Multi-Feature Collaborative Fusion Network with Deep Supervision for SAR Ship Classification

Hao Zheng, *Student Member, IEEE,* Zhigang Hu, Liu Yang, Aikun Xu,
Meiguang Zheng, and Keqin Li, *Fellow, IEEE*

***Abstract*—Multi-feature SAR ship classification aims to build models that can process, correlate, and fuse information from both handcrafted and deep features. Although handcrafted features provide rich expert knowledge, current fusion methods inadequately explore the relatively significant role of handcrafted features in conjunction with deep features, the imbalances in feature contributions, and the cooperative ways in which features learn. In this paper, we propose a novel multi-feature collaborative fusion network with deep supervision (MFCFNet) to effectively fuse handcrafted features and deep features for SAR ship classification tasks. Specifically, our framework mainly includes two types of feature extraction branches, a knowledge supervision and collaboration module, and a feature fusion and contribution assignment module. The former module improves the quality of the feature maps learned by each branch through auxiliary feature supervision and introduces a synergy loss to facilitate the interaction of information between deep features and handcrafted features. The latter module utilizes an attention mechanism to adaptively balance the importance among various features and assign the corresponding feature contributions to the total loss function based on the generated feature weights. We conducted extensive experimental and ablation studies on two public datasets, OpenSARShip-1.0 and FUSAR-Ship, and the results show that MFCFNet is effective and outperforms single deep feature and multi-feature models based on previous internal FC layer and terminal FC layer fusion. Furthermore, our proposed MFCFNet exhibits better performance than the current state-of-the-art methods.**

***Index Terms*—Multi-Feature fusion, handcrafted feature, deep supervision, synthetic aperture radar (SAR), SAR ship classification.**

I. Introduction

SYNTHETIC Aperture Radar (SAR) is a high-resolution radar system that can operate from either spaceborne or airborne platforms, providing a variety of features such as all-day, all-weather, and cloud-penetrating capabilities. Unlike optoelectronic sensors, SAR image mainly reflects the backscattered information of the target, and the image signal-to-noise ratio is low. Moreover, the signal-to-noise ratio decreases with increasing radar distance, and its amplitude value fluctuates randomly with the change of target observation angle, which makes SAR target identification far more complex than optical images. Ships are the main means of transportation for maritime trade, and SAR is the main means of marine detection. It is important to develop SAR ship classification for marine fisheries management, maritime traffic management, combating illegal activities at sea, maritime search and rescue, and so on. Therefore, SAR image interpretation and ship target information extraction are the critical issues for SAR ship classification.

Features are the keys to SAR ship classification, since the goodness of features largely determines the accuracy of classification. These features can be divided into two categories: traditional handcrafted features and deep features based on modern convolutional neural networks (CNN) according to the differences in feature extraction. The handcrafted feature often needs to describe images from different perspectives using mature and explicable mathematical theories, such as grayscale, texture, edge, or shape [1–3]. Specifically, they are only applicable to a specific environment, so generalization in unknown environments is not sufficient. However, multi-sensor and multi-scene variations require ship images to be highly descriptive and distinguishable, so it is not possible with handcrafted features alone.

Different from shallow learning methods that rely on handcrafted features, deep learning methods, supported by powerful computing platforms and big data, can extract features directly from raw data through self-driven learning. Deep features can be seen as multi-level representations of the essence of objects, so they are more descriptive than handcrafted features. However, they have low interpretability. Existing CNN-based SAR ship models rely excessively on abstract deep networks, leading to a single cycle of network structure modification, training skill optimization, and loss function improvement.

As most CNN networks have a black-box behavior, improving model performance through optimizing CNN network architecture has become more challenging. Some SAR experts have therefore begun to study explainable artificial intelligence, exploring the importance of features or neurons in image analysis[4]. Other experts have incorporated prior knowledge of handcrafted features, exploring efficient ways to combine them with deep features. Extensive experiments have shown that handcrafted features can provide supplementary information to deep features, thereby enhancing the classification performance of CNN models[5, 6]. However, existing feature fusion methods simply concatenate deep features with

Manuscript received December 15, 2022; revised March 7, 2023; accepted xxxx xx, xxxx. This work was supported in part by the Natural Science Foundation of China under Grant 62172442 and Youth Science Foundation of Natural Science Foundation of Hunan Province 2020JJ5775. (*Corresponding author: Liu Yang.*)

Hao Zheng, Zhigang Hu, Liu Yang, Aikun Xu and Meiguang Zheng are with the School of Computer Science and Engineering, Central South University, Changsha 410083, China. E-mail:{ zhenghao, zghu, yangliu, aikunxu, zhengmeiguang}@csu.edu.cn

Keqin Li is with the Department of Computer Science, State University of New York, New Paltz, NY 12561 USA. E-mail: lik@newpaltz.edu.

Color versions of one or more of the figures in this letter are available online at https://ieeexplore.ieee.org.

Digital Object Identifier xxxxxxx

handcrafted features and directly input the high-dimensional fused feature vector to the fully connected (FC) layer, leading to a very complex optimization plane. This direct concatenation causes the computation of FC layer to grow exponentially and contains a lot of noise, which ultimately fails to provide satisfactory results. Alternatively, this concatenation considers all features to be equally important, ignoring different contributions of each feature. It results in the negative impact among different features, and causes the ultimate decision ability to be diminished.

To solve above issues, a multi-feature collaborative fusion network (MFCFNet) with deep supervision is proposed to achieve SAR ship classification. In MFCFNet, inspired by supervised learning, handcrafted feature auxiliary branches are added to the deep backbone network for the first time to improve the accuracy of the model through feature fusion. The relative importance of deep and handcrafted features is also considered, and an attention mechanism is used to adaptively balance the contribution of different features to the model performance. We introduce a new synergy loss to achieve knowledge interaction between all supervised branches. It normalizes the network training based on knowledge dynamically learned by all classifiers to achieve dynamic knowledge extraction and fusion. We perform a comprehensive evaluation on two public datasets (OpenSARShip and FUSAR-Ship) and carefully studied the performance of each module in MFCFNet. The experimental results demonstrate the effectiveness and robustness of MFCFNet with advanced SAR ship classification accuracy compared to modern CNN-based methods and other handcrafted feature fusion methods.

The main contributions of this article are specified as follows.

1) A novel feature fusion network, MFCFNet, is proposed to fuse traditional handcrafted features with deep features by adding an auxiliary branch for the first time to achieve better SAR ship classification.
2) In MFCFNet, the attention-guided feature fusion and contribution assignment module address the importance differences between deep and handcrafted features as well as the contribution imbalance.
3) In MFCFNet, branch loss in the knowledge supervision and collaboration module plays a role in judging the quality of the corresponding feature maps, and synergy loss can facilitate deep knowledge and handcrafted knowledge to learn from each other.
4) MFCFNet can enhance the classification performance of CNN networks by incorporating handcrafted features, and has demonstrated superior classification accuracy compared to modern handcrafted feature fusion methods on the OpenSARShip and FUSAR-Ship datasets.

