain-contaminated marine radar images.

Fig. 4.1 shows the framework of the proposed method for rain identification, mitigation and wind parameter estimation. Texture features extracted from each pixel of the radar image are input into the SOM-based clustering model introduced in Section 3.1, which generates rain-contaminated region identification results. Images

contaminated by rain are then subjected to the rain correction procedure based on the DehazeNet proposed in [114]. After obtaining rain-corrected radar images, curve fitting techniques are used to measurement wind direction, while a SVR-based model is trained for wind speed estimation.

4.1.2 DehazeNet-based Rain-contaminated Region Correction

In recent years, significant progress has been made on terrestrial image haze removal thanks to the important assumptions about the influence of haze on images proposed in previous studies [116, 117, 118]. Basically, these assumptions assert that haze will change several important characteristics of the radar image, such as reducing the "dark" pixel percentage, local contrast, and local saturation of the image. While there are many differences in the physical process between rain affecting sea surface and radio waves and haze scattering atmospheric light, it has been observed in this study that rain-contaminated radar images also conform to those assumptions, which makes it reasonable to apply image dehazing algorithms on rain correction. Here, the similarities between the influence of rain on marine radar images and the influence of haze on terrestrial photography based on those assumptions are first illustrated below.

1. Reduction of "dark" pixel proportion: Previous statistical observation has found that in most of the haze-free patches of terrestrial photography, their minimum intensities in one or more channels should have very low values or even values close to zero. That assumption is referred to as dark channel prior in [117]. In contrast, patches affected by haze generally look brighter than haze-free ones due to the scattering introduced by the haze. As a result, it is common that none of the channels has pixels with very low intensities. Likewise, for rain-free marine radar image regions, the "dark" pixels located in areas shadowed by surface wave

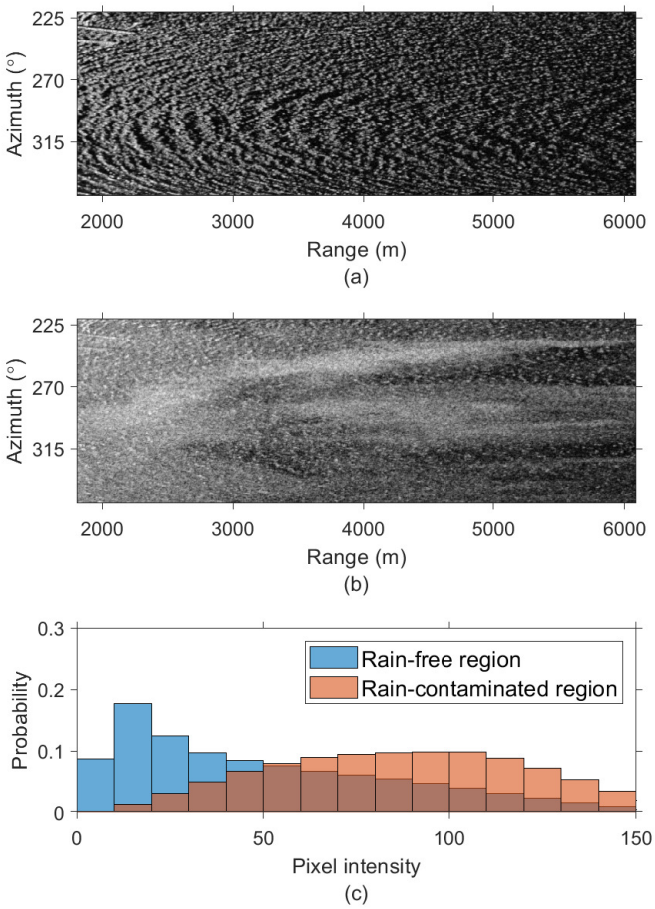

Figure 4.2: Comparison between the normalized histogram of a rain-free and a rain-contaminated region obtained under similar wind speeds. The blue and orange histogram in (c) are obtained from the rain-free (a) and rain-contaminated image (b), respectively. The radar data presented here is described in Subsection 4.2.1.

crests generally have very low backscatter intensities. Consequently, as presented by the blue bins in Fig. 4.2(c), the first bin of the normalized histogram obtained from a rain-free region (i.e., Fig. 4.2(a)) should have a relatively large portion. In contrast, the rain-contaminated region shown in Fig. 4.2(b), it is uniformly bright [119] with a very low value in the first bin of its normalized histogram, as shown by the orange bins in Fig. 4.2(c).

2. Reduction of local contrast: It has been widely acknowledged that terrestrial photography with enhanced visibility (or clear-day images) has higher contrast than images obtained under bad weather such as the presence of haze. Based on

the definition of RGB image contrast proposed in [116], the contrast $\mathrm{Cont}(I)$ of a grayscale radar image $I(m,n)$ can be quantitatively defined as

$$\mathrm{Cont}(I) = \sum_{m,n} |\nabla I(m,n)|, \tag{4.1}$$

where ∇ is the differential operator over azimuth and range coordinate. By observing the radar dataset, the contrast of a radar image contaminated by rain is significantly lower than a rain-free image obtained under similar wind speeds. For instance, the contrast of Fig. 4.2(a) is 65.7% higher than that calculated from Fig. 4.2(b).

3. Reduction of local saturation: Based on the definition of saturation in an RGB image patch proposed in [118], the saturation of a radar image patch $\mathrm{Sat}(I)$ can be expressed as

$$\mathrm{Sat}(I) = \frac{\max(I(m,n)) - \min(I(m,n))}{\max(I(m,n))}. \tag{4.2}$$

As the decline of saturation under the influence of haze has been widely observed, it is also found here that the saturation of a rain-contaminated radar image region is generally lower than a rain-free region obtained under similar wind speeds. For example, for Fig. 4.2(a), the saturation of the region between 275° to 315° in azimuth and 3000 m to 4000 m in range is 16.3% higher than the saturation of the same region in Fig. 4.2(b).

Designed and trained in [114], DehazeNet is a CNN-based end-to-end system for single image haze removal. The reasons of adopting DehazeNet instead of other AI models for rain correction are explained as follows. First, as an end-to-end neural network, DehazeNet, which is both efficient and easy to use, is able to take the whole original radar image as an input and generate its corresponding transmission map

directly without additional processing steps. Second, it has been observed that the effect of haze on terrestrial images and the influence of rain on marine radar images are similar on several assumptions (e.g., the three assumptions enumerated above). Unlike other AI models or previous proposed image dehazing models that are based on a single assumption (e.g., [117] and [116]), DehazeNet takes all of the above assumptions into consideration in its feature extraction step, and this improves the performance of the model in haze removal and rain correction. In addition, compared to other CNNs with classical architecture, several modifications are made in the components and structure of DehazeNet, which further improves the robustness and efficiency of the model. For example, instead of using standard nonlinear activation functions in deep networks such as ReLU, a novel nonlinear activation function, i.e., Bilateral Rectified Linear Unit (BReLU), is proposed and applied in DehazeNet. As ReLU is mainly used for the problems of classification, BReLU is specifically designed for regression problems. This can reduce search space and improve convergence during the training process. Fig. 4.3 shows the framework of the proposed rain correction method based on DehazeNet with each step elaborated below.

Haze-relevant Feature Extraction

As the first step of operation conducted by the DehazeNet, a variety of features based on those haze-related assumptions/priors are extracted through convolution. While DehazeNet differs from other typical CNNs in structure, convolution is still one of the main building blocks. The term convolution refers to the mathematical combination of two functions to produce a third function which merges two sets of information. In the case of a CNN, each convolution operation is performed on the input data with the use of a filter (or kernel) to produce a feature map, which is executed by sliding the filter over the input. At every location, a multiplication between the filter (matrix) and image intensities (matrix) is performed and the result in the feature map is the sum of the element-by-element product. Since the network is originally used to

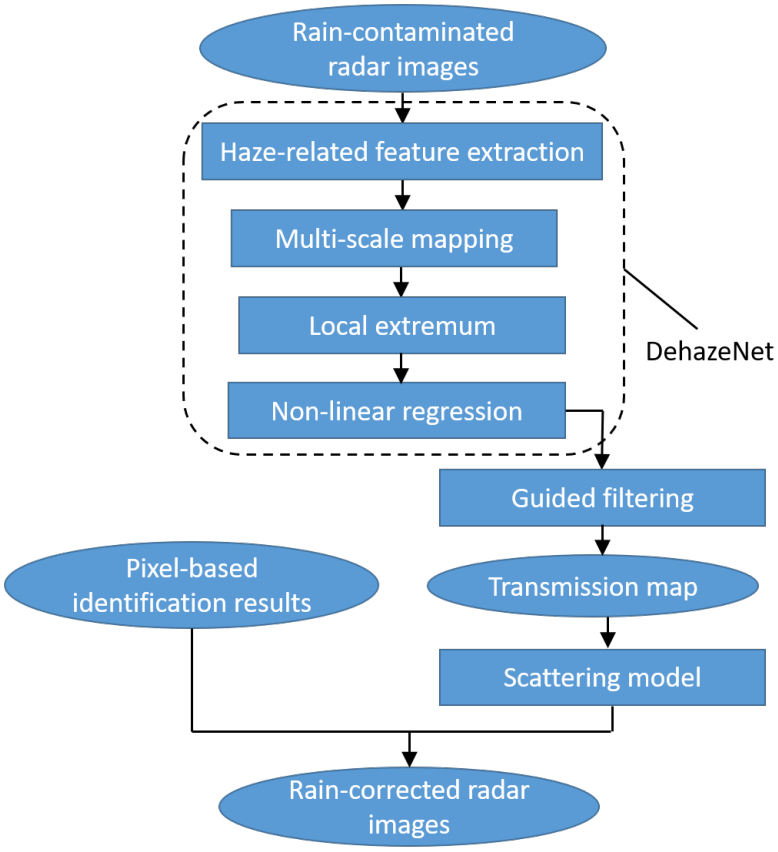

Figure 4.3: A flowchart of the proposed rain correction method based on DehazeNet.

process RGB images, each rain-contaminated radar image is first converted into RGB format by replication before inputting the image into the network. Sixteen $5 \times 5 \times 3$ filters, denoted as $\mathbf{W_1^i}$ $(i = 1, ..., 16)$, are first convolved with the radar image, which generates 16 haze-related feature maps (denoted as $\mathbf{f_1}$) expressed as

$$\mathbf{f_1^i} = \mathbf{W_1^i} * I + \mathbf{B_1^i}, \tag{4.3}$$

where $*$ and $\mathbf{B_1^i}$ represent the convolution operation and the bias term for the i^{th} filter, respectively. Then, an unusual activation function called Maxout unit is introduced to reduce the dimension of the feature maps, which generates a new feature map by taking a pixel-wise maximization operation for every 4 consecutive feature maps,

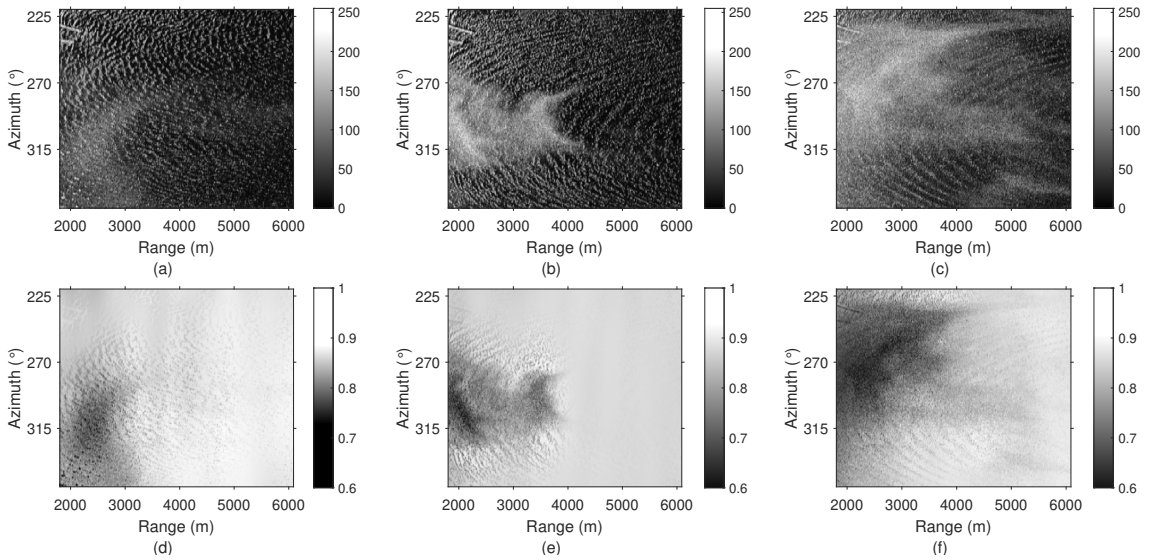

Figure 4.4: (a)(b)(c) Examples of shore-based X-band marine radar images with rain-contaminated regions. (d)(e)(f) The transmission map obtained from (a)(b)(c) using the DehazeNet. The radar data presented here is described in Subsection 4.2.1.

which can be expressed as

$$\mathbf{F_1^k}(m,n) = \max_{i\in\{0,1,2,3\},k\in\{1,2,3,4\}} \mathbf{f_1^{k\times 4-i}}(m,n). \tag{4.4}$$

Therefore, a total of 4 feature maps can be generated after the feature extraction steps.

Multi-scale Feature Generation

In this step, filters with different spatial scales (i.e., 3×3, 5×5 and 7×7) are used to filter $\mathbf{F_1}$ in order to generate multi-scale features, which have been proven effective for haze removal [120]. Since 16 filters are assigned for each selected filter size, a total of 48 feature maps, denoted as $\mathbf{F_2}$ ($i = 1, ..., 48$), can be generated using the convolution operation in Eq. (4.3).

Local Extremum

The max-pooling layer calculates and preserves the maximum value for each patch of

the feature map in order to reduce the number of parameters and computation in the network. In this way, the training time can be shortened and overfitting can be better avoided. However, since the size of the final output image in DehazeNet is supposed to be the same as the input image's, the traditional max-pooling operation is replaced by another type of operation named as local extremum in [114], in which the original image size is retained while the local maximum values can still be extracted. Specifically, for each pixel in the feature map $\mathbf{F_2^i}$, its local extremum equals the maximum value in a 7×7 window centered at this pixel. Thus, the output feature maps $\mathbf{F_3^i}$ $(i = 1, ..., 48)$ obtained from local extremum operation can be expressed as

$$\mathbf{F_3^i}(m_0, n_0) = \max_{m,n \in \Omega(m_0,n_0)} \mathbf{F_2^i}(m, n), \tag{4.5}$$

where $\Omega(m_0, n_0)$ is the 7×7 neighborhood centered at the pixel located at (m_0, n_0). In this way, the size of the output image remains the same as the input image.

Non-linear Regression

Activation functions perform a nonlinear transformation on the input received, which keep values within a manageable range. Since values in the input layers are generally centered at zero and have already been appropriately scaled, they do not require transformation in the first place. However, the range of these values becomes much larger after they are multiplied by weights and summed together during the convolution operation. Thus, the activation functions should be introduced after convolution, which force values back within an acceptable range and make them meaningful. For classic CNNs, an activation function is normally applied to the filtering outputs for non-linear regression and approximation. In the DehazeNet, a novel activation function modified from the ReLU called Bilateral ReLU is used to regress and generate the network output $\mathbf{F_4}$, which can be expressed as

$$\mathbf{F_4} = \min(t_{max}, \max(t_{min}, \mathbf{F_3} * \mathbf{W_4} + \mathbf{B_4})), \tag{4.6}$$

where $\mathbf{W_4}$ and $\mathbf{B_4}$ represent a $6 \times 6 \times 48$ filter and its corresponding bias, respectively. t_{min} and t_{max} are the marginal values limiting the minimum and maximum of the regression output, which equal 0 and 1, respectively.

Rain-corrected Radar Image Generation

As shown in Fig. 4.3, the output image obtained from the DehazeNet, i.e., $\mathbf{F_4}$, is input into the guided filters proposed in [121] for edge-preserving smoothing. The filtering output, denoted as $\mathbf{t}(m, n)$, is called the transmission map, which is used to describe the light portion that is not scattered and reaches the camera in terrestrial photography. In this study, the transmission map generated from a rain-contaminated radar image can be used to reflect the degree of rain influence. That is because it has been observed that regions contaminated by heavy rain (larger than 10 mm/h) generally have much lower corresponding pixel intensities in the transmission map than those less affected or unaffected by rain (see a few examples presented in Fig. 4.4). Hence, the formulation of the atmospheric scattering model obtained from previous studies [122, 123, 116] is introduced to suppress the rain echoes in the original radar image. Its output of each pixel, denoted as $\mathbf{J}(m, n)$, can be calculated as

$$\mathbf{J}(m, n) = \frac{I(m, n) - \max(I)}{\mathbf{t}(m, n)} + \max(I), \tag{4.7}$$

where the intensity range of I has been scaled to $[0, 1]$. Finally, $\mathbf{J}(m, n)$ is combined with the rain-contaminated region identification results ($P(m, n)$, described in Section 3.1) to generate the rain-corrected radar image with smooth transition between rain-free and rain-contaminated regions. Specifically, for each radar image consisting of both rain-free and rain-contaminated regions, if c_f and c_r denote the centroids of rain-free and rain-contaminated regions, respectively, the Euclidean distances between each pixel $I(m, n)$ and c_f, c_r are obtained and expressed as $d_f(m, n), d_r(m, n)$. Each pixel

intensity of the rain-corrected image $R_c(m,n)$ is then calculated as

$$R_c(m,n) = \begin{cases} I(m,n) + \dfrac{d_f(m,n)[\mathbf{J}(m,n) - I(m,n)]}{5[d_f(m,n) + d_r(m,n)]}, \\ \qquad\qquad \text{if } P(m,n) = 0, \\ \mathbf{J}(m,n) + \dfrac{d_r(m,n)[I(m,n) - \mathbf{J}(m,n)]}{d_f(m,n) + d_r(m,n)}, \\ \qquad\qquad \text{if } P(m,n) = 1. \end{cases} \tag{4.8}$$

It should be noted that Eq. (4.8) is only applicable for images with both rain-free and rain-contaminated regions. For a rain-contaminated image without any rain-free regions, its corresponding rain-corrected result equals to $\mathbf{J}(m,n)$ directly. Also, it should be noted that the intensity range of I has been scaled to $[0,1]$.

In [114], the training of DehazeNet has been illustrated in detail and will be introduced briefly here. First, in order to obtain sufficient training data, 10,000 haze-free patches (a patch refers to a subsection of a whole image) are sampled randomly from terrestrial images collected from the Internet. Then, 10 random transmission maps (with values uniformly distributed from 0 to 1) are generated from each patch, which can therefore generate 10 corresponding synthetic hazy patches. Hence, a total of 100,000 artificial hazy patches are generated and used to train the network. A Gaussian distribution function (with a mean value 0 and a standard deviation of 0.001) is introduced to give random values to the filters in each layer of the network, and the initial values of all bias terms are set to 0. The initial learning rate is set to 0.005 and will decrease by half for every 100,000 iterations. In each iteration, a total of 128 patches are utilized for training. Based on the parameter settings mentioned above, the DehazeNet can be trained after a total of 500,000 iterations. It should be noted that since the learnable parameters (i.e., filters and biases) of the network are inherited from the training results obtained in [114], the network can be incorporated into the proposed scheme directly without further training. As we know, training a

deep network requires a large volume of data and the available radar data in this study are much less than the photo data. Training the network only using radar data might cause underfitting of the network. The network should be further trained if more radar data are available in the future.

4.1.3 Wind Parameter Estimation Algorithms

After obtaining the rain-corrected radar image, for each azimuthal direction, its average pixel intensity is calculated and curve fitted using the cosine-squared function in Eq. (1.1). The estimated wind direction corresponds to the azimuth located at the peak of the fitted function. It should be noted that for each rain-corrected radar image, the curve fitting is conducted multiple times (20 in this study), which generates 20 different combinations of coefficients (i.e., a_0, a_1 and a_2) for the fitted function. The combination of coefficients with the highest R-squared (coefficient of determination) value, which reflects how close the data are to the fitted regression line, is selected for producing the final fitted curve. The reason to do so is because during each curve fitting, the value for each coefficient is initialized randomly from the interval $(0, 1)$. As a result, multiple fits using the same data and function might lead to different fitted coefficients. Thus, in order to generate the best fitted curve for each image, it is necessary to repeat the curve fitting multiple times.

Table 4.1: The texture feature maps for histogram feature extraction

Texture Feature Type	Specification	Histogram Intensity Range	Num. of Bins
Gabor features	wavelength is 5.7, orientation is 90°	0 - 1000	20
Gabor features	wavelength is 11.3, orientation is 90°	0 - 2000	20
DWT features	the local standard deviation image of low-low channel	0 - 30	30

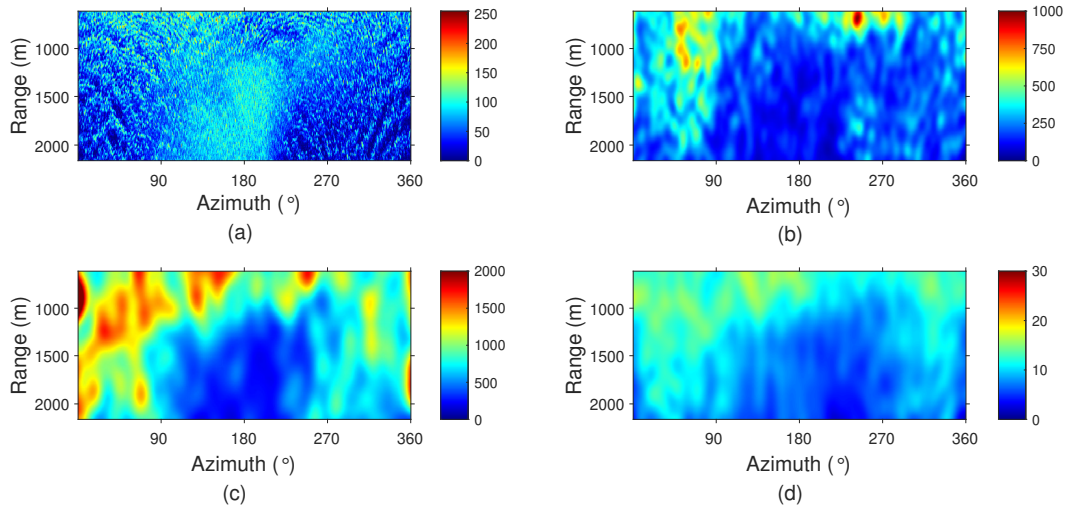

Figure 4.5: (a) Rain-contaminated Decca radar image collected at 9:53, December 01. Anemometer-measured wind speed is 15.6 m/s. (b) The Gabor texture feature map of (a) obtained from a Gabor filter with 5.7 in wavelength and 90° in orientation. (c) The Gabor texture feature map of (a) obtained from a Gabor filter with 11.3 in wavelength and 90° in orientation. (d) The DWT texture feature map of (a) obtained from the local standard deviation image of the low-low channel.

As for wind speed estimation, a SVR-based wind speed estimation model proposed in [91] is modified and trained using the rain-corrected radar images. Specifically, in the feature extraction step, instead of extracting the normalized histogram bin values of the original radar image as in [91], several of the texture feature maps (presented in Table 4.1) introduced in Section 3.1 are generated from the rain-corrected radar image and incorporated into the feature vector. This further improves the accuracy and robustness of the model. The first two feature maps generated by two Gabor filters with the same orientation (90°) and two different wavelengths manifest the clarity of the stripe-like wave patterns in the radar image. Although under rain conditions, the sea surface roughness might be altered and wave patterns in the radar image might be blurred, it has been observed that the wave patterns in rain-contaminated images obtained under high wind speeds tend to be less affected or even unaffected in some image regions. An example of a rain-contaminated radar image obtained under high

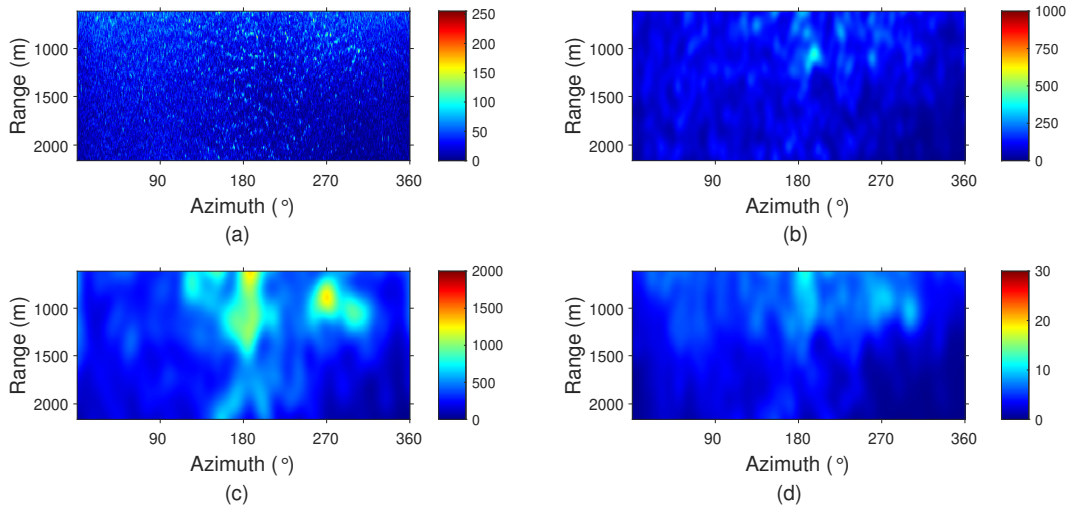

Figure 4.6: (a) Rain-contaminated Decca radar image collected at 3:28, November 27. Anemometer-measured wind speed is 3.1 m/s. (b) The Gabor texture feature map of (a) obtained from a Gabor filter with 5.7 in wavelength parameter and 90° in orientation. (c) The Gabor texture feature map of (a) obtained from a Gabor filter with 11.3 in wavelength parameter and 90° in orientation. (d) The DWT texture feature map of (a) obtained from the local standard deviation image of the LL channel.

wind speed is presented in Fig. 4.5(a). It can be observed that regions with clear wave signatures have relatively high intensities in the corresponding pixels of those two Gabor feature maps, as shown in Figs. 4.5(b) and (c). In contrast, for the rain-contaminated radar image obtained under low wind speed presented Fig. 4.6(a), the wave patterns are blurred significantly in almost all areas. In consequence, the pixel intensities in its corresponding feature maps, i.e., Figs. 4.6(b) and (c) are generally much lower than that of Fig. 4.5(a)'s. As for the third feature map, i.e., the local standard deviation image of the radar image's LL channel generated by DWT, it is used to reflect of the influence of wind speed and rain on sea surface roughness. It is well known that an increasing wind speed leads to the increase in sea surface roughness [124] as well as average radar backscatter intensity [24]. On the other hand, the change of sea surface roughness due to the presence of rain results in the decrease of pixel intensity variation in rain-contaminated images/regions [59]. For instance, it

can be observed from Fig. 4.5(d), that regions that are less affected by rain with high sea surface roughness and simultaneous wind speed have relatively high intensities. In contrast, the pixel intensities of most pixels in Fig. 4.6(d) are very low compared to Fig. 4.5(d)'s because the sea surface roughness is altered significantly by rain with wave patterns barely observable in almost all areas in Fig. 4.6(a), which also causes fairly uniform pixel intensities in the radar image. Bin values are extracted from the histograms of the texture feature maps presented in Table 4.1. Thus, the feature vector extracted from each image consists of 70 elements.

As we know, in most linear regression models, the objective is to minimize the sum of squared errors. However, in many practical problems based on real data such as wind speed estimation, we are only concerned about reducing error to a certain degree as long as they fall within an acceptable range. As the application of SVM on regression problems, SVR provides us with the flexibility to define how much error is acceptable in our model and find an appropriate hyperplane in higher dimensions to fit the data. Instead of minimizing the squared error in linear regression, the objective function of SVR is to minimize the coefficients i.e., the l_2-norm of the coefficient vector (denoted as $||\mathbf{w}||$ in Section 2.1). The error term is instead handled in the constraints, where we set the absolute error less than or equal to a specified margin, called the maximum error (denoted as ϵ). We can tune ϵ to gain the desired accuracy of our model. Therefore, the goal of training is to find a function that has at most ϵ deviation from anemometer-measured wind speed while minimizing the norm of $\mathbf{w}$ at the same time [125]. Similar to [91], the classic ϵ-SVR algorithm proposed in [125] is employed to train the wind speed estimation model. Thus, the function can be expressed as

$$y' = \mathbf{w}^T \psi(\mathbf{x}) + b, \tag{4.9}$$

where y' is the radar-derived wind speed and $\psi(\cdot)$ is a Gaussian kernel function that maps the input texture feature vector $\mathbf{x}$ into a higher dimension space for linear

separation. Similar to Eq. (2.3), $\mathbf{w}$ is the weight of $\psi(\mathbf{x})$ and b is the bias term. Finding such a function is equivalent to solving the following convex optimization problem, which looks similar to Eq. (2.5) as

$$\min \left(\frac{1}{2} \|\mathbf{w}\|^{\mathbf{2}} + C \sum_{i=1}^{l} (\xi_i + \xi_i^*) \right) \tag{4.10}$$

subject to

$$\begin{cases} y_i - y_i' \leq \epsilon + \xi_i, \\ y_i' - y_i \leq \epsilon + \xi_i^*, \\ \xi_i, \xi_i^* \geq 0 \end{cases} \tag{4.11}$$

where y_i' and y_i denote the radar-derived wind speed and anemometer-measured wind speed for the i^{th} sample, respectively. ξ_i and ξ_i^* represent the values of estimation errors that exceed either of the two margins around the hyperplane, $\mathbf{l}$ is the total number of training samples. Since the existence of such errors is common when using real data for training, it is necessary to add the error terms into our objective function so that they can be minimized as much as possible. C is a positive constant referred to as the box constraint, which determines the penalty imposed on y_i' that lies outside the ϵ margin and helps avoid overfitting. In this work, the Regression Learner toolbox from MATLAB is used to train the SVR-based wind estimation model, in which the SMO algorithm is adopted again for solving the convex optimization problem.

4.2 Experimental Results

4.2.1 Data Overview

Datasets collected from the two radar systems (i.e., Decca and Koden) introduced in Subsection 3.6.1 are used to validate the effectiveness of the proposed scheme. The radar information can be found in Table 3.4. For the Decca radar dataset, by selecting

the first pulse radar image generated per minute, a total of 3902 polar images were collected during several time periods from November 26 to December 4, 2008. The presence of rain during data collection period can be indicated by the simultaneous rain rate information collected by two rain gauges installed on the ship.

Table 4.2: Dataset information

	Decca	Koden
Collection location	~300 km south-southeast of Halifax, NS, Canada (42°30 N, 62°05 W)	USCG Station Yaquina Bay, Newport, OR, USA (44°62 N, 124°06 W)
Collection period	Nov. 26 - Dec. 4, 2008	Jan. 11 - Jul. 18, 2019
Wind speed range (m/s)	2.1 - 15.4	0 - 19.0
Ship speed range (m/s)	0 - 6.8	N/A
Significant wave height range (m)	1.5 - 5.0	0.6 - 7.2
Rain rate range (mm/h)	0 - 50.0	0 - 33.5
Number of images	3902	980

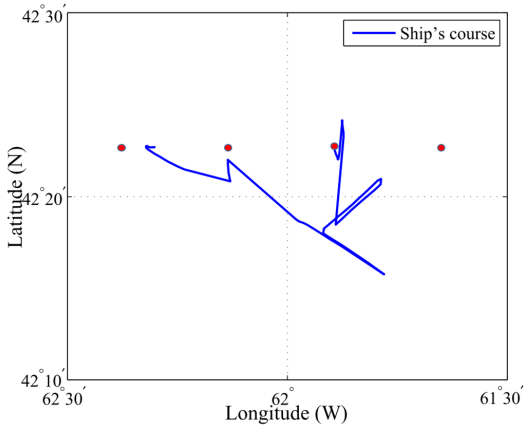

Figure 4.7: Ship's course and available measurement points of rainfall.

For the Koden dataset, a total of 980 pulse images randomly collected from January 11 to July 18, 2019 with various time intervals are selected for wind parameter estimation. Since nearly half of the radar coverage area consists of land, the radar

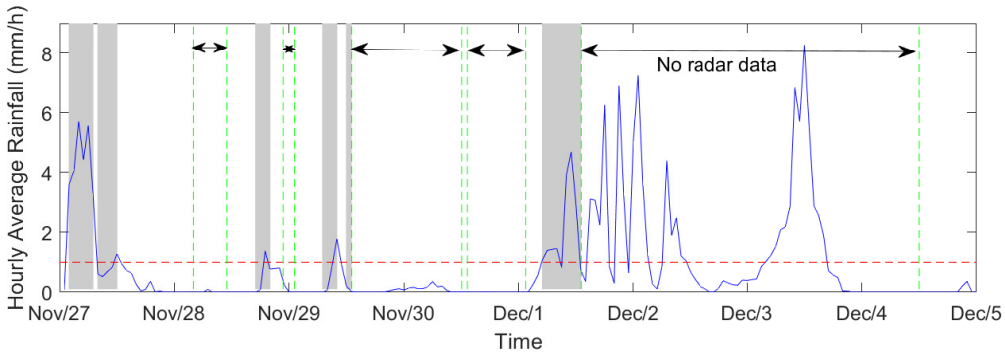

Figure 4.8: Hourly average rain rate (dots connected by blue lines) values provided by ECMWF during Decca radar data collection periods. Simultaneous radar images with rainy regions detected by the proposed method are indicated by gray shadows.

image only preserves azimuths that point towards the sea surface. The ground truth wind information was reported hourly by the weather station NWPO3 near the base of South Jetty of Yaquina Bay Inlet. It should be noted that the provided wind speed has already been averaged over a two-minute period, while wind direction was reported in 10 degree increments. In addition, rain rate data was obtained from rain station AS512 (ocrg.org) in the Newport residential area North of Yaquina Bay. It can be observed from Table 4.2 that both datasets cover a relatively wide range of wind speeds, rain rates, and wave heights (obtained from buoys deployed near radar sites), which permits the proposed method to be evaluated under various sea states and rain conditions.