The remainder of the article is organized as follows. Section II describes the related works about SAR ship classification based on handcrafted and deep features. Section III presents a detailed introduction of the proposed MFCFNet. Section IV shows experimental settings and comparative analysis of results. The ablation studies are introduced in Section V. Finally, Section VI provides the limitation and conclusion.

II. Related Work

In this section, we review previous research works about three main types: traditional handcrafted feature methods, modern deep feature methods, and feature fusion methods.

A. Traditional Handcrafted Feature Methods

Traditional handcrafted visual features are used to express low-level information, which amplify some visual features of an image, such as color, texture, shape, etc. These features are often accompanied by some interpretable theories.

Karvonen [7] pointed out that in addition to the areal backscattering, the information in SAR images was also in the edges. The canny edge detection algorithm can effectively improve the SAR image classification task. Similarly, some local features such as the mast position, were found to have more substantial discriminatory power in ship classification [8]. In addition, various feature frameworks have shown better performance. In [9], Li viewed the Gabor filter as a global operator to capture global texture features (e.g., orientation and scale) and the local binary pattern (LBP) as a local operator to characterize local spatial textures (e.g., edges, corners, and nodes). The classification was improved by combining Gabor features and LBP features from different perspectives. Wu et al. [10] analyzed the reflectivity histogram and estimated the values of some macroscopic features such as length, width and radar cross-sectional profile of the ship, which were evaluated using the fuzzy logic module. Lin et al. [11] designed an MSHOG feature describing the ship structure and used a task-driven dictionary learning algorithm to increase the ship separability. Although they achieved excellent performance in some specific settings, these methods were highly dependent on handcrafted features. These features were time-consuming and labor-intensive to extract manually, and they did not describe the image content in a comprehensive manner, limiting the classification accuracy in complex tasks.

B. Modern Deep Feature Methods

Compared with traditional handcrafted features, modern deep feature methods can automatically extract robust and adaptive deep features from labeled data. These methods have been widely used in SAR ship classification tasks and have achieved excellent performance due to the powerful multi-level characterization capability of deep features. For example, Shi et al. [12] applied 2D discrete fractional Fourier transform (2D-DFrFT) and two-branch CNN to obtain features. Wang et al. [13] developed a semi-supervised learning framework based on ResNet50, in which self-consistent augmentation rule enables the network to efficiently utilize unlabeled data. Dong et al. [14] designed a deeper SAR ship classification model by introducing a residual module. Zheng [15] proposed an ensemble network to improve the robustness and accuracy of classification by fusing multiple heterogeneous deep convolutional neural networks. Huang et al. [16] presented a novel method for CNNs, called Group Squeeze Stimulated Sparsely Connected Convolutional Networks (GSESCNNs), which made the concatenation of feature maps from different layers more efficient through sparse connection operations.

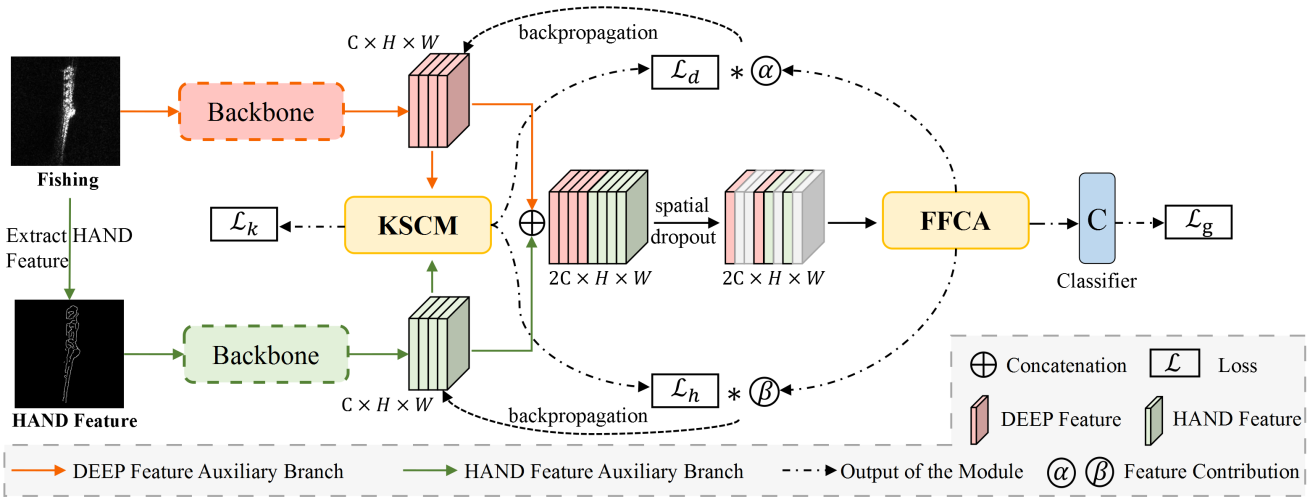


Fig. 1: The overall architecture of our MFCFNet. **KSCM**: Knowledge Supervision and Collaboration Module. **FFCA**: Feature Fusion and Contribution Assignment Module.

With the rise of artificial intelligence, the fact that deep feature-based SAR ship classifiers have achieved higher accuracy than traditional handcrafted feature classifiers, leading these models to uncritically discard handcrafted features. To further improve the characterization of deep features, the CNN network structure becomes increasingly complex, deeper, and uninterpretable. This cramming enhancement will soon face a bottleneck. In addition, in the modern information military, uninterpretable abstract features pose great risks in applications such as precision strikes.

C. Feature Fusion Methods

In order to enhance model interpretability and further improve CNN classification performance, some researchers in recent studies combined handcrafted features with deep features to achieve complementary effects.

Zhang [17] thoroughly investigated the effect of fusing handcrafted features with deep features at the internal FC layer and the terminal FC layer. The results showed that the best classification accuracy can be achieved by injecting handcrafted ones into the terminal FC layer, due to the handcrafted features had rich expert experience. He also pointed out that different CNN models differ in their sensitivity to handcrafted features. The worse the performance, the more significant accuracy improvement of the CNN model. In [18], Zhang et al. integrated handcrafted features into CNN models, demonstrating that mature handcrafted features can play an important role. They studied the fusion of 2D handcrafted features with deep features by first flattening 2D handcrafted features to one dimension, then using PCA to reduce the dimensionality of handcrafted features, and finally combining them in the FC layer. The article [19] proposed a HOG-ShipCLSNet network that combined HOG feature with multiscale CNN-based features at the FC layer to improve the classification accuracy. The HOG-ShipCLSNet used a multiscale mechanism to enrich the deep features, and then flattened the multiscale ones with HOG into 1D and fused them in the terminal FC layer to enhance the global representation. Similarly, Li [20] adopted feature alignment and adaptive weights to achieve multi-scale feature fusion. The low-scale images contained precise locations and contours, while the high-scale images provided complete contextual and structural information. In [21], the authors used multi-head encoders to extract complementary features of optical, SAR, and terrain modalities separately, and implemented multimodal knowledge fusion using an indicator-guided decoder.