4.2.2 Rain Detection Results

In order to investigate the accuracy of the proposed scheme in rainy image detection, the global rainfall data provided by ECMWF are used as ground truth for comparison since they are derived from the advanced precipitation hindcasting techniques. Each value in the data corresponds to the accumulated rainfall in the previous hour (i.e., hourly average rain rate, mm/h) obtained from a certain measurement point. The distance between each adjacent measurement point (indicated by red dots in Fig. 4.7)

in space is 0.25° in either latitude or longitude (around 27 km). Therefore, since the Decca radar was installed on a moving ship whose course is shown in Fig. 4.7, the hourly average rain rate is acquired from the measurement at the red dot that was closest to the ship location at that time. In Fig. 4.8, the hourly average rain rate values during the sea trial are depicted by the blue dotted line. It should be noted that the radar data were not collected in several time periods during the sea trial and those periods are indicated by the black arrow lines between the green dash lines. In order to show whether the proposed model can detect rainy images accurately using the Decca radar data, the identification of rainy images using the proposed method is indicated by the gray shadows in Fig. 4.8. It can be observed that when hourly average rain rate is greater or equal to 1 mm/h, the presence of rainy images can always be detected (i.e., a detection success rate of 100%) among the 60 radar images obtained in each hour using the proposed model. On the other hand, when the hourly average rain rate is nonzero but less than 1 mm/h, rainy images might not be identified. That is because when the average rain rate is low, the presence of rainfall might be sparse in either time or space and is not within the radar coverage. Another possible reason is that under certain sea states (e.g., high wind speeds), the radar backscatter might not be altered significantly under low rain rates. As for the Koden radar data, it is not used to validate the rain detection accuracy because the time interval between each provided image is longer than an hour. Due to the fact that the rainfall distribution can be non-uniform in both time and space, as well as the fact that the radar coverage is much smaller than the spatial interval of each measurement point, it is likely that while rainfall was recorded, the only available radar image during a particular hour might not be affected by rain. Nevertheless, if more Koden radar images are available within a single hour, it is expected that the rainy images would be easily identified by the model.

4.2.3 Wind Direction Results

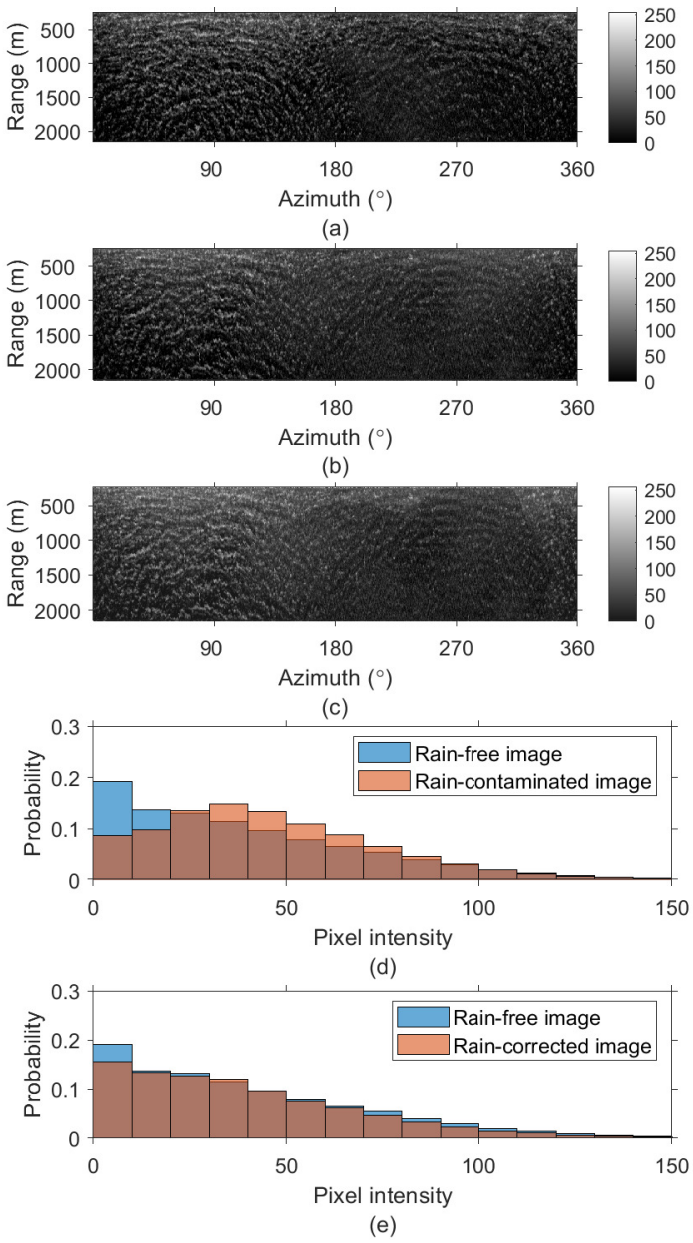

Figure 4.9: An example of a rain-free (a), a rain-contaminated (b) and its corresponding rain-corrected radar image (c) obtained in the same region under similar wind speeds. Comparisons of their normalized histograms are presented in (d) and (e).

The rain-correction result of Fig. 4.9(b) is presented in Fig. 4.9(c) and compared with Fig. 4.9(a), i.e., a rain-free radar image obtained from the same region under similar wind speeds. As shown in Fig. 4.9(e), the differences of normalized histogram bin

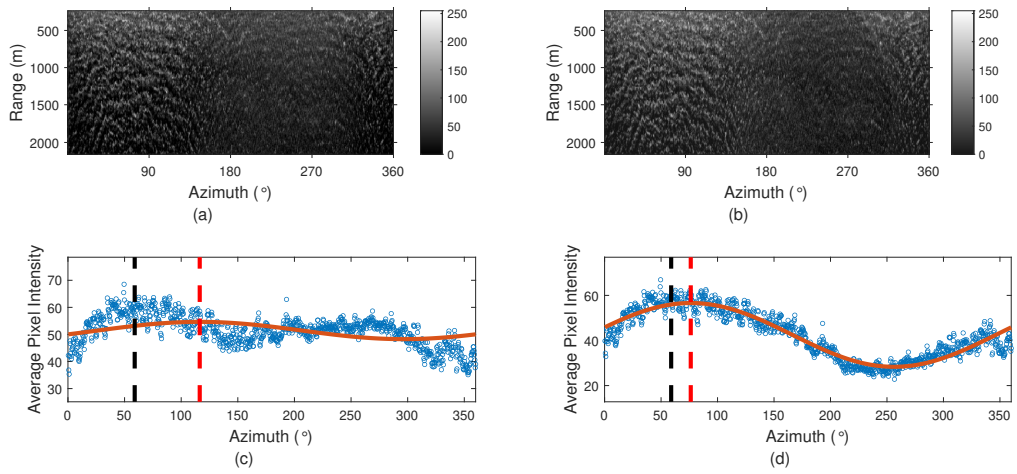

Figure 4.10: (a) An example of a rain-contaminated Decca radar image. (b) The rain-corrected image obtained from (a) using the proposed method. (c) Wind direction estimation result using (a). (d) Wind direction estimation result using (b). The blue data points refer to the average pixel intensity in each azimuth of the radar image, while the orange line is the corresponding best-fit curve of the blue data points using the cosine-squared function. Red and black dash line indicate radar-derived wind direction and anemometer-measured wind direction, respectively.

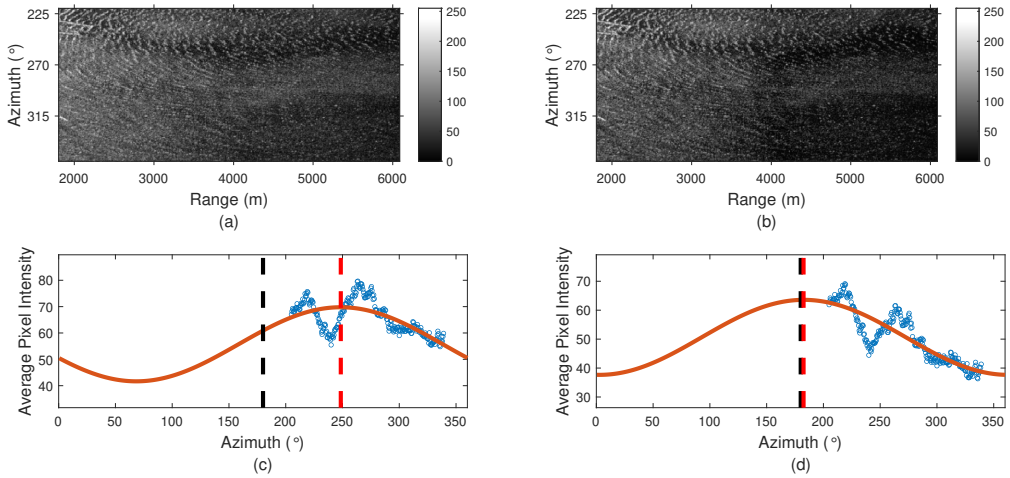

Figure 4.11: (a) An example of a rain-contaminated Koden radar image. (b) The rain-corrected image obtained from (a) using the proposed method. (c) Wind direction estimation result using (a). (d) Wind direction estimation result using (b). The blue data points refer to the average pixel intensity in each azimuth of the radar image, while the orange line is the corresponding best-fit curve of the blue data points using the cosine-squared function. Red and black dash line indicate radar-derived wind direction and anemometer-measured wind direction, respectively.

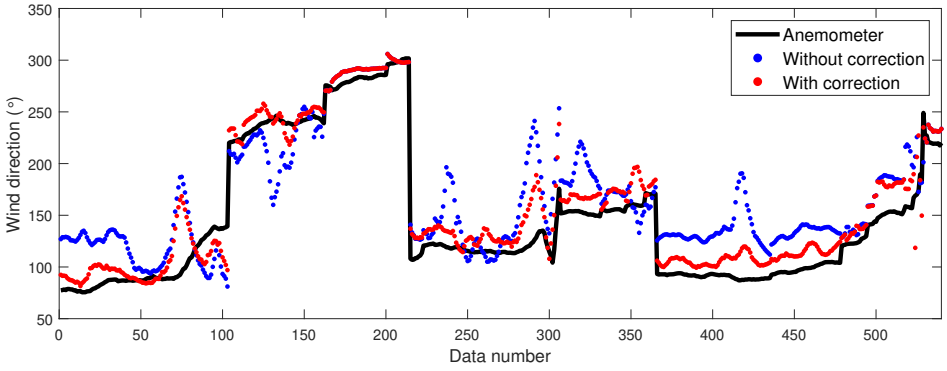

Figure 4.12: Comparison of the sequences of Decca wind direction results.

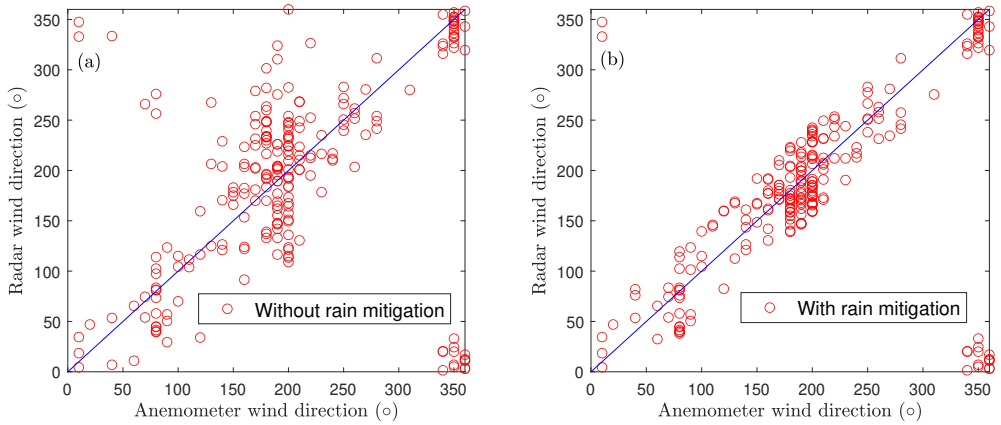

Figure 4.13: Comparison of wind direction estimation results using (a) rain-contaminated Koden radar images without correction and (b) rain-corrected Koden radar images.

values between two types of images are very small for most of the bins (no larger than 0.04), which indicates that both the rain-free and rain-corrected images have similar characteristics in pixel intensity distribution. Figs. 4.10 and 4.11 show the comparison of wind direction estimates using, respectively, the original rain-contaminated radar images without and with rain mitigation. As shown in Fig. 4.10(b) and Fig. 4.11(b), although it is impossible to fully recover the wave signatures contaminated by rain, the noise introduced by rain has been effectively suppressed. It can also be observed that the estimation errors can be reduced significantly for both the Decca (Fig. 4.10) and Koden data (Fig. 4.11) after introducing the proposed rain mitigation method.

Thus, in order to evaluate the effectiveness of the proposed method on rain mitigation quantitatively, for the Decca radar data, all images are first input into the SOM-based identification model. In consequence, a total of 540 images that contain rain-contaminated regions are selected for wind estimation study. Fig. 4.12 shows the sequences of wind direction estimation results using rain-contaminated Decca radar images. As the time interval between each image in the same time period is one minute, both the anemometer and radar results are moving-averaged over 10 min. The RMSD between wind directions derived using the proposed scheme (i.e, the red dots in Fig. 4.12) and measured by the anemometers (i.e., black line in Fig. 4.12) is 18.6°. In contrast, if the original rain-contaminated images are used directly for curve fitting without any correction, the RMSD is as high as 37.7°, as indicated by the blue dots in Fig. 4.12. Hence, the proposed method significantly improves the wind direction estimation accuracy with a reduction of 19.1° in RMSD.

As for the Koden radar data, the region presented in Fig. 4.11(a) is first cropped from the original radar image in order to avoid the influence of the shore on the sea clutter. As a result, the azimuth and range coverages of the region are around 135° and 5130 m, respectively. It should be noted that the images obtained under heavy precipitation or very low wind speeds ($\leq$2 m/s) with little or no wave signatures are excluded from wind direction estimation in order to ensure that all testing images contain sufficient electromagnetic backscatter from the ocean surface [126]. As the acquisition time interval between each image is relatively large, the wind direction estimates obtained from 230 rain-contaminated images identified by the SOM-based model are presented as scatter plots in Fig 4.13. Specifically, for Fig. 4.13(a), the curve fitting is applied to rain-contaminated radar images directly without correction, while Fig. 4.13(b) shows the results obtained from the rain-corrected images. Since the RMSDs between radar wind direction and anemometer wind direction calculated from Fig. 4.13(a) and (b) are 47.5° and 23.6°, respectively, the proposed scheme

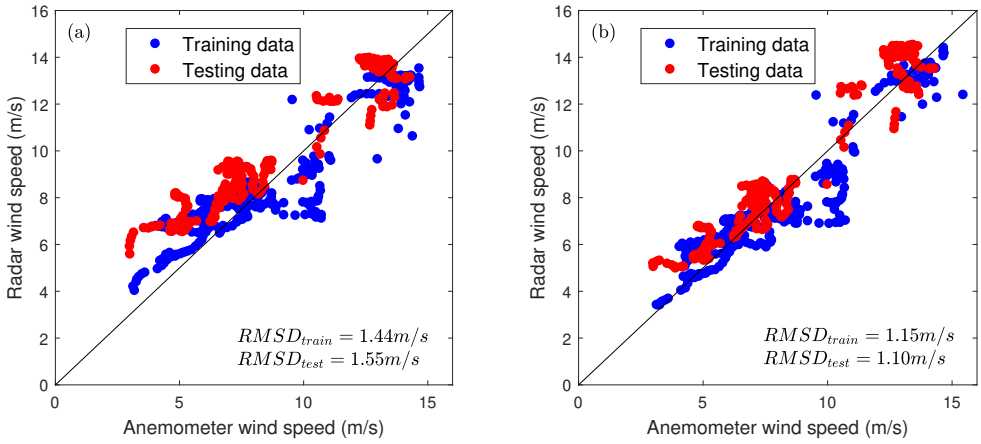

Figure 4.14: Comparison of wind speed estimates using (a) rain-contaminated Decca radar images without correction and (b) rain-corrected Decca radar images.

improves the estimation accuracy significantly, with a reduction of 23.9° in RMSD. Although the RMSD obtained from Koden images is 5.0° higher than using Decca images, it should be noted that the azimuth coverage in Koden images is only 135°, which indicates that it is more difficult to fit an accurate curve for wind direction estimation.