In conclusion, the above methods simply concatenate handcrafted features with deep features and feed them into the FC layer, treating handcrafted features and deep features equally, without digging deeper into the relationship between 2D handcrafted features and deep features. At the same time, sending many numerical features directly to the FC layer will result in a very complex optimization hyperplane, causing the overfitting phenomenon often mentioned in the above article. Finally, the fused features may contain a lot of noise, making the network unable to converge.

III. Methodology

We propose a novel multi-feature collaborative fusion network framework with deep supervision, as shown in Fig. 1, containing two branches (DEEP Branch, HAND Branch) and two modules (KSCM, FFCA). In the HAND branch, we design a new location for feature injection. Specifically, the handcrafted feature map is treated as input, and the backbone network is used to deeply explore the contained expert knowledge. In this way, it can solve the optimization hyperplane problem caused by traditional feature fusion directly in front of the FC layer. At the same time, we design the KSCM module to improve the quality of feature maps by auxiliary supervision units and adopt synergy loss to promote dynamic information interaction between DEEP knowledge and HAND knowledge. Secondly, in order to reduce the overfitting problem caused by channel feature redundancy, we use the Spatial Dropout mechanism [22] to randomly zero out 50% of the feature maps in channel units. Finally, the feature map is input to the FFCA module, and the difference in importance between

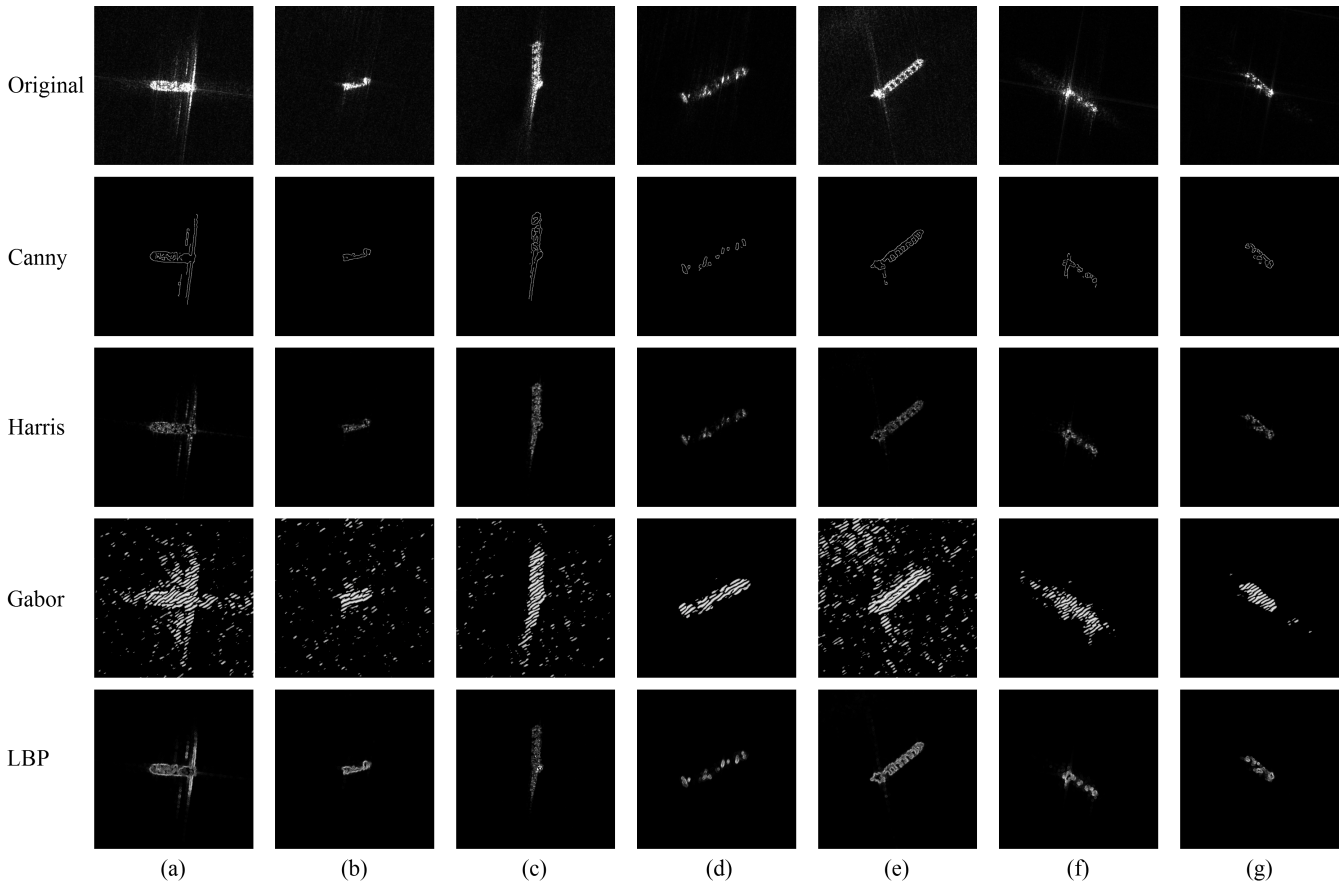


Fig. 2: (a)-(g) Columns are seven categories of SAR ship images in the FUSAR-Ship dataset, (a) Container, (b) General cargo, (c) Fishing, (d) Tanker, (e) Bulk, (f) Other cargo, and (g) Others. The first row is the original image, and the other rows are corresponding handcrafted feature visualizations.

deep and handcrafted features is weighed by the channel attention mechanism, and the total weights of deep and handcrafted features are output separately to solve the feature contribution imbalance problem. To our knowledge, this is the first work to achieve multi-feature collaborative fusion using handcrafted feature maps as input, and the experimental results demonstrate the effectiveness of MFCFNet. The modules are described in detail in the following sections.

A. Handcrafted Feature Extraction

Traditional handcrafted features enhance some of the visual information of an image, such as edges, corners and textures, which are often accompanied by some interpretable theories. According to the effect of previous use in the field of SAR ship classification and the requirement that the dimension of handcrafted features is 2-dimensional, we selected handcrafted features of each type, such as Canny edge, Harris corner, Gabor filter, and LBP histogram, respectively. As shown in Fig. 2, these handcrafted features all have the same dimensional as the original image. All methods are well known and each method is briefly explained below.

Canny edge feature is used to extract the edge information of SAR ships [23], which has the advantages of high localization accuracy and effective suppression of false edge points. Similar to the traditional edge detection step, the original image $f(x, y)$ is firstly smoothed and denoised by using the following Gaussian filter $G(x, y)$:

$$G(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \tag{1}$$

$$H(x, y) = f(x, y) * G(x, y) \tag{2}$$

where $H(x, y)$ is the smoothed image and $*$ is the representation of the convolution operator. σ is the standard deviation of $G(x, y)$, which affects the Gaussian filtering quality. Then, the gradient amplitude and direction of the pixel are calculated by computing the first-order partial derivatives in both directions for each pixel and transforming the coordinate system.

Harris corner feature is used to characterize the ship's corner information [24], which is more effective for ship positioning recognition. The Harris feature is defined by:

$$E(u, v) = \sum_{(x,y)} w(x, y) \times [f(x + u, y + v) - f(x, y)]^2 \tag{3}$$

where $w(x, y)$ is a window function, which can also be a Gaussian function $G(x, y)$. When the w is shifted in both x and y directions, the $E(u, v)$ is calculated.

Gabor filter [25, 26] feature is also widely used for ship classification because it can represent the spatial structure at

different scales and orientations, enhancing the global rotation invariance. The mathematical expression of the 2D Gabor function takes the following form:

$$g(x,y;\lambda,\theta,\sigma,\gamma)=\exp\left(-\frac{x_0^2+\gamma^2y_0^2}{2\sigma^2}\right)\cdot\exp\left(i\left(2\pi\frac{x_0}{\lambda}+\psi\right)\right) \tag{4}$$

$$\begin{cases} x_0 = x\cos\theta + y\sin\theta \\ y_0 = -x\sin\theta + y\cos\theta \end{cases} \tag{5}$$

where (x, y) is the spatial domain coordinate, λ is the wavelength, θ is the directional separation angle of the Gabor core, γ is the spatial aspect ratio, ψ is the phase shift.

LBP descriptor is a simple and effective pixel-based texture descriptor for extracting spatial texture features of ship images [27]. The descriptor computes each neighborhood pixel using the centroid pixel gray value as a threshold, which can be expressed as:

$$\mathrm{LBP}\left(x_c,y_c\right)=\sum_{0}^{p}2^p s\left(i_p-i_c\right) \tag{6}$$

$$s(x)=\begin{cases} 1, x>0 \\ 0, x<0 \end{cases} \tag{7}$$

where (x_c, y_c) is the pixel coordinate, p is the pth pixel in the domain, c is the pixel in the neighbor center, i_p is the pth pixel value, i_c is the pixel value in the neighbor center. Then, the whole LBP feature map is counted using the histogram to obtain the final LBP feature vector histogram.

B. Knowledge Supervision and Collaboration Module

Knowledge supervision and collaboration module (KSCM) consists of an auxiliary feature supervision unit and a knowledge collaboration learning unit, as shown in Fig. 3. Briefly, The auxiliary feature supervision unit is responsible for providing supervision on the output features of each branch and introducing accompanied objective functions $\mathcal{L}_d$ and $\mathcal{L}_h$ to improve the convergence rate of the model. The knowledge collaboration learning unit uses a knowledge synergy strategy $\mathcal{L}_k$ to facilitate the information interaction between deep features and handcrafted features.

In the auxiliary feature supervision unit, we add auxiliary classifiers after two feature extraction branches. HAND branch is used for low-level visual features and the other DEEP branch is used for high-level semantic features. Let $D=\{(x_1,y_1),\ldots,(x_n,y_n)\}$ be an annotated SAR dataset having N training samples collected from K ship classes, where each member (x_i, y_i) contains $x_i \in \mathbb{R}^d$ and y_i is the corresponding ship category. Let $W=\{W_d, W_h, W_g\}$ be the weights of the DEEP branch, HAND branch and global network that needs to be learned. Hence, $f(W, x_i)$ is the k-dimensional output vector of the W branch for a training sample x_i. According to the deeply supervised network fusing the losses of each branch, the global optimization objective can be expressed by the following equation:

$$\underset{W_g,W_d,W_h}{\operatorname{argmin}}\ \mathcal{L}_g\left(W_g,D\right)+\alpha\mathcal{L}_d\left(W_d,W_g,D\right)+\beta\mathcal{L}_h\left(W_h,W_g,D\right) \tag{8}$$

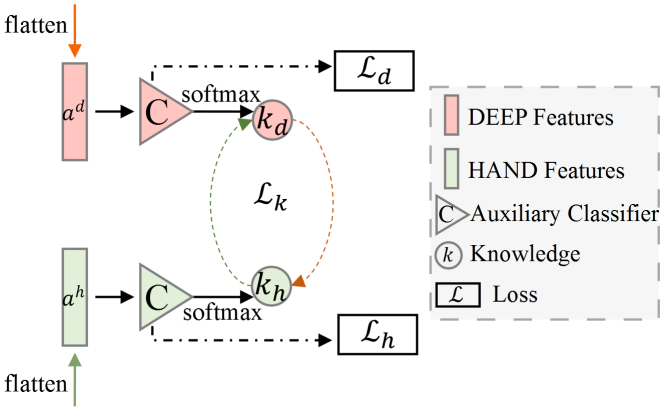


Fig. 3: Knowledge Supervision and Collaboration Module (KSCM). $\mathcal{L}_k$ denotes synergy loss, $\mathcal{L}_d$ denotes DEEP branch loss, $\mathcal{L}_h$ denotes HAND branch loss.

where $\mathcal{L}$ is calculated with the cross-entropy cost function, $\mathcal{L}_g$ is the default loss, the auxiliary loss $\mathcal{L}_d$ and $\mathcal{L}_h$ are the corresponding DEEP and HAND auxiliary classifiers evaluated on the training set, making the learned deep and handcrafted features more discriminative and robust.

In deep supervised networks, Sun [28] states that setting a fixed value of 1.0 for α and β gives the same performance as the best CNN model trained by the ZERO-ing strategy [29]. However, we found in our experiments that when deep features are added, the two branches contribute differently to the final classification, and if the same weights are used it will lead to poor fusion. Therefore how to set the weights of α and β, we will introduce in section III-C.

In the knowledge collaboration learning unit, the knowledge synergy strategy can facilitate the aggregation of deep features and handcrafted ones to improve the information consistency among them. Specifically, the class probability outputs of the two auxiliary classifiers on the training data are utilized as learned knowledge to regularize the network's training. The knowledge matching between the DEEP auxiliary classifier and the HAND auxiliary classifier is a KL divergence

$$\mathcal{L}_k=-\frac{1}{N}\sum_{i=1}^{N}\left(\mu_{dh}f_d\log\frac{f_d}{f_h}+\mu_{hd}f_h\log\frac{f_h}{f_d}\right) \tag{9}$$

where f_d and f_h are the class probability outputs of DEEP and HAND classifiers using the softmax function, and μ weights the information loss of knowledge matching among them. In this study, to make the knowledge learned by the classifiers transferable to each other, we set $\mu = 1$ and keep them fixed like in [28].