4.2.4 Wind Speed Results

For both datasets, nearly half of the rain-contaminated radar images are selected as training samples for the SVR-based wind speed estimation models. It should be noted that those samples should be obtained under a relatively wide range of wind speeds in order to ensure the models' robustness. The other half of the rain-contaminated images are all used to test the models' accuracy. In addition, in order to validate the effectiveness of the proposed rain mitigation method, for each dataset two models are trained and tested separately using the original and rain-corrected radar images, respectively. The wind speed estimates obtained from the Decca radar data are shown in Fig. 4.14 with both the anemometer and radar results being moving-averaged over 10 min. By comparing the results presented in Fig. 4.14(a) and (b), it can be observed

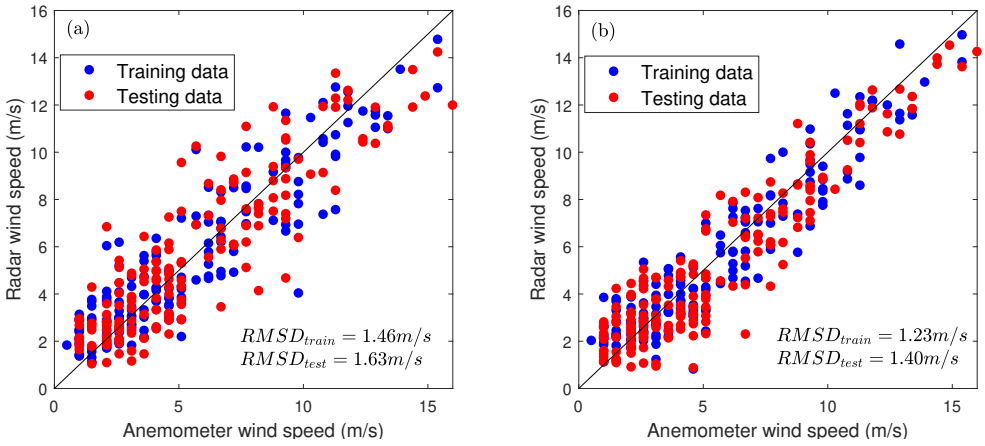

Figure 4.15: Comparison of wind speed estimates using (a) rain-contaminated Koden radar images without correction and (b) rain-corrected Koden radar images.

Table 4.3: Wind estimation statistics results

Dataset	Wind direction				Wind speed							
	Without correction		With correction		Without correction				With correction			
	RMSD (◦)	CC	RMSD (◦)	CC	Training		Testing		Training		Testing	
					RMSD (◦)	CC	RMSD (◦)	CC	RMSD (◦)	CC	RMSD (◦)	CC
Decca	36.7	0.89	18.6	0.98	1.44	0.88	1.55	0.93	1.15	0.90	1.10	0.94
Koden	47.5	0.88	23.6	0.96	1.46	0.91	1.63	0.90	1.23	0.94	1.40	0.93

that the proposed rain correction method further improves estimation accuracy in both training and testing samples. In particular, the RMSD between testing images and anemometer measurements is reduced by 0.45 m/s.

As for the Koden radar data, since the texture features extracted from each image are no longer dependent on the azimuthal intensity information, images obtained under low wind speeds (except zero wind speed) may still be utilized for wind speed estimation. Thus, a total of 350 rain-contaminated images identified by the SOM-based model are used to train and test the proposed wind speed estimation model. The estimates obtained from models using original radar data (i.e., rain-contaminated images without correction) and rain-corrected images are presented in Fig. 4.15(a) and (b), respectively. Hence, it can be concluded that the proposed rain mitigation method improves both training and testing accuracy, with a reduction of 0.23 m/s in RMSD for both samples. Nevertheless, it should be noted that the RMSD obtained

from the Koden radar data is generally higher than that of the Decca radar data. One reason may be due to the fact that compared to the Decca radar data, a relatively large portion of rain-contaminated Koden images were obtained under low wind speeds ($\leq$4 m/s) with little wave signatures. In consequence, texture features extracted from those images are more likely to be similar to each other, which makes it hard to provide highly precise estimation. Thus, it can be concluded that although the texture-feature-incorporated SVR-based wind speed estimation method is already more accurate than previous proposed models, introducing rain correction procedure will further improve measurement accuracy. To sum up, the RMSDs and correlation coefficients (CCs) of the wind speed and direction results for different schemes and radar data sets are shown in Table 4.3. It can be clearly observed that all the CC values are improved after introducing the proposed rain correction techniques. In the end, in order to ensure the proposed scheme can be implemented in real-time, the testing was conducted in MATLAB R2019b installed on a normal PC running Windows 10 with a 3.0 GHz Intel Core i5-2320 CPU, 8 GB memory, and a 64-bit operating system. For each rain-contaminated radar image it took an average of around 45 s to conduct rain correction and generate wind estimates.

4.3 Chapter Summary

In this chapter, a novel scheme to mitigate the rain influence and estimate sea surface wind parameters from rain-contaminated marine radar images is proposed. In order to detect the presence of rain, texture features extracted from radar data are input into a SOM-based model, which generates pixel-based rain identification results. Then, an image dehazing CNN called DehazeNet is applied to images that contain rain-contaminated pixels, which produces a transmission map reflecting the degree of rain influence. The influence of rain can therefore be corrected by inputting both the

transmission map and the original image into the atmospheric scattering model. After obtaining rain-corrected radar images, wind direction can be estimated by curve fitting on average azimuthal intensities. As for wind speed, normalized histogram bin values extracted from three texture feature maps are combined as feature vectors to train the SVR-based wind speed estimation models.

Both shipborne (Decca) and shore-based radar (Koden) datasets are employed to validate the proposed scheme, with simultaneous anemometer measurements used as ground truth to evaluate estimation accuracy. In addition, in order to validate the effectiveness of the proposed rain correction method, wind measurements using the original rain-contaminated images without correction are also conducted for comparison. For wind direction estimation, the RMSDs obtained using the proposed scheme are 18.6° for the Decca radar images and 23.6° for the Koden radar images, respectively. Compared to the results obtained without applying rain correction, the estimation errors are reduced significantly, with RMSDs decreased by 19.1° and 23.9° for the Decca and Koden data, respectively. As for wind speed estimation, for both datasets, two SVR-based models are trained and tested separately under a wide range of wind speeds, with one using rain-contaminated images without correction and another using rain-corrected images, respectively. Results show that compared to the SVR-based models trained using the original radar images, the rain-corrected SVR-based models further improve both training and testing accuracy. Specifically, the RMSDs calculated from rain-corrected testing samples are 1.10 m/s and 1.40 m/s for the Decca and Koden data, respectively, which are 0.45 m/s and 0.23 m/s lower than that obtained from the original rain-contaminated images. Since rain echoes are more likely to overwhelm wave signatures under low wind speeds, the proposed method produces better results at higher wind speeds than lower wind speeds. In addition, as it takes around 45 s to obtain rain correction and wind estimates from each rain-contaminated radar image, the proposed scheme can be implemented in real

time.

As the first work to apply pixel-based rain identification model and image dehazing techniques for wind parameter measurements using X-band marine radar images, the proposed method effectively improves the accuracy of wind estimation from rain-contaminated data. Also, this is the first method validated using rain-contaminated shore-based marine radar images with limited azimuthal coverage. Due to similar imaging mechanisms, it is also possible to extend the proposed method to radar images captured at other frequency bands such as S-band, which is another common frequency band for commercial marine radars. Although compared to the X-band radar, the S-band radar gives a coarser spatial resolution, previous studies have found that the influence of rain on S-band radar images is less severe than that on X-band radar images because the signal attenuation in rain of S-band radar is smaller than X-band radar [127].

Chapter 5

Significant Wave Height Estimation Using Convolutional Gated Recurrent Unit Network

As illustrated in Section 1.2, although CNN is good at exploiting spatial patterns, it cannot observe temporal evolution of radar image sequences. As a class of neural networks that allow previous outputs to be used as inputs while having hidden states, recurrent neural networks (RNNs) have been widely applied in time sequence analysis problems [128]. Since the connections between nodes (cells) of the RNN form a directed graph along a temporal sequence, temporal dynamic behavior can be observed. As a subclass of RNNs, the gated recurrent unit (GRU) proposed in [129] performs superbly in capturing long term dependencies and greatly alleviates the gradient vanishing problem occurred in the naive RNN unit. Thus, in this chapter, other than the CNN-based model, a novel SWH estimation model combining deep convolutional layers with GRU network (abbreviated as CGRU) is proposed. Both spatial and temporal features can be extracted from the radar image sequence while the end-to-end characteristic of CNNs is maintained. Section 5.1 gives an overview of the radar and

environmental data used in this work. The structure and components of the proposed CNN and CGRU-based models are illustrated in Section 5.2. The training and testing results obtained from the shipborne marine radar data using SNR (i.e., Eq. (1.8)), CNN, and CGRU-based models are presented and compared with each other in Section 5.3. Finally, the conclusion and outlook of this work appear in Section 5.4.

5.1 Data Overview

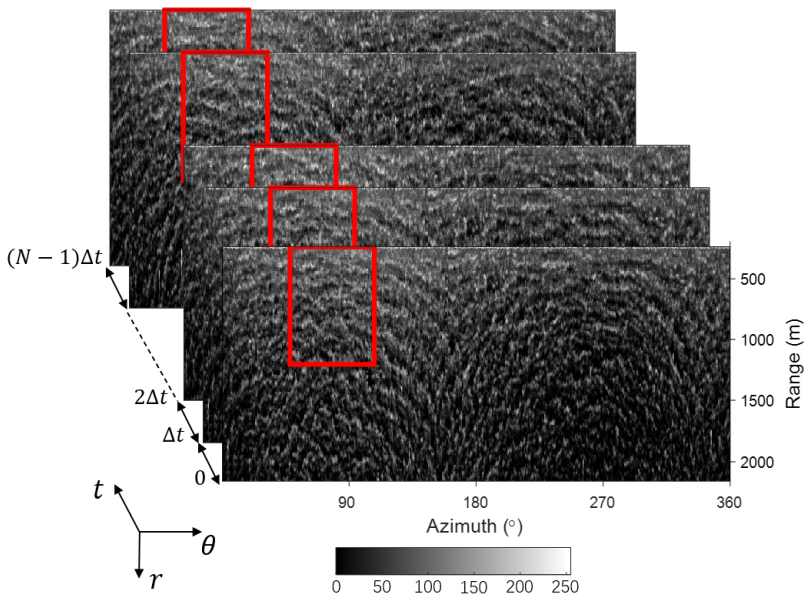

Figure 5.1: An illustration of a sequence with N polar radar images in a file. For each image, the data within the red rectangular region are used as input into the proposed networks.

The shipborne Decca radar data used in this work has already been introduced in Section 2.1.2. Fig. 5.1 is an example of the polar radar image sequence in one file. Since the antenna rotation speed is 28 rpm, the time to generate each image is around 2.19 s (denoted as Δt in Fig. 5.1). Also, as each file consists of 32 images, here N equals to 32. The reference SWH data were provided by three free-floating Triaxys directional wave buoys as well as a 6-meter NOMAD buoy (owned and maintained by Environment and Climate Change Canada) deployed approximately 10 km away

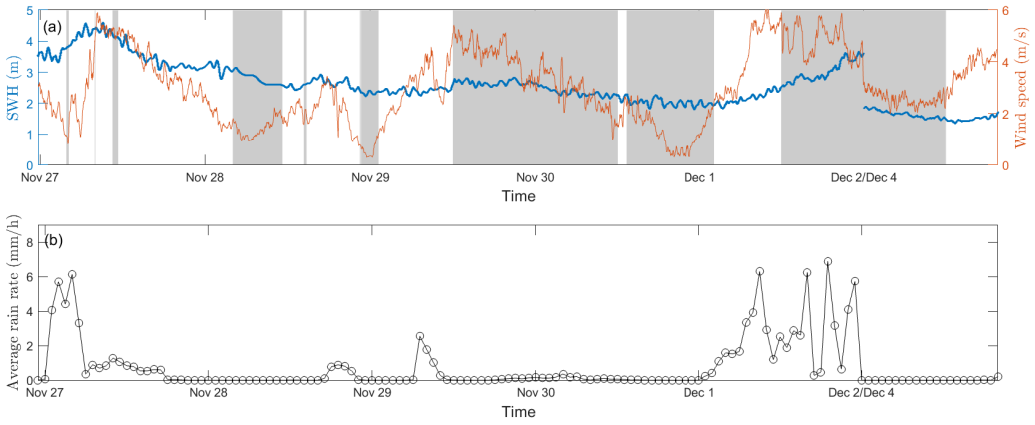

Figure 5.2: (a) The simultaneous anemometer-measured wind speed and buoy-measured interpolated SWH data. Note that no radar data were collected during time periods in gray shadows. Also, no data were collected between December 02 to December 04. (b) Simultaneous hourly average rainfall values (dots connected by black lines) provided by ECMWF.

from the radar. Since one measurement is obtained approximately every 10 minutes, temporal interpolation based on bilinear interpolation is conducted in order to provide simultaneous SWH reference for each radar image sequence. As shown in Fig. 5.2, the SWHs during data collection periods range from 1.3 to 4.6 m. On the other hand, the simultaneous local wind speeds (measured by two anemometers deployed on the vessel) range from 2 m/s to 14 m/s. This permits the proposed method to be evaluated under typical wind and wave conditions. The reason why the wave heights do not follow the changes in winds is that wind speed over the ocean surface has a delayed effect on the highest one-third of the waves for higher sea states [130]. For example, while the wind speed increased from 6 m/s to more than 13 m/s during the time period between 1:27 to 12:02, December 01, the simultaneous SWH were consistently low. Nevertheless, it gradually increased to around 4 m at 23:37, December 01. An abrupt reduction of about 2 m in SWH in Fig. 5.2(a) is due to a two-day time gap in data collection. Specifically, since no radar data were collected on December 02 and December 03, the wind and wave data of these two days are not shown in Fig. 5.2(a). As a result, the abrupt reduction in SWH appears when the data points at 23:59,

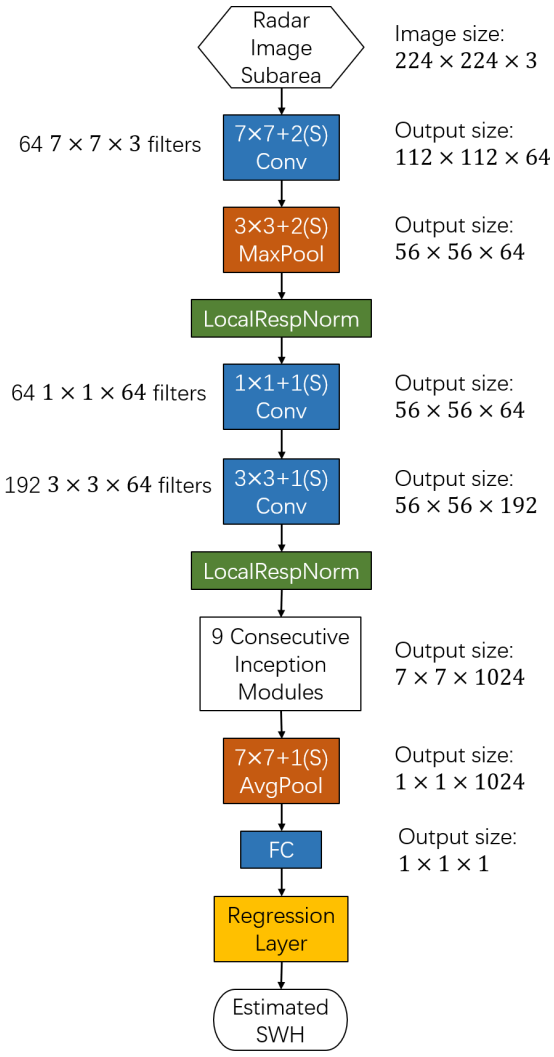

Figure 5.3: Schematic diagram of the CNN-based SWH estimation model.

December 01 and 00:00, December 04 are displayed next to each other. Nevertheless, in other time periods, there could be many other reasons why wind speed and SWH do not correlate such as swell dominated seas or ship superstructure interference with the anemometer measurements (from certain directions).

5.2 Methodology

In this section, the two SWH estimation models proposed will be introduced.