C. Feature Fusion and Contribution Assignment Module

Traditional multi-feature fusion methods usually use a simple concatenated feature map, and this concatenation defaults the deep and handcrafted features to have the same important information. In order to more clearly characterize the features of different channels after concatenation, as shown in Fig. 4, we designed a feature fusion and contribution assignment

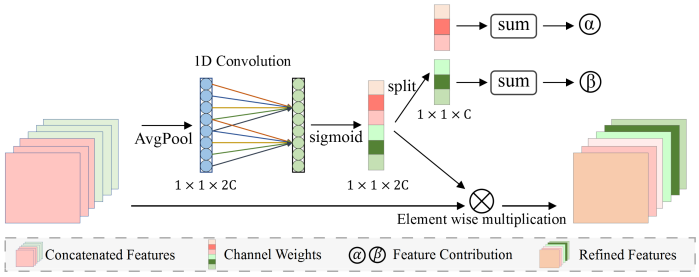


Fig. 4: Feature fusion and contribution assignment module (FFCA), which outputs refined features and feature contributions α, β

module(FFCA). Similar to the widely applied attention mechanism [30, 31], FFCA module uses global average pooling to aggregate the spatial dimensional information of the multi-channel feature maps $F \in R^{2C\times H\times W}$, compressed into a $1 \times 1 \times 2C$ sequence of real numbers. Then, the feature sequence is fed into a shared multilayer perceptron(MLP) to learn the relationship between each channel and generate a more representative feature vector. After that, using the sigmoid function obtains the feature channel weights. It outputs the two types of feature weights according to the summation operation of the deep feature channel and the handcrafted feature channel, respectively. Finally, we use the feature channel weights multiplied by the input feature map to obtain the final channel attention map. The equation of the channel attention mechanism is shown in the following equation:

$$W(F) = \text{sigmoid}(\text{MLP}(\text{AvgPool}(F))) \tag{10}$$

where F represents the concatenated feature map and $W(F)$ represents the feature weights of each channel. Thus the deep feature weights α and handcrafted feature weights β are:

$$\alpha = \sum_{c=1}^{C} W_c(F), \beta = \sum_{c=C+1}^{2C} W_c(F) \tag{11}$$

We use the deep feature weights α and the handcrafted feature weights β to measure the corresponding supervised loss functions, thus balancing the contribution of different features to the model classification. As a result, combining the loss function of the previous section with the contribution weights of this section, the total loss function of the whole framework is:

$$\mathcal{L}_{total} = \mathcal{L}_g + \alpha\mathcal{L}_d + \beta\mathcal{L}_h + \mathcal{L}_k \tag{12}$$

where $\mathcal{L}_g$ is the default loss, $\mathcal{L}_d$ and $\mathcal{L}_h$ can play the role of judging the good or bad quality of the corresponding feature maps, and $\mathcal{L}_k$ can promote the auxiliary classifiers to learn from each other.

IV. Experiment And Result Analysis

All programs are implemented in Python language, and the CNN network is implemented using the open source PyTorch framework, with the handcrafted feature extraction methods partially derived from the skimage library. The model inference is accelerated using CUDA11.6 platform calling GPU.

TABLE I: Distribution of the SAR ship datasets.

Dataset	Category	Training	Test	All
OpenSARShip-1.0	Bulk	338	328	666
	Container	338	808	1146
	Tanker	338	146	484
FUSAR-Ship	Container	1219	523	1742
	General cargo	1205	517	1722
	Fishing	1101	473	1574
	Tanker	1215	521	1763
	Bulk	1150	494	1644
	Other cargo	1214	520	1734
	Others	1211	521	1732

(a) Bulk (b) Container (c) Tanker

Fig. 5: Three-category of SAR ship images in the OpenSARShip-1.0 dataset.

A. Data Description

To evaluate the feasibility and effectiveness of MFCFNet to fuse handcrafted features, we perform extensive experimental analysis on two popular SAR ship datasets like other scholars [32–34]. The distribution ratio and preprocessing of the datasets are the same as our previous work [15]. Table I lists the distribution of two datasets, such as categories, totals, and allocations. The validation set is randomly divided from the training set by using the three-fold cross validation method, which is used to verify a variety of hyperparameters. On the basis of the minimum average error hyperparameters, the training set and the validation set are combined to retrain the final model, and then its generalization ability is tested through the test set.

(1) *OpenSARShip Dataset*: The OpenSARShip images were derived from the dual-polarization SAR detected by the European Space Agency's Sentinel 1 satellite, including both VH and VV polarization channels. Combining the coordinates and categories provided by Huang and the experimental setup of earlier research [13, 19], three main types of ships are extracted and the same training-test ratio is set to solve the sample imbalance problem. Additionally, the resolution of this dataset was decreased compared to FUSAR-Ship. As shown in fig. 5, there are three types of ships: bulk, container and tanker.

(2) *FUSAR-Ship Dataset*: The FUSAR-Ship dataset was extracted from 126 hyperfine images acquired on the quad-polarization Gaofen-3 satellite, and had a greater variety of

TABLE II: SAR ship classification results with and without handcrafted features on OpenSARShip and FUSAR-Ship. For each network, we run each method 5 times and report the "mean±std" accuracy. () indicates the performance improvement for Baseline. ABG refers to the average backbone gain. AFG refers to the average feature gain. The best gain for BackBone and Handcrafted features are highlighted in **Red** and **Blue** respectively.

Dataset	Backbone	Baseline	**+ Canny**	**+ Harris**	**+ Gabor**	**+ LBP**	**ABG(%)**
OpenSARShip	AlexNet [35]	70.47±1.45	**77.70±0.36**(+7.23)	74.56±0.94(+4.09)	75.93±0.67(+5.46)	77.25±0.44(+6.78)	5.89
	VGG-11 [36]	69.90±0.96	76.66±0.48(+6.76)	76.13±0.52(+6.23)	**76.85±0.61**(+6.95)	76.72±0.46(+6.82)	**6.69**
	VGG-16 [36]	70.62±1.18	75.93±0.94(+5.31)	74.43±0.36(+3.81)	**77.97±0.94**(+7.35)	77.05±0.59(+6.43)	5.73
	ResNet-18 [37]	72.66±0.87	74.49±0.56(+1.83)	74.19±0.91(+1.53)	**74.62±0.85**(+1.96)	72.79±0.36(+0.13)	1.36
	DenseNet-121 [38]	73.59±1.44	76.07±0.93(+2.48)	**78.60±0.21**(+5.01)	78.36±0.30(+4.77)	77.18±0.33(+3.59)	3.96
AFG(%)		—	4.72	4.13	**5.30**	4.75	—
FUSAR-Ship	AlexNet [35]	77.64±0.83	79.70±0.56(+2.06)	79.54±0.31(+1.9)	**79.91±0.49**(+2.27)	79.15±0.22(+1.51)	1.94
	VGG-16 [36]	80.30±0.19	84.50±0.22(+4.2)	82.07±0.31(+1.77)	**85.13±0.13**(+4.83)	82.51±0.19(+2.21)	3.25
	ResNet-18 [37]	78.94±0.56	**83.04±0.24**(+4.1)	82.39±0.19(+3.45)	82.48±0.26(+3.54)	81.66±0.17(+2.72)	**3.45**
	ResNet-152 [37]	80.48±0.33	80.79±0.20(+0.31)	82.21±0.17(+1.73)	**82.76±0.21**(+2.28)	80.56±0.22(+0.08)	1.1
	DenseNet-121 [38]	82.18±0.59	84.86±0.43(+2.68)	83.53±0.14(+1.35)	**85.79±0.10**(+3.61)	83.82±0.25(+1.64)	2.32
	DenseNet-201 [38]	83.35±0.47	85.21±0.13(+1.86)	85.29±0.21(+1.94)	**87.23±0.26**(+3.88)	83.77±0.18(+0.42)	2.01
AFG(%)		—	2.54	2.02	**3.40**	1.43	—

ships compared to the OpenSARShip dataset. As shown in the first row of Fig. 2, seven types of ships from are used in the experiment, i.e., bulk, container, fishing, tanker, general cargo, other cargo, and others. We use the same data pre-processing method and training–testing ratio as in [15]. Specifically, the image is first padded by 5 pixels to both sides, and then 224×224 crops are randomly sampled from the padded image or its horizontal flips.

B. Experiment settings

(1) *Backbone and implementation details on OpenSARShip.* We use the four most representative CNN architectures for evaluation, namely AlexNet [35], VGG-16[36], ResNet-18[37], and DenseNet-121[38]. We employ the open-source model code in Pytorch and train each backbone network following the standard settings. For ResNet-18, we use an SGD optimizer with the momentum of 0.9 and the learning rate as 0.001. The rest of the models use Adam optimizer with a learning rate as 0.0001 and the weight decay as 5×10^{-4}. All models are trained with 100 epochs and the batchsize is set to 16.

(2) *Backbone and implementation details on FUSAR-Ship.* Due to the larger FUSAR-Ship dataset, we add two deeper models to test the validity of MFCFNet, namely ResNet101[37] and DenseNet201[38]. For ResNet-18 and ResNet-101, we use an SGD optimizer with the momentum of 0.9 and the learning rate set to 0.01. The rest of the models use Adam optimizer with a learning rate as 0.001 and the weight decay as 5×10^{-4}. All models are trained for 100 epochs, and the learning rate is decayed by 10% at the 60^{th} epoch, and the batchsize is set to 32.

(3) *Auxiliary classifier implementation details.* The auxiliary classifiers on both branches have the same structure as the classifiers in the original backbone network.

C. Metric index

For the SAR ship classification task, we use the Accuracy, F1, Precision, and Recall metric to measure the classification performance and compare with the state of the arts.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{13}$$

$$F1 = \frac{2 \times TP}{2 \times TP + FN + FP} \tag{14}$$

$$Precision = \frac{TP}{TP + FP} \tag{15}$$

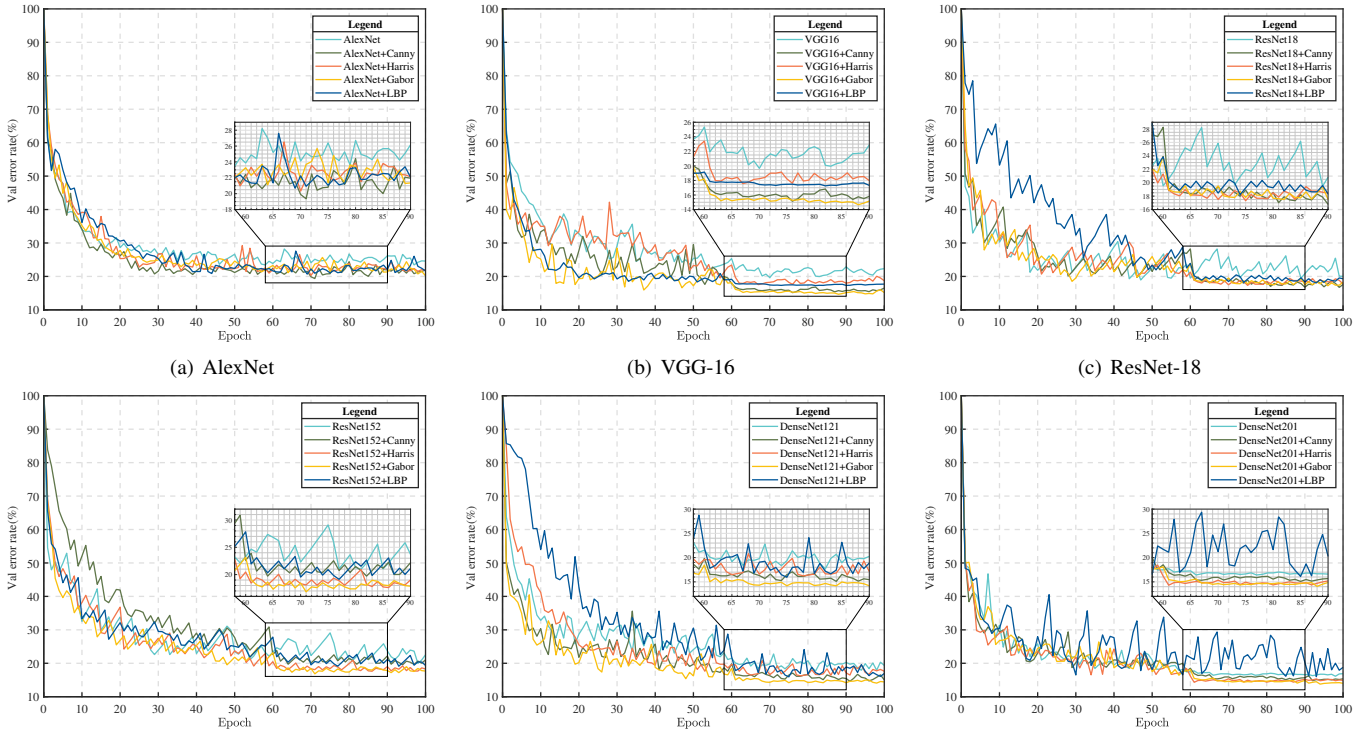
$$Recall = \frac{TP}{TP + FN} \tag{16}$$

where *TP*, *TN*, *FP*, and *FN* denotes the number of correctly classified ships, number of correctly classified opposite classes, number of incorrectly classified ships, and number of the misclassified ships, respectively.

D. SAR Ship Classification Results

Table II shows the SAR ship classification results of MFCFNet on OpenSARShip and FUSAR-Ship with and without handcrafted features. In the table, the *Backbone* refers to the deep features, the *Baseline* denotes the standard training scheme, and *Canny*, *Harris*, *Gabor*, and *LBP* indicate the corresponding handcrafted feature fusion schemes. We run each combination 5 times and report the "mean±std" accuracy. For better comparison, we also present the average gain *ABG* and *AFG*. *ABG* refers to the average gain of the identical backbone combined with different handcrafted features. Similarly, *AFG* refers to the average gain of the same handcrafted feature combined with different backbones.

Results on the OpenSARShip are summarized in Table II where our method MFCFNet consistently improves



(a) AlexNet (b) VGG-16 (c) ResNet-18

(d) ResNet152 (e) DenseNet-121 (f) DenseNet-201

Fig. 6: The training performance of AlexNet, VGG-16, ResNet-18, ResNet-152, DenseNet-121, DenseNet-201 with and without handcrafted features on FUSAR-Ship.

the performance of all backbones. Among them, Densenet-121+Harris achieved the highest accuracy of 78.60%. From the perspective of average backbone gain, we find that the accuracy improvement is more significant for the original model with poorer performance. For example, the original VGG-11 model has 69.90% classification accuracy, and the average gain after adding handcrafted features is 6.69%. However, the original DenseNet-121 model has 73.59% accuracy, and the average gain after adding features is 3.96%. The average gain of the ResNet-18 model is only 1.36%, partly because of the high accuracy of the original network, but mainly because the sparse residual summation operation in the network disrupts the feature information flow to some extent. Meanwhile, the same model has sensitivity differences to different handcrafted features. For example, the accuracy improvement of the VGG-16 model is 7.35% after fusing Gabor features and only 3.81% after fusing Harris. Therefore, we need to further consider the intrinsic relationship between deep and handcrafted features. From the average feature gain perspective, the textures features are described by the Gabor and LBP for the Top-2 gains, which are 5.30% and 4.75%, respectively. It shows that texture features have a substantial gain for deep features on OpenSARShip. The experimental results powerfully illustrate the effectiveness of MFCFNet for the fusion of handcrafted features with deep features.