5.2.1 CNN-based SWH Estimation Model

Before introducing the CGRU-based model, the CNN-based model is first designed and introduced for validation and comparison purposes. In the past decade, various types of classic deep CNNs have been proposed for image classification. Instead of designing and training a brand new CNN from scratch, one of those classic CNNs is leveraged as a pretrained network to build the proposed CNN-based SWH estimation model for the following reasons. First, designing a CNN from scratch is time-consuming as there are no specific rules to determine the optimal structure and parameters of a CNN. Second, training a CNN from scratch requires a large amount of radar data (much more than the data available in this study) and more iterations to converge sufficiently. In contrast, fine tuning on the weight values of the pretrained network requires less training data with shorter convergence time. In this study, the convolutional layer structure and original weight values are obtained from the pretrained GoogLeNet proposed in [131]. Compared to other classic CNNs such as AlexNet [132] and Vgg-16 [133], GoogLeNet has a much smaller network size and consumes much less training time while obtaining higher training accuracy. In order to ensure the input image consists of strong backscatter signals with distinct wave signatures, a subarea that covers $\pm 30^\circ$ around upwind direction and 240-1200 m in range is selected from the whole radar image. The subarea (the red rectangle in each image shown in Fig. 5.1) is then resized and duplicated into RGB channel. In consequence, each subarea image has a uniform size of 224×224 with 3 channels. A schematic diagram of the CNN-based SWH estimation model is presented in Fig. 5.3 and the design and function of each type of component are briefly introduced below:

- **Convolutional layer (Conv)**: In order to produce deep spatial features from the radar image, multiple convolutional layers are added throughout the network. The filter number and kernel size in each convolutional layer are listed on the left

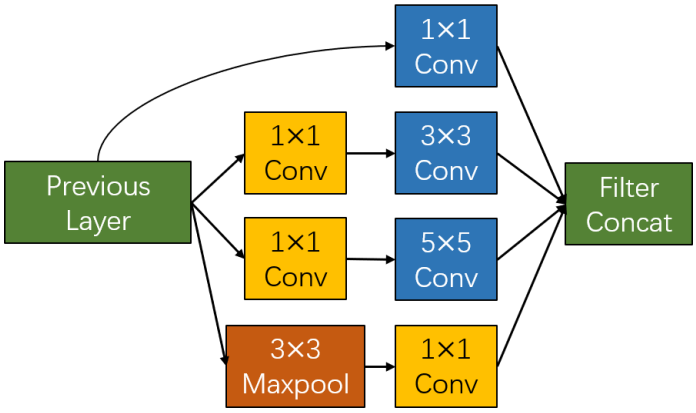

Figure 5.4: The structure and component of each inception module in the proposed CNN-based model.

side of the network diagram. A feature map can be generated by sliding a certain filter over the input. Specifically, in each convolution, a matrix multiplication is performed between the filter and the elements at the same location. The result in feature map is the sum of the element-by-element product. It should be noted that the value before "(S)" in each convolutional block in Fig. 5.3 refers to the number of elements by which the filter shifts after each convolution (also known as stride). It can be inferred that if the stride is 1, the size of the output remains unchanged. If the stride is 2, the output size is half as that of the input's. The size of the output feature maps from each convolutional layer is listed on the right side in the network diagram. Finally, each feature map is passed through a rectified linear unit (ReLU), which is a piecewise ramp function that will output the input directly if it is positive. Otherwise, it will output zero. The reason for doing so is to prevent the exponential growth in the computation required to operate the CNN-based model.

- **Pooling layers (MaxPool and AvgPool)**: The max pooling (MaxPool) operation can be regarded as the convolution between each non-overlapping subregion of the input and a max filter, which preserves the maximum value within the subregion while discarding the others. For each max pooling layer, the size of

the subregion and the amount the filter shifts after each operation is shown in Fig. 5.3. As a downsampling strategy, the max pooling is done for the following purposes. First, by providing an abstracted form from the input, the overfitting problem occurring in model training can be alleviated. Second, as the spatial size of the representation has been reduced, the number of parameters and computational cost of the network can be reduced as well. Compared to max pooling, average pooling (AvgPool) calculates the average value rather than the maximum value of the elements within the subregion, and this maintains retention of information about all the elements.

- **Local response normalization layer (LocalRespNorm)** is added to constrain the unbounded activation function (e.g., ReLU) from increasing the output layer values. As a cross-channel operator, it replaces each element with a normalized value calculated from the elements across a certain number of neighboring channels within the normalization window. That is, for each element x_{in} in the input, its output value x_{out} after normalization can be obtained using

$$x_{out} = \frac{x_{in}}{(K_L + \frac{\alpha_L \times \mathbf{ss}}{\text{windowChannelSize}})^{\beta_L}}, \tag{5.1}$$

  where $K_L = 1$, $\alpha_L = 10^{-4}$, and $\beta_L = 0.75$ are all hyperparameters in the normalization and **ss** is the sum of squares of the elements in the normalization window. Note that the size of the normalization window is set as 5.

- **Inception module**: In order to get better performance, most previously proposed popular CNNs simply stacked convolutional layers deeper and deeper, which are prone to overfitting and computationally expensive. Also, it is hard to determine the optimal kernel size for the convolution operation. Thus, in order to solve these problems, a novel structure, named the inception module, was introduced in GoogLeNet. This conducts multi-scale covolution with much

smaller parameter size and computation cost. Fig. 5.4 shows the structure of an inception module with 3 different sizes of filters (i.e., the 1×1, 3×3, and 5×5 convolutional layers in blue) incorporated, which enables the extraction of both global and local spatial features from the radar image. Besides, max pooling is also performed. The output feature maps are concatenated and sent to the next inception module. Note that an extra 1×1 convolution (indicated by the yellow component in Fig. 5.4) is added before or after each of the above mentioned layers. This plays an important role in reducing computational cost by limiting the number of input channels. As shown in Fig. 5.3, a total of 9 inception modules are connected one by one, with max pooling layers added between each of the neighboring modules. Detailed guidance about the parameter specifications (e.g., the number of filters, input and output size) of each module can be found in [131].

- **Fully connected layer (FC)**: As the output from convolution/pooling has been flattened into a single 1024-dimension vector of values representing deep spatial features, the object of a fully connected layer is to multiply each element in the vector by a certain weight value and then sum together. The summed value corresponds to the estimated SWH.

- **Regression layer**: The regression layer is added after the final FC layer in order to compute the half-mean-squared-error loss for the CNN-based SWH estimation model. If H_s is the buoy-measured SWH and H_s' is the network's estimation, the loss function can be expressed as

$$\text{loss} = \frac{1}{2}(H_s - H_s')^2. \tag{5.2}$$

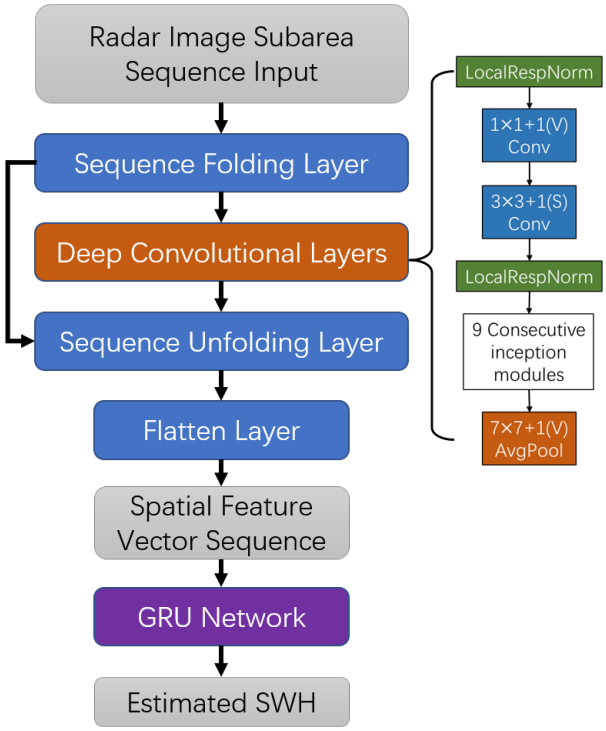

Figure 5.5: Schematic diagram of the CGRU-based SWH estimation model.

5.2.2 CGRU-based SWH Estimation Model

Although the CNN-based model is able to extract multi-scale deep spatial features from the radar image, one of its main shortcomings is that it cannot extract features from temporal variations of the radar image sequence. On the other hand, RNNs can only take vectors as input. Thus, the output of the average pooling layer in the CNN-based model proposed above is used as the input of the GRU network. Compared to another popular type of RNN subclass called the long short term memory (LSTM) unit, GRU is more computationally efficient with simpler structure and similar performance. A schematic diagram of the CGRU-based model is displayed in Fig. 5.5. Specifically, the sequence folding layer converts a batch of image sequences to a batch of images, which allows convolution operations to be performed on each individual subarea image. Then, a sequence unfolding layer and a flattening layer are introduced to restore the sequence structure and reshape the output to spatial

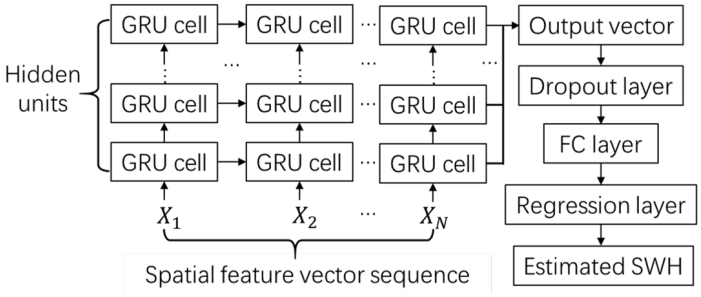

Figure 5.6: Graphic model of the GRU network.

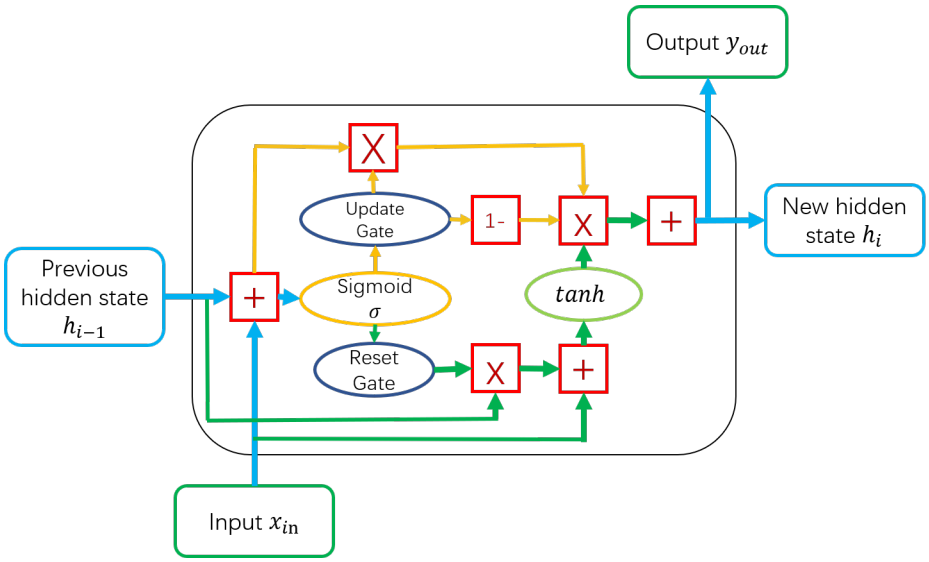

Figure 5.7: The inner structure of a GRU cell.

feature vector sequences. The graphic model of the GRU network is shown in Fig. 5.6. Each spatial feature vector obtained from the deep convolutional layers is input into a series of hidden units (GRU cells) that are connected with each other. Instead of choosing a large value that might cause overfitting, the number of hidden units in each time step (i.e., the number of GRU cells in each column in Fig. 5.6) is set to be 64, which can store sufficient temporal information. The inner structure of each GRU cell is presented in Fig. 5.7. The variables and each type of gate appearing in a cell in the i^{th} column are described below:

- h_{i-1}: the hidden state of preceding cell ($i \geq 1$);

- $\mathbf{x_{in}}$: the input of current cell (a 1024-dimensional vector);
- h_i: the new hidden state generated by current cell;
- $\mathbf{y_{out}}$: the output of current cell, which equals to h_i;
- **Update gate**: the update gate helps the current cell to determine how much of the past information from preceding cells needs to be passed along to future cells and can be expressed as

$$\mathbf{z_i} = \text{sigmoid}(\mathbf{W^{(z)}x_{in}} + \mathbf{U^{(z)}}h_{i-1}), \tag{5.3}$$

  where $\mathbf{W^{(z)}}$ and $\mathbf{U^{(z)}}$ are the weight vectors of the update gate, and the sigmoid activation function transforms the input into a value between 0 and 1.

- **Reset gate**: in contrast to the update gate, reset gate decides how much of the past information to forget. Its expression looks very similar to the update gate's:

$$\mathbf{re_i} = \text{sigmoid}(\mathbf{W^{(r)}x_{in}} + \mathbf{U^{(r)}}h_{i-1}), \tag{5.4}$$

  where $\mathbf{W^{(r)}}$ and $\mathbf{U^{(r)}}$ are weight vectors of the reset gate.

- **Final memory** of current cell: before generating h_i, a new memory content, denoted as $h_i^{'}$, is used to store the relevant information from the past and is calculated as

$$h_i^{'} = \tanh(\mathbf{W^{(h)}x_{in}} + \mathbf{re_i} \odot \mathbf{U^{(h)}}h_{i-1}), \tag{5.5}$$

  where $\mathbf{W^{(h)}}$ and $\mathbf{U^{(h)}}$ are two new weight vectors, $\odot$ represents the Hadamard (element-wise) product between the reset gate and $\mathbf{U^{(h)}}h_{i-1}$. The final memory of current cell, i.e., h_i, can therefore be expressed as

$$h_i = \mathbf{z_i} \odot h_{i-1} + (1 - \mathbf{z_i}) \odot h_i^{'}, \tag{5.6}$$

where $\mathbf{z_i}$ is used to determine what to collect from $h_i^{'}$ and h_{i-1}.

As shown in Fig. 5.6, the output generated by each cell (indicated by the arrows pointing to the top in Fig. 5.6) is used as the input of the next deeper cell in the same time step, while its final memory is passed to the cell of the next time step with the same depth (indicated by the arrows pointed to the right). The final output of the last time step, i.e., a 64-dimensional spatial-temporal feature vector, is first input into a dropout layer, which randomly sets 20% of the input elements to zero to prevent the network from overfitting. Then, similar to the CNN-based regression model, a fully connected layer and the regression layer are added at the end to generate SWH estimation.

5.3 Experiments and Results

5.3.1 Model Training

In the authors' previous works [78, 97, 98], it has been found that rain might blur the wave signatures in the radar image by changing sea surface roughness and scattering the electromagnetic waves in the atmosphere, which might cause inaccurate estimations of SWHs. As shown in Fig. 5.2(b), since precipitation was recorded during data collection periods, it is necessary to identify those rainy image sequences and exclude them from model training for data quality control purposes. In this work, the first image of each sequence is input into a recently proposed CNN-based rainy region segmentation model [134], which is able to conduct pixel-based rainy region identification. If an image contains more than 20% rainy pixels, the sequence it belongs to is identified as rainy and excluded from model training and only used for testing. Also, in order to ensure the proposed models perform similarly under different SWH values, training samples should be obtained under a wide range of simultaneous wave

heights. Therefore, all the image sequences (excluding rainy ones) are first sorted according to simultaneous SWH values obtained from interpolated buoy-measurements. Then, for each 0.5 m value interval between 1 to 5 m, 40% and 30% of the samples are randomly selected for model training and validation, while the remaining 30% are used for testing. In order to validate the robustness of the model, the training and testing experiments for each model are repeated 10 times. In this way, the proposed models can be evaluated using different combinations of training and testing samples. Besides, the widely accepted SNR-based model proposed in [40] (also described in Section 1.3) is also implemented for comparison purposes. The training options for the CNN-based model are listed in the first column of Table 5.1. SGDM denotes the stochastic gradient descent optimizer with momentum, which is an iterative method widely used in the training of CNNs for optimizing an objective function. As for

Table 5.1: Training options of the CNN and GRU networks

Network	CNN	GRU
Solver for network training	SGDM	ADAM
Maximum number of epochs	30	60
Size of mini-batch	16	16
Validation frequency	100 iterations	100 iterations
Initial learning rate	3×10^{-4}	1×10^{-4}
Learn rate drop factor	0.1	0.8
Learn rate drop period	10 epoches	5 epochs

the training of the CGRU-based model, before assembling each component, the spatial feature vectors generated by the convolutional layers of the trained CNN-based model are first used to train the GRU network and the training options are listed in the second column of Table 5.1. ADAM stands for the Adam algorithm proposed in [135], which is an extension to SGDM that has recently seen broader adoption for deep learning applications in time sequence analysis. The trained GRU network is then connected to the flatten layer in Fig. 5.5 and the CGRU-based model can finally be assembled. The training and testing of all models were implemented on MATLAB R2020a installed on a PC running Windows 10 with a 2.5 GHz Intel Core i7-6500U

CPU, 16 GB memory, and a 64-bit operating system.