Results on the FUSARShip are similar to those on OpenSARShip, and MFCFNet achieves an effective accuracy improvement even on larger datasets and deeper networks. The accuracy results of all backbone networks are consistent with that reported in the literature [19]. Benefiting from the proposed handcrafted feature fusion, MFCFNet improves 1.94%, 3.25%, 3.45%, 1.1%, 2.32%, and 1.73% in average accuracy gain for AlexNet, VGG-16, ResNet-18, ResNet-152, DenseNet-121 and DenseNet-201, respectively. The accuracy

TABLE III: Confusion Matrix Of MFCFNet Classification Results on OpenSARShip.

True \ Predicted	Bulk	Container	Tanker	**Recall(%)**
Bulk	251	65	12	76.52
Container	126	636	46	78.71
Tanker	9	16	121	82.88
Precision(%)	65.03	88.70	67.60	**Accuracy=78.62%**
F1(%)	70.31	83.41	74.46	

TABLE IV: Confusion Matrix Of MFCFNet Classification Results on FUSAR-Ship.

True \ Predicted	Container	General cargo	Fishing	Tanker	Bulk	Other cargo	Others	**Recall(%)**
Container	509	0	14	0	0	0	0	97.32
General cargo	0	511	0	0	6	0	0	98.84
Fishing	12	0	321	14	13	4	109	67.86
Tanker	1	0	5	499	0	7	9	95.78
Bulk	4	16	24	3	427	8	12	86.44
Other cargo	2	0	15	28	17	426	32	81.92
Others	2	5	78	2	10	15	409	78.50
Precision(%)	96.04	96.05	70.24	91.39	90.27	92.61	71.63	**Accuracy**
F1(%))	96.68	97.43	69.03	93.53	88.31	86.94	74.91	**86.92%**

TABLE V: Comparison of MFCFNet with handcrafted feature-based, deep feature-based and state-of-the-art feature fusion methods on two datasets.

Feature	Method	OpenSARShip				FUSAR-Ship			
		Accuracy(%)	Recall(%)	Precision(%)	F1(%)	Accuracy(%)	Recall(%)	Precision(%)	F1(%)
Handcrafted Feature	SVM [39]	55.74	56.77	49.51	52.91	67.74	68.13	67.51	67.82
	Decision Tree [40]	57.38	60.09	**54.85**	**57.32**	65.57	65.79	65.30	65.54
	Random Forest [41]	57.05	59.56	52.94	56.04	67.05	67.33	67.65	67.49
	MLP [42]	**59.67**	**60.70**	53.44	56.84	**72.35**	**72.25**	**72.98**	**72.61**
Deep Feature	Wide-ResNet-101 [43]	73.04±0.73	72.12±2.77	67.37±1.46	69.62±1.27	80.98±0.53	81.02±0.54	81.10±0.51	81.06±0.52
	MobileNet-v1 [44]	69.91±1.08	66.30±2.87	63.49±2.40	64.83±2.03	77.61±0.54	77.79±0.56	77.92±0.59	77.86±0.56
	SqueezeNet-v1.0 [45]	72.15±1.25	71.47±1.31	66.73±1.70	69.01±1.28	78.76±0.38	78.87±0.40	79.07±0.52	78.97±0.44
	Inception-v4 [46]	72.44±0.70	69.26±3.16	67.43±2.39	68.28±1.97	80.50±0.37	80.55±0.40	80.89±0.55	80.72±0.45
	Xception [47]	73.74±0.86	71.56±3.00	68.60±1.67	70.00±1.29	77.29±0.38	77.42±0.39	77.39±0.36	77.41±0.37
	GSESCNNs [16]	**74.98±1.46**	**74.74±1.60**	**69.56±2.38**	**72.04±1.60**	**83.19±0.31**	**83.19±0.41**	**83.34±0.31**	**83.27±0.35**
Feature Fusion	DUW-Cat-FN [17]	78	**78.65**	**72.99**	**75.21**	**86.86**	85.49	85.28	85.22
	HOG-ShipCLSNet [19]	**78.15±0.57**	77.87±1.14	72.42±1.06	75.04±0.68	86.69±0.47	**86.62±0.51**	**86.54±0.50**	**86.58±0.50**
	Internal FC layer	74.75±1.21	73.57±2.11	71.64±2.52	73.31±1.98	84.25±0.42	84.16±0.52	84.29±0.43	84.29±0.42
	Terminal FC layer	74.10±1.42	73.22±1.89	70.21±2.21	72.29±2.19	83.17±0.51	83.22±0.54	83.08±0.48	83.16±0.54
	MFCFNet	**78.60±0.21**	**79.37±0.58**	**73.78±0.62**	**76.06±0.93**	**87.23±0.26**	**86.67±0.43**	**86.89±0.45**	**86.69±0.41**

The standard deviation of DUW-Cat-FN are not given in the source.

improvement of ResNet-152 and DenseNet-201 with deeper layers is lower than that of the corresponding shallow networks, indicating that the deeper networks contain richer semantic information. ResNet-18 achieves the best average backbone gain of 3.45% on a larger dataset in contrast to OpenSARShip. So the performance of residual blocks can be exploited when the dataset is sufficiently complex and diverse. The Gabor feature also achieves the best average feature gain of 3.40% on FUSAR-Ship. The training performance of all backbones with and without handcrafted features is shown in Fig. 6. From the Fig. 6, we can find that the backbone network combined with handcrafted features can accelerate the convergence speed and improve the accuracy, but each network has sensitivity differences to various handcrafted features. For example, the DenseNet-121 network, after combining Canny and Gabor features, obviously converges faster than the original network. However, the combination of Gabor causes oscillations in training process. The internal mechanism of this phenomenon needs to be further investigated in the future. In conclusion, as the backbones become deeper (e.g., ResNet-152 and DensenNet-201)/the datasets become larger (e.g., FUSAR-Ship), our method MFCFNet has the same significant accuracy improvement for all backbones.

Table III and Table IV show the top-1 accuracy DenseNet121+Harris and DenseNet201+Gabor on both datasets, and illustrate the classification performance for each ship category in the form of confusion matrices. The tables also have many misclassifications due to the significant interference of background noise in the images of the two datasets. However, Table IV performs better than Table III because the FUSAR-Ship dataset has a higher resolution, can learn more ship features, and has an accuracy of 86.92%, higher than the 78.62% of OpenSARShip. Clearly, the confusion in category prediction on the FUSAR-Ship dataset mainly occurs in Fishing and Others, as these two ship types have similar geometric shapes. The various types of cargo such as container, general cargo, tankers, and bulk achieved better classification performance.