5.3.2 Result Analysis

Table 5.2: The statistical results of SWH estimation using SNR-based method

Experiment num.	Training			Validation			Testing (rainless)			Testing (rainy)		
	RMSD	CC	Bias	RMSD	CC	Bias	RMSD	CC	Bias	RMSD	CC	Bias
1	0.64	0.57	0.03	0.62	0.61	0.03	0.63	0.58	0.09	0.90	0.46	-0.39
2	0.63	0.59	0.05	0.63	0.60	0.02	0.64	0.56	0.04	0.91	0.46	-0.40
3	0.63	0.60	0.04	0.62	0.60	0.02	0.65	0.56	0.02	0.91	0.46	-0.41
4	0.63	0.59	0.04	0.63	0.59	0.03	0.64	0.57	0.04	0.90	0.46	-0.40
5	0.64	0.58	0.05	0.62	0.60	0.03	0.64	0.56	0.06	0.90	0.46	-0.40
6	0.64	0.58	0.04	0.63	0.61	0.03	0.65	0.57	0.09	0.90	0.46	-0.39
7	0.63	0.60	0.04	0.66	0.55	0.02	0.63	0.59	0.06	0.90	0.46	-0.40
8	0.64	0.58	0.02	0.63	0.61	0.06	0.65	0.57	0.08	0.90	0.46	-0.39
9	0.63	0.59	0.05	0.65	0.58	0.01	0.63	0.59	0.09	0.90	0.46	-0.39
10	0.63	0.60	0.04	0.63	0.59	0.01	0.64	0.56	0.02	0.91	0.46	-0.41
Average	0.63	0.59	0.04	0.63	0.59	0.03	0.64	0.57	0.06	0.90	0.46	-0.40

Table 5.3: The statistical results of SWH estimation using the CNN-based model

Experiment num.	Training			Validation			Testing (rainless)			Testing (rainy)		
	RMSD	CC	Bias	RMSD	CC	Bias	RMSD	CC	Bias	RMSD	CC	Bias
1	0.15	0.98	0.02	0.34	0.89	0.00	0.34	0.89	0.02	0.81	0.77	-0.51
2	0.20	0.97	-0.03	0.32	0.90	-0.02	0.32	0.90	-0.03	0.95	0.75	-0.64
3	0.21	0.96	0.03	0.32	0.91	0.02	0.31	0.91	0.01	0.88	0.76	-0.57
4	0.10	0.99	0.00	0.34	0.89	0.01	0.34	0.89	0.01	0.74	0.78	-0.42
5	0.21	0.96	-0.03	0.31	0.91	-0.03	0.33	0.90	-0.05	1.05	0.72	-0.73
6	0.16	0.98	-0.02	0.31	0.91	-0.05	0.29	0.92	-0.02	0.81	0.77	-0.52
7	0.22	0.96	-0.03	0.33	0.90	-0.03	0.32	0.91	-0.03	0.87	0.76	-0.56
8	0.14	0.98	0.00	0.32	0.91	-0.02	0.31	0.91	0.02	0.77	0.76	-0.46
9	0.15	0.98	0.01	0.35	0.88	0.03	0.35	0.88	0.02	0.84	0.74	-0.48
10	0.20	0.96	0.03	0.32	0.91	0.05	0.33	0.90	0.03	0.92	0.74	-0.58
Average	0.17	0.97	0.00	0.33	0.90	0.00	0.32	0.90	0.00	0.86	0.75	-0.55

The comprehensive statistics of experimental results using all the three methods are presented in Tables 5.2, 5.3 and 5.4, respectively. It can be observed from Table 5.2 that for rainless samples, while the performance of the SNR-based model is very stable between different experiments (i.e., different combinations of training and testing samples), the CCs of both training and rainless testing samples are relatively low (around 0.60). This indicates that SWH is not totally linearly dependent on the square root of SNR, but also dependent on other variables derived from the radar

Table 5.4: The statistical results of SWH estimation using the CGRU-based model

Experiment num.	Training			Validation			Testing (rainless)			Testing (rainy)		
	RMSD	CC	Bias	RMSD	CC	Bias	RMSD	CC	Bias	RMSD	CC	Bias
1	0.21	0.96	0.02	0.26	0.94	0.00	0.29	0.92	0.00	0.53	0.86	-0.24
2	0.23	0.95	-0.03	0.28	0.93	-0.03	0.27	0.93	-0.02	0.62	0.86	-0.40
3	0.22	0.96	-0.01	0.27	0.93	-0.01	0.28	0.93	-0.01	0.46	0.90	-0.20
4	0.23	0.96	0.07	0.27	0.94	0.09	0.29	0.93	0.08	0.50	0.87	-0.19
5	0.22	0.96	-0.07	0.28	0.93	-0.08	0.30	0.92	-0.10	0.60	0.87	-0.37
6	0.25	0.96	0.12	0.31	0.92	0.13	0.28	0.94	0.13	0.48	0.87	-0.11
7	0.21	0.96	0.02	0.28	0.93	0.02	0.27	0.93	0.02	0.52	0.87	-0.26
8	0.24	0.95	-0.05	0.27	0.94	-0.06	0.29	0.92	-0.05	0.62	0.86	-0.41
9	0.23	0.96	0.06	0.28	0.93	0.08	0.27	0.93	0.07	0.46	0.88	-0.13
10	0.27	0.95	-0.15	0.29	0.94	-0.14	0.32	0.93	-0.16	0.58	0.86	-0.34
Average	0.23	0.96	0.00	0.28	0.93	0.00	0.29	0.93	0.00	0.54	0.87	-0.26

image. Since the RMSDs obtained in other studies using the SNR method range from 0.18-0.82 m, the performance of the SNR method is consistent with them. In addition, the errors obtained from rainy image sequences are even much larger, which can be manifested by an average RMSD and average CC of 0.90 m and 0.46, respectively. An average estimation bias of -0.40 m also indicates that SWHs are generally underestimated during rainy conditions, which is in accordance with the findings in [47].

After introducing the CNN-based model, the average RMSD of rainless testing samples drops to 0.32 m. Thus, compared to the SNR-based method, the average RMSD of is reduced by 0.32 m, which indicates the effectiveness of extracting multi-scale deep spatial features and nonlinear regression in improving SWH estimation accuracy. However, the improvement of performance in rainy testing samples is small (the average RMSD is only reduced by 0.04 m) with an even larger average bias (-0.55 m). Hence, it can be inferred that only exploiting the spatial information is unable to improve the estimation accuracy in rainy conditions. In addition, it is found that the RMSDs obtained from validation and testing samples are almost two times higher than that of the training samples, which shows that the CNN-based model is somewhat overfitted (but not severely).

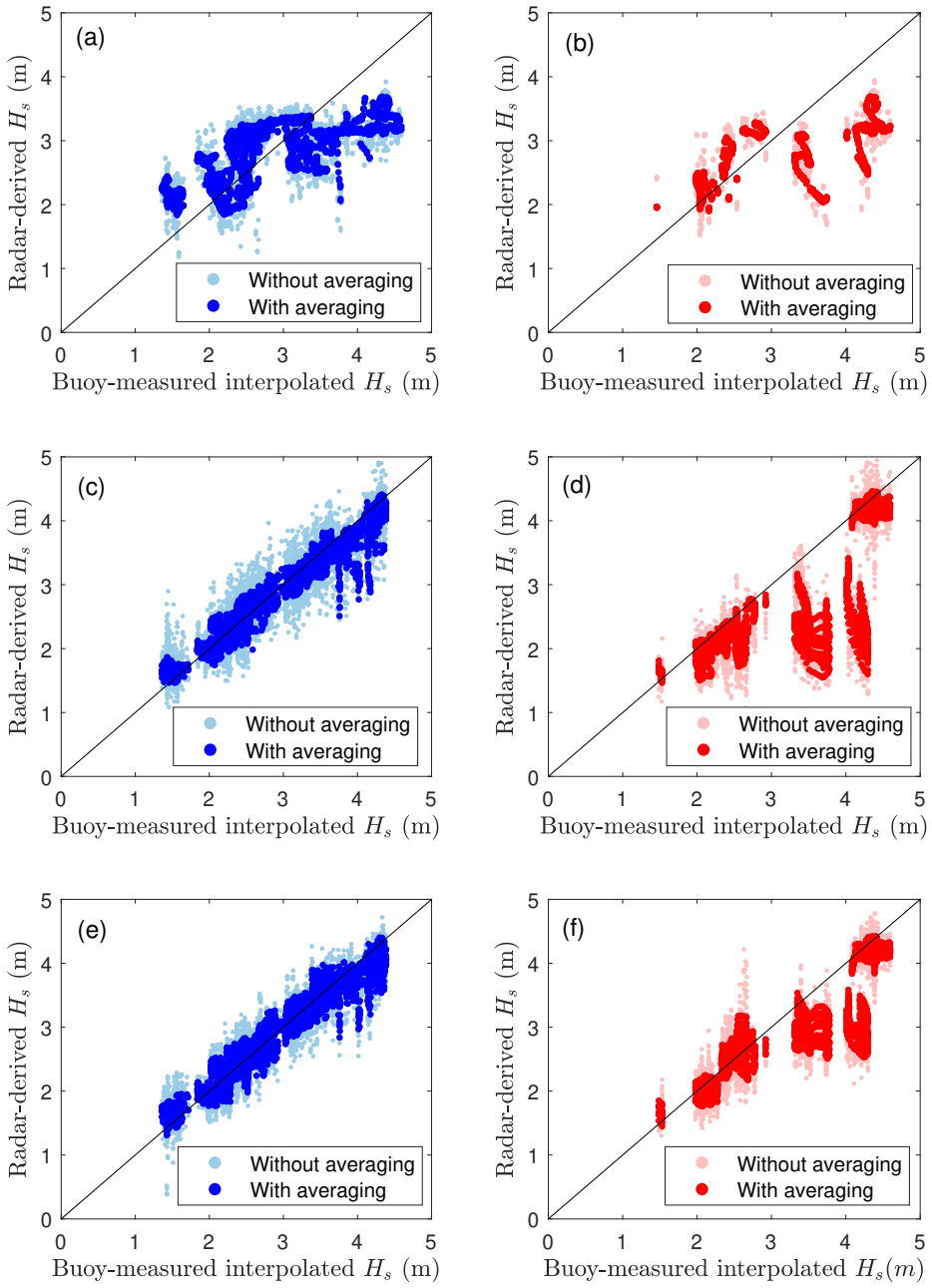

Figure 5.8: Scatter plots of the buoy-measured interpolated H_s and the radar-derived H_s using testing samples. Blue and red dots correspond to rainless and rainy samples, respectively. (a)(b) Testing results obtained from the SNR-based method. (c)(d) Testing results obtained from the proposed CNN-based model. (e)(f) Testing results obtained from the proposed CGRU-based model.

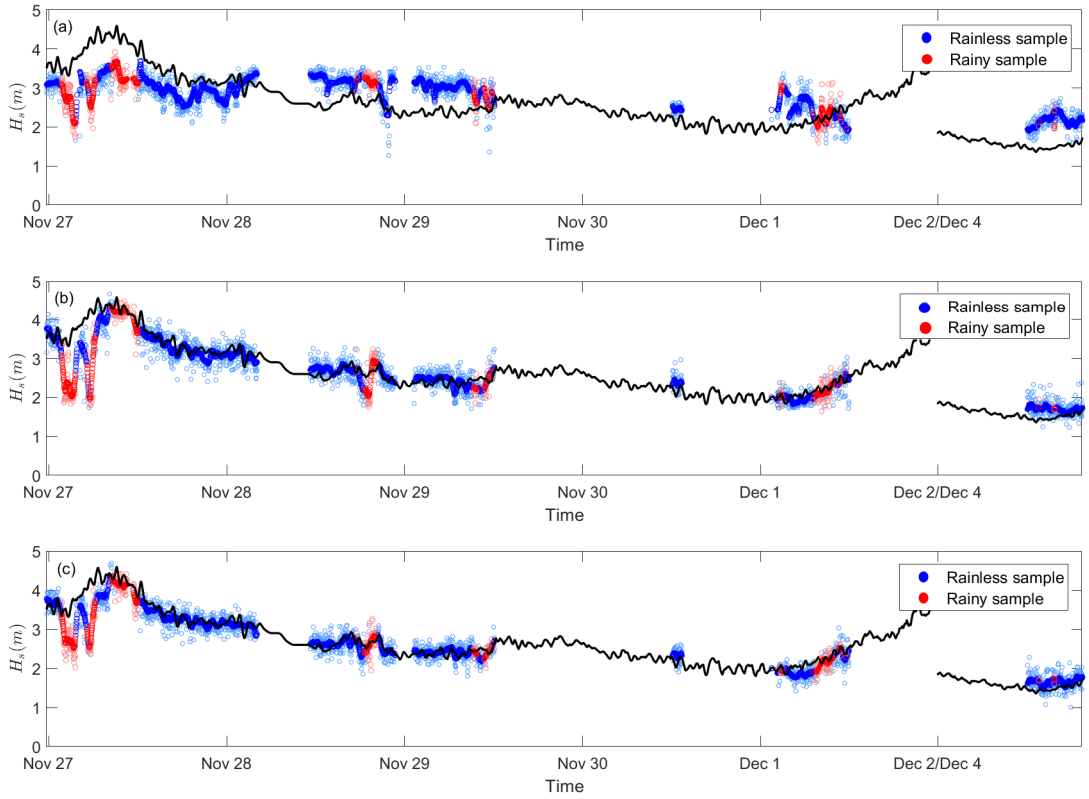

Figure 5.9: Time sequences of SWH results derived by (a) SNR-based, (b) CNN-based, and (c) CGRU-based models.

As shown in Table 5.4, the CGRU-based model obtains the highest accuracy of testing samples by further reducing RMSD and improving the CC. The average RMSD and CC for rainless testing samples are 0.29 m and 0.93, respectively. Besides, compared to the CNN-based model, the differences in RMSDs between training and testing samples are much smaller, which proves that the overfitting issue is alleviated significantly by incorporating the GRU network. As for the rainy samples, the estimation accuracy is improved significantly compared to both the SNR and CNN-based models, with the average RMSD reduced to 0.54 m. In addition, the average bias (-0.26 m) is also closest to zero among the three methods. Thus, it can be concluded that the extraction of temporal features improves both the robustness and accuracy of the model, especially under rainy conditions.

Table 5.5: Comparison of SWH estimation results using testing samples after applying temporal moving average

Method	Testing results with temporal moving average					
	Rainless			Rainy		
	RMSD	CC	Bias	RMSD	CC	Bias
SNR	0.58	0.68	0.07	0.84	0.61	-0.40
CNN only	0.22	0.97	-0.02	0.75	0.80	-0.48
CGRU	0.19	0.98	0.00	0.49	0.91	-0.24

Since the reference buoy-measured SWHs have already been interpolated and moving averaged over 30 minutes, the estimation results of the testing samples shown in Table 5.2, 5.3, 5.4 are also temporally moving averaged for further analysis. Scatter plots that compare radar-derived and buoy-measured SWHs before and after moving average obtained from all 10 experiments are presented in Fig. 5.8. It can be observed that samples are generally moved closer to the black line (perfect fit) after applying temporal moving average. For the SNR-based method, most results obtained with SWHs less than 3 m are overestimated, while underestimation occurs to samples with higher SWHs. In contrast, the under/overestimation issue is alleviated significantly using either CNN or CGRU-based model. As shown in Fig. 5.8(b) and (d), results from rainy samples collected under a sea state with SWHs over 3 m are mostly underestimated using either the SNR or CNN-based models. Although the same issue occurs for the CGRU-based model, the deviations are somewhat decreased, as presented in Fig. 5.8(f). The average RMSDs, CCs, and biases of all 10 experiments calculated after temporal moving average are calculated and displayed in Table 5.5. Compared to the results without applying temporal moving average, the RMSDs are further reduced slightly and the CGRU-based model still produces the highest accuracy for both rainless and rainy testing samples. Finally, the estimation results in time sequences are depicted in Fig. 5.9. For the SNR-based methods, samples obtained under relatively high SWHs (around November 27 to 28) are generally underestimated, while overestimation happens to samples obtained under low SWHs

(December 4). For the CNN-based and CGRU-based models, although the deviations have been reduced significantly in most samples, the results for data collected from 1:30 am to 8:00 am on November 27 are still significantly underestimated (less severe for the CGRU-based model). During this time period, the simultaneous wind speeds were mostly below 4 m/s and precipitation was recorded, as shown in Fig. 5.2. As the wave signatures of radar images obtained under low wind speeds are generally much weaker than those obtained under high wind speeds, rain echoes are more likely to dominate over wave signatures. In contrast, the testing results of rainy samples obtained from around 8:30 am to 12:00 pm on November 27 deviate little from buoy reference. That is because during this period the wind speeds were mostly above 12 m/s, which results in the dominance of wind over rain. A comparison between two rainy images obtained from those two time periods mentioned above is presented in Figs. 3.11(a) and (c). It can be observed that under high wind speeds, rain echoes only increase average radar backscatter intensity without blurring wave signatures significantly, which is in accordance with the findings in [33]. In the end, the efficiency of those methods is evaluated. It took an average of around 20.6 s, 0.6 s, and 3.1 s for the SNR, CNN, and CGRU-based models to generate an SWH estimation from a 32-image sequence, respectively. Hence, compared to the SNR-based method, both the CNN and CGRU-based models speed up the computation significantly. The computational time of the SNR method includes the time for conducting 3D FFT, calculating the current, and implementing the bandpass filtering. The main reason why CNN is faster than CGRU is that CNN only needs to take one single image as input for SWH estimation, while CGRU needs to calculate the features of the whole image sequence. Nevertheless, there is no doubt that both models can be implemented on a normal PC for real-time SWH estimation with higher efficiency than those traditional methods.

5.4 Chapter Summary

In this chapter, the end-to-end deep neural networks are exploited to estimate SWHs from X-band marine radar backscatter image sequences. A CNN-based regression model is first designed based on the convolutional layers of a pretrained GoogLeNet. As the CNN-based model is only able to extract spatial features from the radar image, another model is constructed by connecting a GRU network after the GoogLeNet-based convolutional layers, which enables the model to analyze the temporal behavior of wave signatures in the radar image sequence. Both models are trained and tested using shipborne radar data collected on the East Coast of Canada. Simultaneous buoy measurements are interpolated and used as ground truth for calibration and performance evaluation. Experimental results show that compared to the classic SNR-based method [40], both models improve estimation results significantly using rainless samples, with a reduction of average RMSD by 0.32 m and 0.35 m, respectively. On the other hand, the performance of the CGRU-based model for rainy image sequences is much better than both the SNR and CNN-based models as the average RMSD is reduced from nearly 0.90 m to 0.54 m and the underestimation issue is alleviated. Thus, while extracting multi-scale deep spatial features significantly improves the estimation accuracy, exploiting temporal information boosts the robustness and accuracy of the model under rainy conditions. Besides, the testing results can be further improved after applying temporal moving average. In addition, it has been found that for both of the proposed models, the underestimation issue in rainy conditions mainly occurs under low wind speeds. In contrast, for high wind speeds the estimation results might not be affected significantly since wind-generated wave signatures dominate over rain echoes. Also, compared to the SNR-based method, both of the proposed models reduce the computation time significantly.

Chapter 6

Conclusion

6.1 Summary

The objectives of this thesis are to first detect the presence of rain in X-band marine radar images using several machine learning-based methods, and then propose new methods for estimating wind parameters and significant wave height (SWH) with high accuracy from the images. Since rain alters the pixel intensity distribution of radar images, a SVM-based model with histogram features incorporated is proposed to classify between radar images obtained under rainless and rainy conditions. Then, in order to identify regions that are contaminated by rain echoes in each radar image collected under rainy conditions, two methods are developed based on unsupervised and supervised learning techniques, respectively. For the unsupervised-based method, texture features extracted from radar images are used to train an SOM-based clustering model, which is able to classify each pixel into rain-free, rain-contaminated, and low-backscatter types. As for the supervised-based method, the manually labeled training samples are employed to train an image segmentation CNN which is designed based on SegNet. Then, based on the similarities between the effect of haze on terrestrial images and that of rain on radar images, the rain influence on rain-contaminated

pixels is corrected using DehazeNet, which is a type of CNN developed for image dehazing purpose. Curve-fitting and a SVR-based model are then introduced to estimate wind direction and wind speed from rain-corrected images, respectively. Finally, CNN-based and a convolutional GRU-based networks are designed for significant wave height estimation by extracting deep spatial and temporal features from radar image sequences.

The proposed models in this thesis have been trained and validated using multiple radar datasets collected from both a moving vessel and a shore-based tower under different sea states and rain rates. Simultaneous measurements obtained from in-situ sensors (i.e., rain gauges, anemometers, and wave buoys) are used for calibration and performance evaluation. Experimental results show that all methods obtain satisfactory results with higher accuracy and robustness compared to existing methods. For example, compared to the ZPP threshold method, the proposed SVM-based model improves the rainy image detection accuracy by 16.5% (Decca radar dataset) and 4.2% (Furuno radar dataset). As the first proposed work for pixel-based rain-contaminated region identification, both unsupervised and supervised learning-based methods achieve a pixel identification accuracy of around 90%. For wind direction estimation under rainy conditions, the RMSD is reduced by 19.1° (Decca radar dataset) and 23.9° (Koden radar dataset) after introducing rain correction. On the other hand, the SVR-based wind speed estimation model with texture feature extraction and rain correction incorporated shows higher accuracy and robustness compared to other methods by obtaining a RMSD of 1.10 m/s (Decca radar dataset) and 1.40 m/s (Koden radar dataset) from testing samples. As for SWH estimation, although compared to the classic SNR-based method, the CNN-based model improves estimation accuracy using rain-free images, it does not perform very well under rainy conditions. In contrast, the convolutional GRU-based model achieves relatively high accuracy under both rainless and rainy conditions with a reduction of RMSD by 0.35 m and 0.36

m, respectively. It should be noted that it is likely that one of the reasons CGRU performed better than CNN is that the temporal wave patterns are less influenced by rain than spatial patterns. In other words, information implicated in some temporal modulation can still be deduced from rain-contaminated image sequence (for example, if only one pixel is not rain-contaminated then wave information based on the time series is possible, but the wave spatial pattern would not be observable at all). In addition, compared to the SNR-based method, the computational efficiency has been improved by around 6 times.

Thus, to sum up, the research contributions of this thesis are listed in bulletin points below:

- Significantly improve the estimation accuracy, efficiency, and robustness of wind and wave parameters using X-band marine radar backscatter images, especially under rain conditions.
- Further expand the application of ML techniques on ocean remote sensing using X-band marine radar backscatter images.
- The first work to evaluate the rain detection accuracy from X-band marine radar backscatter images based on different methods.
- The first work to conduct pixel-based identification of rain noise in X-band marine radar backscatter images.
- The first work to introduce deep learning techniques to ocean remote sensing using measured X-band marine radar data.

6.2 Discussion and Future Work

Although the performance of rain detection methods has been evaluated numerically for the first time, the proposed SVM-based model should be further validated using radar datasets collected at different locations under a variety of sea states and rain rates. In particular, the detection accuracy under different rain rates and wind speeds should be investigated. While the study concerning rain-contaminated pixel identification has received satisfactory results, its application should be further explored since so far it has only been incorporated into the schemes for wind and wave parameter estimation. Therefore, in the future it should be introduced into other ocean remote sensing topics using X-band marine radar images as an important step for data quality control. In addition, it should be noted that since the SegNet-based model is trained using manually labeled images based on the authors' expert knowledge, the label accuracy is subject to human experience and errors. It is inevitable that errors might be introduced into output results. Thus, in order to further improve the accuracy and robustness of the model, it is essential to combine the knowledge of different expert groups to eliminate human bias, which requires worldwide collaborations [108]. As for rain mitigation, while this is the first work to apply image dehazing techniques to correct the rain influence on marine radar images, its effectiveness has only been validated in wind parameter estimation. Future work should expand its application to other tasks concerning ocean remote sensing using X-band marine radar backscatter images.

In this thesis, wind estimation accuracy has been further improved compared to existing methods, especially under rain conditions. Wind measurements also facilitate the selection of the training samples under different sea states for the models developed in this research. Nevertheless, it should be acknowledged that non-coherent marine

radars are not radiometrically calibrated instruments. Hence, marine radar wind sensors require independent wind measurements for calibration purposes. Complicating matters further, we are still lacking a complete physical understanding of the grazing incidence X-band radar backscatter intensity's dependency on antenna polarization, antenna look direction, and range [136]. These challenges are likely the main reasons why marine radar wind sensors have not yet been embraced by the broader oceanographic / atmospheric science community as an alternative to standard anemometers. However, in contrast to standard anemometers, marine radar backscatter intensity measurements hold valuable and unique information on the wind's spatial variability and, more specifically, wind gusts. The retrieval of wind gusts from radar measurements is therefore a highly promising (albeit difficult) field for future research. On the other hand, although the SNR-based wave estimation algorithm has already been incorporated into several marine radar-based wave monitoring systems that are commercially available, the deep neural network-based SWH estimation model proposed in the thesis has the great potential to be one of the prototypes for next-generation wave monitoring systems because of its superb performance. Although the proposed algorithms can be applied to other datasets once they are trained, the difference of some features (e.g., radar parameters, water depth) between training and testing datasets may significantly affect the performance of the algorithms. Therefore, the suitability of applying an algorithm to a certain dataset should be evaluated empirically beforehand. Since the accuracy, robustness, and generalization ability of deep neural networks depend on the amount of data used for training, future versions of deep learning-based SWH estimation models should include as many available radar datasets as possible.

Bibliography

[1] H. Dankert and J. Horstmann, "A marine radar wind sensor," *J. Atmos. Ocean. Technol.*, vol. 24, no. 9, pp. 1629–1631,1633–1642, Sep. 2007.

[2] G. Engen and H. Johnsen, "SAR-ocean wave inversion using image cross spectra," *IEEE Trans. Geosci. Remote Sens.*, vol. 33, no. 4, pp. 1047–1056, 1995.

[3] S. Lehner, J. Schulz-Stellenfleth, B. Schattler, H. Breit, and J. Horstmann, "Wind and wave measurements using complex ERS-2 SAR wave mode data," *IEEE Trans. Geosci. Remote Sens.*, vol. 38, no. 5, pp. 2246–2257, 2000.

[4] H. Lin, Q. Xu, and Q. Zheng, "An overview on SAR measurements of sea surface wind," *Progr. Natural Sci.*, vol. 18, no. 8, pp. 913–919, 2008.

[5] B. Lipa, D. Barrick, J. Isaacson, and P. Lilleboe, "CODAR wave measurements from a north sea semisubmersible," *IEEE J. Ocean. Eng.*, vol. 15, no. 2, pp. 119–125, 1990.

[6] W. Huang, S. Wu, E. Gill, B. Wen, and J. Hou, "HF radar wave and wind measurement over the eastern china sea," *IEEE Trans. Geosci. Remote Sens.*, vol. 40, no. 9, pp. 1950–1955, 2002.

[7] L. R. Wyatt, J. J. Green, A. Middleditch, M. D. Moorhead, J. Howarth, M. Holt, and S. Keogh, "Operational wave, current, and wind measurements with the pisces HF radar," *IEEE J. Ocean. Eng.*, vol. 31, no. 4, pp. 819–834, 2006.

[8] I. R. Young, W. Rosenthal, and F. Ziemer, "A three-dimensional analysis of marine radar images for the determination of ocean wave directionality and surface currents," *J. Geophys. Res.-Oceans*, vol. 90, no. C1, pp. 1049–1059, 1985.

[9] J. Nieto Borge and K. Hessner, "Inversion of marine radar images for surface wave analysis," *J. Atmos. Ocean. Technol.*, vol. 21, no. 8, pp. 1291–1300, Aug. 2004.

[10] W. Huang and E. W. Gill, "Ocean remote sensing using X-band shipborne nautical radar—applications in eastern canada," in *Coast. Ocean Observing Syst.* Elsevier, 2015, pp. 248–264.

[11] S. Chen and W. Huang, "Maneuvering target tracking from nautical radar images using particle-kalman filters," *J. Electromagn. Waves Appl.*, vol. 27, no. 18, pp. 2366–2378, 2013.

[12] P. S. Bell, "Shallow water bathymetry derived from an analysis of X-band marine radar images of waves," *Coast. Eng.*, vol. 37, no. 3-4, pp. 513–527, 1999.

[13] P. Chernyshov, T. Vrecica, M. Streßer, R. Carrasco, and Y. Toledo, "Rapid wavelet-based bathymetry inversion method for nearshore X-band radars," *Remote Sens. Environ.*, vol. 240, p. 111688, 2020.

[14] M. C. Haller, D. Honegger, and P. A. Catalan, "Rip current observations via marine radar," *J. Waterw. Port Coast. Ocean Eng.*, vol. 140, no. 2, pp. 115–124, 2014.

[15] F. Raffa, G. Ludeno, B. Patti, F. Soldovieri, S. Mazzola, and F. Serafino, "X-band wave radar for coastal upwelling detection off the southern coast of Sicily," *J. Atmos. Ocean. Technol.*, vol. 34, no. 1, pp. 21–31, 2017.

[16] B. Lund, H. C. Graber, J. Xue, and R. Romeiser, "Analysis of internal wave signatures in marine radar data," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 9, pp. 4840–4852, 2013.

[17] Z. Chen, J. Pan, Y. He, and A. T. Devlin, "Estimate of tidal constituents in nearshore waters using X-band marine radar image sequences," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 11, pp. 6700–6711, 2016.

[18] B. Lund, H. C. Graber, P. Persson, M. Smith, M. Doble, J. Thomson, and P. Wadhams, "Arctic sea ice drift measured by shipboard marine radar," *J. Geophys. Res.-Oceans*, vol. 123, no. 6, pp. 4298–4321, 2018.

[19] R. Vicen-Bueno, J. Horstmann, E. Terril, T. de Paolo, and J. Dannenberg, "Real-time ocean wind vector retrieval from marine radar image sequences acquired at grazing angle," *J. Atmos. Ocean. Technol.*, vol. 30, no. 1, pp. 127–139, 2013.

[20] W. Huang, E. Gill, and J. An, "Iterative least-squares-based wave measurement using X-band nautical radar," *IET Radar Sonar Nav.*, vol. 8, no. 8, pp. 853–863, 2014.

[21] W. Huang and E. Gill, "Surface current measurement under low sea state using dual polarized X-band nautical radar," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 5, no. 6, pp. 1868–1873, 2012.

[22] C. Shen, W. Huang, E. Gill, R. Carrasco, and J. Horstmann, "An algorithm for surface current retrieval from X-band marine radar images," *Remote Sens.*, vol. 7, no. 6, pp. 7753–7767, 2015.

[23] W. Huang, R. Carrasco, C. Shen, E. W. Gill, and J. Horstmann, "Surface current measurements using X-band marine radar with vertical polarization," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 5, pp. 2988–2997, 2016.

[24] B. Lund, H. C. Graber, and R. Romeiser, "Wind retrieval from shipborne nautical X-band radar data," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 10, pp. 3800–3811, Oct. 2012.

[25] W. Huang, X. Liu, and E. Gill, "Ocean wind and wave measurements using X-band marine radar: A comprehensive review," *Remote Sens.*, vol. 9, no. 12, Jan. 2017.

[26] P. Lee, J. Barter, E. Caponi, M. Caponi, C. Hindman, B. Lake, and H. Rungaldier, "Wind-speed dependence of small-grazing-angle microwave backscatter from sea surfaces," *IEEE Trans. Antennas Propag.*, vol. 44, no. 3, pp. 333–340, 1996.

[27] D. B. Trizna and D. J. Carlson, "Studies of dual polarized low grazing angle radar sea scatter in nearshore regions," *IEEE Trans. Geosci. Remote Sens.*, vol. 34, no. 3, pp. 747–757, 1996.

[28] Z. Chen, Y. He, B. Zhang, Z. Qiu, and B. Yin, "A new algorithm to retrieve wave parameters from marine X-band radar image sequences," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 7, pp. 4083–4091, 2014.

[29] Y. Du, P. W. Vachon, and J. Wolfe, "Wind direction estimation from sar images of the ocean using wavelet analysis," *Can. J. Remote Sens.*, vol. 28, no. 3, pp. 498–509, 2002.

[30] H. Dankert, J. Horstmann, and W. Rosenthal, "Ocean wind fields retrieved from radar-image sequences," *J. Geophys. Res.-Oceans*, vol. 108, no. C11, 2003.

[31] Z. Chen, Y. He, B. Zhang, and Z. Qiu, "Determination of nearshore sea surface wind vector from marine X-band radar images," *Ocean Engi.*, vol. 96, pp. 79–85, 2015.

[32] P. Izquierdo and C. G. Soares, "Analysis of sea waves and wind from X-band radar," *Ocean Eng.*, vol. 32, no. 11-12, pp. 1404–1419, 2005.

[33] Y. Wang and W. Huang, "An algorithm for wind direction retrieval from X-band marine radar images," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 2, pp. 252–256, 2016.

[34] W. Huang and Y. Wang, "A spectra-analysis-based algorithm for wind speed estimation from X-band nautical radar images," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 5, pp. 701–705, 2016.

[35] X. Liu, W. Huang, and E. W. Gill, "Wind direction estimation from rain-contaminated marine radar data using the ensemble empirical mode decomposition method," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 3, pp. 1833–1841, 2017.

[36] W. Huang, X. Liu, and E. W. Gill, "An empirical mode decomposition method for sea surface wind measurements from X-band nautical radar data," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 11, pp. 6218–6227, 2017.

[37] L. H. Holthuijsen, *Waves in oceanic and coastal waters.* Cambridge University Press, 2010.

[38] J. N. Borge and C. G. Soares, "Analysis of directional wave fields using X-band navigation radar," *Coast. Eng.*, vol. 40, no. 4, pp. 375–391, 2000.

[39] W. Alpers and K. Hasselmann, "Spectral signal to clutter and thermal noise properties of ocean wave imaging synthetic aperture radars," *Intl. J. Remote Sens.*, vol. 3, no. 4, pp. 423–446, 1982.

[40] J. Nieto-Borge, K. Hessner, P. Jarabo-Amores, and D. De La Mata-Moya, "Signal-to-noise ratio analysis to estimate ocean wave heights from X-band marine radar image time series," *IET Radar Sonar Nav.*, vol. 2, no. 1, pp. 35–41, 2008.

[41] L. Cornejo-Bueno, J. N. Borge, E. Alexandre, K. Hessner, and S. Salcedo-Sanz, "Accurate estimation of significant wave height with support vector regression algorithms and marine radar images," *Coast. Eng.*, vol. 114, pp. 233–243, 2016.

[42] Y. Wei, Z. Lu, G. Pian, and H. Liu, "Wave height estimation from shadowing based on the acquired X-band marine radar images in coastal area," *Remote Sens.*, vol. 9, no. 8, p. 859, 2017.

[43] X. Liu, W. Huang, and E. W. Gill, "Estimation of significant wave height from X-band marine radar images based on ensemble empirical mode decomposition," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 10, pp. 1740–1744, 2017.

[44] J. An, W. Huang, and E. W. Gill, "A self-adaptive wavelet-based algorithm for wave measurement using nautical radar," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 1, pp. 567–577, 2015.

[45] K. Ma, X. Wu, X. Yue, L. Wang, and J. Liu, "Array beamforming algorithm for estimating waves and currents from marine X-band radar image sequences," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 3, pp. 1262–1272, 2017.

[46] R. Gangeskar, "An algorithm for estimation of wave height from shadowing in X-band radar sea surface images," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 6, pp. 3373–3381, 2013.

[47] X. Liu, W. Huang, and E. W. Gill, "Wave height estimation from shipborne X-band nautical radar images," *J. Sens.*, vol. 2016, 2016.

[48] ——, "Comparison of wave height measurement algorithms for ship-borne X-band nautical radar," *Can. J. Remote Sens.*, vol. 42, no. 4, pp. 343–353, 2016.

[49] Z. Chen, Y. He, and B. Zhang, "An automatic algorithm to retrieve wave height from X-band marine radar image sequence," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 9, pp. 5084–5092, 2017.

[50] G. Ludeno and F. Serafino, "Estimation of the significant wave height from marine radar images without external reference," *J. Marine Sci. Eng.*, vol. 7, no. 12, p. 432, 2019.

[51] W. Navarro, J. C. Velez, A. Orfila, and S. Lonin, "A shadowing mitigation approach for sea state parameters estimation using X-band remotely sensing radar data in coastal areas," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 9, pp. 6292–6310, 2019.

[52] R. Vicen-Bueno, C. Lido-Muela, and J. C. Nieto-Borge, "Estimate of significant wave height from non-coherent marine radar images by multilayer perceptrons," *EURASIP J. Adv. Signal Process.*, vol. 2012, no. 1, p. 84, 2012.

[53] S. Salcedo-Sanz, J. N. Borge, L. Carro-Calvo, L. Cuadra, K. Hessner, and E. Alexandre, "Significant wave height estimation using SVR algorithms and shadowing information from simulated and real measured X-band radar images of the sea surface," *Ocean Eng.*, vol. 101, pp. 244–253, 2015.

[54] W. Duan, K. Yang, L. Huang, and X. Ma, "Numerical investigations on wave remote sensing from synthetic X-band radar sea clutter images by using deep convolutional neural networks," *Remote Sens.*, vol. 12, no. 7, p. 1117, 2020.

[55] J. Tournadre and Y. Quilfen, "Impact of rain cell on scatterometer data: 1. theory and modeling," *J. Geophys. Res.-Oceans*, vol. 108, no. C7, 2003.

[56] R. F. Contreras and W. J. Plant, "Surface effect of rain on microwave backscatter from the ocean: Measurements and modeling," *J. Geophys. Res.-Oceans*, vol. 111, no. C8, 2006.

[57] W. Huang, Y. Liu, and E. W. Gill, "Texture-analysis-incorporated wind parameters extraction from rain-contaminated X-band nautical radar images," *Remote Sens.*, vol. 9, no. 2, p. 166, 2017.

[58] J. Shen, Y. Li, Y. Dai, and S. Wang, "Identification and suppression of rain interference on X-band radar images," *Opt. Precis. Eng.*, vol. 20, no. 8, pp. 1846–1853, 2012.

[59] Z. Chen, Y. He, B. Zhang, and Y. Ma, "A method to correct the influence of rain on X-band marine radar image," *IEEE Access*, vol. 5, pp. 25 576–25 583, 2017.

[60] Y. Liu, W. Huang, E. W. Gill, D. K. Peters, and R. Vicen-Bueno, "Comparison of algorithms for wind parameters extraction from shipborne X-band marine radar images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 2, pp. 896–906, 2015.

[61] V. Vapnik, "Principles of risk minimization for learning theory," in *Adv. Neural Inf. Process Syst.*, 1992, pp. 831–838.

[62] ——, *The nature of statistical learning theory.* Springer science & business media, 2013.

[63] F. Melgani and L. Bruzzone, "Classification of hyperspectral remote sensing images with support vector machines," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 8, pp. 1778–1790, 2004.

[64] C. Huang, L. Davis, and J. Townshend, "An assessment of support vector machines for land cover classification," *Int. J. Remote Sens.*, vol. 23, no. 4, pp. 725–749, 2002.

[65] T. Kavzoglu and I. Colkesen, "A kernel functions analysis for support vector machines for land cover classification," *Int. J. Appl. Earth Obs.*, vol. 11, no. 5, pp. 352–359, 2009.

[66] H. Zhan, P. Shi, and C. Chen, "Retrieval of oceanic chlorophyll concentration using support vector machines," *IEEE Trans. Geosci. Remote Sens.*, vol. 41, no. 12, pp. 2947–2951, 2003.

[67] M. A. Tebbi and B. Haddad, "Artificial intelligence systems for rainy areas detection and convective cells' delineation for the south shore of Mediterranean Sea during day and nighttime using MSG satellite images," *Atmospheric Res.*, vol. 178, pp. 380–392, 2016.

[68] S. Upadhyaya and R. Ramsankaran, "Support vector machine (SVM) based rain area detection from Kalpana-1 satellite data," *ISPRS Ann. Photogramm., Remote Sens. Spat. Inf. Sci.*, vol. 2, no. 8, pp. 21–27, 2014.

[69] D. D. Webster and T. P. Breckon, "Improved raindrop detection using combined shape and saliency descriptors with scene context isolation," in *IEEE Intl. Conf. Image Proc. (ICIP)*, Quebec City, QC, Canada, 2015.

[70] M. I. Skolnik, "Radar handbook," 1970.

[71] R. Moore and A. K. Fung, "Radar determination of winds at sea," *Proc. IEEE*, vol. 67, no. 11, pp. 1504–1521, 1979.

[72] L. C. Schroeder, D. H. Boggs, G. Dome, I. M. Halberstam, W. L. Jones, W. J. Pierson, and F. J. Wentz, "The relationship between wind vector and normalized radar cross section used to derive SEASAT-a satellite scatterometer winds," *J. Geophys. Res.-Oceans*, vol. 87, no. C5, pp. 3318–3336, 1982.

[73] C. Craeye, P. Sobieski, and L. Bliven, "Scattering by artificial wind and rain roughened water surfaces at oblique incidences," *Int. J. Remote Sens.*, vol. 18, no. 10, pp. 2241–2246, 1997.

[74] D. C. Stredulinsky and E. M. Thornhill, "Ship motion and wave radar data fusion for shipboard wave measurement," *J. Ship Res.*, vol. 55, no. 2, pp. 73–85, 2011.

[75] I. WaMoS, "Wave and surface current monitoring system operating manual version 4.0," *OceanWaves GmbH: Luneburg, Germany*, 2012.

[76] R.-E. Fan, P.-H. Chen, and C.-J. Lin, "Working set selection using second order information for training support vector machines," *J. Mach. Learn. Res.*, vol. 6, pp. 1889–1918, 2005.

[77] J. Platt, "Sequential minimal optimization: A fast algorithm for training support vector machines," 1998.

[78] X. Chen, W. Huang, C. Zhao, and Y. Tian, "Rain detection from X-band marine radar images: A support vector machine-based approach," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 3, pp. 2115–2123, 2020.

[79] X. Chen and W. Huang, "Automatic identification of rain-contaminated regions in X-band marine radar images," in *Proc. MTS/IEEE Oceans*, Seattle, USA, 2019, pp. 1–4.

[80] E. J. Thompson, W. E. Asher, A. T. Jessup, and K. Drushka, "High-resolution rain maps from an X-band marine radar," *Oceanogr.*, vol. 32, no. 2, p. 58, 2019.

[81] A. Basile and F. Butini, "Rainfall estimation by means of a meteorological radar: an innovative approach using neural networks," *EARSeL Adv. Remote Sens.*, vol. 2, no. 2, pp. 32–39, 1993.

[82] T. Randen and J. H. Husøy, "Filtering for texture classification: A comparative study," *IEEE Trans. Pattern Anal. Mach. Intell.*, no. 4, pp. 291–310, 1999.

[83] A. K. Jain and F. Farrokhnia, "Unsupervised texture segmentation using Gabor filters," *Pattern Recognit.*, vol. 24, no. 12, pp. 1167–1186, 1991.

[84] I. Ng, T. Tan, and J. Kittler, "On local linear transform and gabor filter representation of texture," in *Proc., Int. Conf. Pattern Recognit.*, 1992, pp. 627–631.

[85] T. Chang and C.-C. J. Kuo, "Texture analysis and classification with tree-structured wavelet transform," *IEEE Trans. Image Process.*, vol. 2, no. 4, pp. 429–441, 1993.

[86] Y. Liu, R. H. Weisberg, and L. K. Shay, "Current patterns on the west Florida shelf from joint self-organizing map analyses of HF radar and ADCP data," *J. Atmos. Ocean. Technol.*, vol. 24, no. 4, pp. 702–712, 2007.

[87] Y. Liu and R. H. Weisberg, "A review of self-organizing map applications in meteorology and oceanography," in *Self Organizing Maps-Applications and Novel Algorithm Design*, J. I. Mwasiagi, Ed. Rijeka, Croatia: IntechOpen, 2011, ch. 14, pp. 253–272.

[88] I. Daubechies, *Ten lectures on wavelets*. SIAM, 1992.

[89] A. Graps, "An introduction to wavelets," *IEEE Comput. Sci. Eng.*, vol. 2, no. 2, pp. 50–61, 1995.

[90] K. R. Rao and P. Yip, *Discrete cosine transform: algorithms, advantages, applications*. Academic press, 2014.

[91] X. Chen, W. Huang, and G. Yao, "Wind speed estimation from X-band marine radar images using support vector regression method," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 9, pp. 1312–1316, 2018.

[92] A. M. Kalteh, P. Hjorth, and R. Berndtsson, "Review of the self-organizing map (SOM) approach in water resources: Analysis, modelling and application," *Environ. Model. Softw.*, vol. 23, no. 7, pp. 835–845, 2008.

[93] J. Vesanto, E. Alhoniemi *et al.*, "Clustering of the self-organizing map," *IEEE Trans. Neural Networks*, vol. 11, no. 3, pp. 586–600, 2000.

[94] Y. Liu, R. H. Weisberg, and C. N. Mooers, "Performance evaluation of the self-organizing map for feature extraction," *J. Geophys. Res.-Oceans*, vol. 111, no. C5, 2006.

[95] J. H. Ward Jr, "Hierarchical grouping to optimize an objective function," *J. Am. Stat. Assoc.*, vol. 58, no. 301, pp. 236–244, 1963.

[96] Y. J. Zhang, "A survey on evaluation methods for image segmentation," *Pattern Recognit.*, vol. 29, no. 8, pp. 1335–1346, 1996.

[97] X. Chen and W. Huang, "Identification of rain and low-backscatter regions in X-band marine radar images: An unsupervised approach," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 6, pp. 4225–4236, 2020.

[98] X. Chen and W. Huang, "Texture features and unsupervised learning-incorporated rain-contaminated region identification from X-band marine radar images," *Mar. Technol. Soc. J.*, vol. 54, no. 4, pp. 59–67, 2020.

[99] R. Li, W. Liu, L. Yang, S. Sun, W. Hu, F. Zhang, and W. Li, "DeepUNet: a deep fully convolutional network for pixel-level sea-land segmentation," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 11, pp. 3954–3962, 2018.

[100] J. Lu, Y. Wang, Y. Zhu, X. Ji, T. Xing, W. Li, and A. Y. Zomaya, "P_SegNet and NP_SegNet: New neural network architectures for cloud recognition of remote sensing images," *IEEE Access*, vol. 7, pp. 87 323–87 333, 2019.

[101] G. Chen, X. Zhang, Q. Wang, F. Dai, Y. Gong, and K. Zhu, "Symmetrical dense-shortcut deep fully convolutional networks for semantic segmentation of very-high-resolution remote sensing images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 5, pp. 1633–1644, 2018.

[102] H. Luo, C. Chen, L. Fang, X. Zhu, and L. Lu, "High-resolution aerial images semantic segmentation using deep fully convolutional network with channel attention mechanism," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 9, pp. 3492–3507, 2019.

[103] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vision Pattern Recog.*, Boston, USA, 2015, pp. 3431–3440.

[104] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, 2017.

[105] A.-A. Al-Habashneh, C. Moloney, E. W. Gill, and W. Huang, "The effect of radar ocean surface sampling on wave spectrum estimation using X-band marine radar," *IEEE Access*, vol. 6, pp. 17 570–17 585, 2018.

[106] X. X. Zhu, D. Tuia, L. Mou, G.-S. Xia, L. Zhang, F. Xu, and F. Fraundorfer, "Deep learning in remote sensing: A comprehensive review and list of resources," *IEEE Geosci. Remote Sens. Mag.*, vol. 5, no. 4, pp. 8–36, 2017.

[107] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[108] X. Li, B. Liu, G. Zheng, Y. Ren, S. Zhang, Y. Liu, L. Gao, Y. Liu, B. Zhang, and F. Wang, "Deep-learning-based information mining from ocean remote-sensing imagery," *Natl. Sci. Rev.*, vol. 7, no. 10, pp. 1584–1605, 2020.

[109] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. Int. Conf. Artif. Intell. Stat.*, Sardinia, Italy, 2010, pp. 249–256.

[110] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proc. 19th Int. Conf. Comput. Statist.*, Paris, France, 2010, pp. 177–186.

[111] D. Marmanis, J. D. Wegner, S. Galliani, K. Schindler, M. Datcu, and U. Stilla, "Semantic segmentation of aerial images with an ensemble of CNNs," in *Proc. ISPRS Ann. Photogramm., Remote Sens. Spatial Inf. Sci.*, 2016, pp. 473–480.

[112] M. C. Haller, D. A. Honegger, R. Pittman, A. O'Dea, and A. Simpson, "Real-time marine radar observations of nearshore waves and flow structures from shore-based towers," in *Proc. IEEE/OES 12th Current, Waves and Turbulence Measurement (CWTM)*, San Diego, USA, 2019, pp. 1–7.

[113] X. Chen, W. Huang, and M. Haller, "A novel scheme for extracting sea surface wind information from rain-contaminated X-band marine radar images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, 2021.

[114] B. Cai, X. Xu, K. Jia, C. Qing, and D. Tao, "Dehazenet: An end-to-end system for single image haze removal," *IEEE Trans. Image Proc.*, vol. 25, no. 11, pp. 5187–5198, 2016.

[115] X. Chen and W. Huang, "Gaussian process regression for estimating wind speed from X-band marine radar images," in *Proc. MTS/IEEE Oceans*, Charleston, USA, 2018, pp. 1–4.

[116] R. T. Tan, "Visibility in bad weather from a single image," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun, 2008, pp. 1–8.

[117] K. He, J. Sun, and X. Tang, "Single image haze removal using dark channel prior," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 12, pp. 2341–2353, 2010.

[118] Q. Zhu, J. Mai, and L. Shao, "A fast single image haze removal algorithm using color attenuation prior," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 3522–3533, 2015.

[119] X. Liu, W. Huang, and E. W. Gill, "Analysis of rain effects on wave height estimation from X-band nautical radar images," in *Proc. MTS/IEEE Oceans*, Washington, USA, 2015, pp. 1–4.

[120] K. Tang, J. Yang, and J. Wang, "Investigating haze-relevant features in a learning framework for image dehazing," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun, 2014, pp. 2995–3000.

[121] K. He, J. Sun, and X. Tang, "Guided image filtering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 6, pp. 1397–1409, 2012.

[122] S. G. Narasimhan and S. K. Nayar, "Vision and the atmosphere," *Int. J. Comput. Vision*, vol. 48, no. 3, pp. 233–254, 2002.

[123] R. Fattal, "Single image dehazing," *ACM Trans. Gr. (TOG)*, vol. 27, no. 3, pp. 1–9, 2008.

[124] O. Phillips, "On the dynamics of unsteady gravity waves of finite amplitude part 1. the elementary interactions," *J. Fluid Mech.*, vol. 9, no. 2, pp. 193–217, 1960.

[125] A. J. Smola and B. Schölkopf, "A tutorial on support vector regression," *Statist. Comput.*, vol. 14, no. 3, pp. 199–222, 2004.

[126] R. Gangeskar, "Verifying high-accuracy ocean surface current measurements by X-band radar for fixed and moving installations," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 8, pp. 4845–4855, 2018.

[127] H.-Y. Cheng and H. Chien, "Implementation of S-band marine radar for surface wave measurement under precipitation," *Remote Sens. Environ.*, vol. 188, pp. 85–94, 2017.

[128] X. Chen and W. Huang, "Spatial-temporal convolutional gated recurrent unit network for significant wave height estimation from shipborne marine radar data," *IEEE Trans. Geosci. Remote Sens.*, 2021.

[129] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

[130] M. T. Silva, E. W. Gill, and W. Huang, "An improved estimation and gap-filling technique for sea surface wind speeds using NARX neural networks," *J. Atmos. Ocean. Technol.*, vol. 35, no. 7, pp. 1521–1532, 2018.

[131] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vision Pattern Recog.*, Boston, USA, 2015, pp. 1–9.

[132] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Adv. Neural Inform. Process. Syst.*, vol. 25, pp. 1097–1105, 2012.

[133] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[134] X. Chen, W. Huang, M. Haller, and R. Pittman, "Rain-contaminated region segmentation of X-band marine radar images with an ensemble of SegNets," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 141–154, 2021.

[135] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[136] B. Lund, H. C. Graber, H. Tamura, C. Collins III, and S. Varlamov, "A new technique for the retrieval of near-surface vertical current shear from marine x-band radar images," *J. Geophys. Res.-Oceans*, vol. 120, no. 12, pp. 8466–8486, 2015.