E. Comparison Results

In the comparison experiments, the best feature combinations DenseNet121+Harris on OpenSARShip and DenseNet-201+Gabor on FUSARShip are used as benchmarks, and then compared them with handcrafted feature-based, deep feature-based, and state-of-the-art feature fusion methods, respectively. The Harris feature is used in OpenSARShip and the Gabor feature is used in FUSARShip.

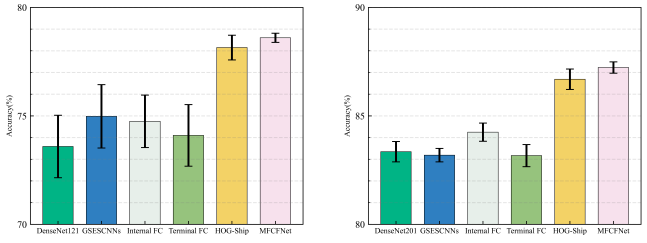
Comparison with handcrafted feature-based methods. In the first item of Table V, four methods based on handcrafted features are listed, SVM, Decisoin Tree, Random Forest and Multilayer Perceptron (MLP). From the table V, the best accurate MLP combined with handcrafted features can reach 59.67% and 72.35%, both much lower than our MFCFNet. But thinking differently, these methods demonstrate the validity of handcrafted features Harris and Gabor that can be used for SAR ship classification. In addition, all handcrafted feature methods are inferior to the deep feature-based methods. This is the reason we use handcrafted features to provide complementary information to the deep features.

Comparison with deep feature-based methods. Combining the backbone network in Table II and the deep feature-based method in Table V, the improved CNN network GSESCNNs proposed by Huang [16] achieve the best accuracy of 74.98% and 83.19%, respectively, which is 5% lower than our MFCFNet approach. The result demonstrates the effectiveness of combining handcrafted features with deep features. It further illustrates that SAR classification should not be caught in a single cycle of network structure modification, training technique optimization, etc. Instead, combining deep features with handcrafted features can solve the aforementioned limitations.

Comparison with feature fusion methods. Among the feature fusion methods for SAR ship classification in Table V,

TABLE VI: Ablation experiments of feature fusion unit. *Baseline* refers to not using handcrafted features, *Attention Removed* refers to not using the attention mechanism, () indicates the performance gain for *Attention Removed*.

Dataset	Networks	Baseline	Attention Removed	SAM [48]	CBAM [49]	**Ours**
OpenSARShip	VGG16+Gabor	70.62±1.18	74.75±1.26	75.74±0.61(+0.99)	**76.99±0.59**(+2.24)	**77.97±0.94**(+3.22)
	AlexNet+Canny	70.47±1.45	76.07±0.52	76.72±0.43(+0.65)	**77.38±0.75**(+1.31)	**77.70±0.36**(+1.63)
FUSAR-Ship	VGG16+Gabor	80.30±0.19	**84.73±0.21**	78.11±0.17(-6.62)	83.83±0.22(-0.90)	**85.13±0.13**(+0.40)
	VGG16+Canny	80.30±0.19	83.08±0.26	83.18±0.49(+0.10)	**84.04±0.24**(+0.96)	**84.50±0.22**(+1.42)



(a) OpenSARShip (b) FUSARShip

Fig. 7: The comparison results of "mean±std" accuracy between the proposed MFCFNet and state-of-the-art.

the state-of-the-art methods are DUW-Cat-FN [17] and HOG-ShipCLSNet [19] proposed by Zhang. Since we study two-dimensional manual features, the dimensionality difference between deep features and manual features that are flattened to one-dimensional is large, and direct concatenation to the FC layer will lead to feature confusion and overfitting. Both of the above methods fuse handcrafted features in the terminal FC layer and achieve the best classification accuracy of 78.15% and 86.86% on both datasets. Therefore, for better comparison, we use the regular internal FC layer and terminal FC layer as well to fuse handcrafted features separately, which have significantly lower performance than our MFCFNet. Since we are study 2D handcrafted features, the dimension difference between the deep feature and handcrafted feature which is flattened to 1D is large, and the direct concatenation to the FC layer leads to feature confusion and overfitting. From Table V, MFCFNet can achieve a state-of-the-art classification accuracy of 78.60% and 87.23%. We find that for all experimental results on the OpenSAR dataset, the precision value is significantly lower than the other three metrics, which is due to the unbalanced test samples in this dataset. Precision can prevent the problem of indicator failure caused by unbalanced positive and negative samples. Similarly, MFCFNet achieves the highest performance in Precision on both datasets. Combined with Fig. 7, the standard deviation produced by MFCFNet is much lower than the deep feature methods and feature fusion methods, which is 0.21 and 0.26 on the two datasets, respectively. The results show that in each random experiment, KSCM and FFCA modules make the fusion features play a maximum and stable role.

V. Ablation Study

In MFCFNet, the feature fusion unit is the core of the FFCA module, the knowledge collaborative learning unit is the core of the KSCM module, and both modules contain an auxiliary feature supervision unit with feature contribution weight. Therefore, we have divided the ablation experiments according to these units. We perform ablation experiments with the top-2 gain combinations on both datasets to allow a more pronounced study of each unit's effectiveness. The top-2 gain on OpenSARShip are VGG16+Gabor(+7.35) and AlexNet+Canny(+7.23). The top-2 gain on FUSAR-Ship are VGG16+Gabor(+4.83) and VGG16+Canny(+4.2). All ablation experiments were also run 5 times to report the "mean±std" accuracy.

A. Ablation Study on Feature Fusion Unit

We conduct several ablation studies to investigate the effectiveness of attentional mechanisms in feature fusion units and the effect of different attentional mechanisms on classification accuracy, including the commonly used types of attention: the spatial attention module (SAM) [48] and the convolutional block attention module (CBAM) [49]. From Table VI, the performance of the network can be improved by using any attention mechanism on OpenSARShip, and the lower the Attention Removed value, the more significant the improvement. However, due to the more diverse and complex ship categories in FUSAR-Ship, and the handcrafted features and deep features characterizing ship information from two different perspectives, the SAM focusing more on spatial pixel relationships can cause feature representation confusion, resulting in a negative impact. The CBAM performs a little better as it is a mixed spatial and channel attention mechanism. Our method based on the channel attention mechanism performs a little better, coming from paying more attention to the relationship between deep features and handcrafted features of each channel to eliminate the effect of feature confusion.

B. Ablation Study on Auxiliary Feature Supervision Unit

We conduct some ablation experiments to verify the effectiveness of auxiliary feature supervision loss and feature contribution weights. Here, we set the loss weight values to 0, 0.2, 0.8 and 1 for the experiments. α represents the DEEP branch loss weight and β represents the HAND branch loss weight. When $\alpha, \beta = 0$, it means no auxiliary loss is used. The model gain at this time is attributed to the attention mechanism and knowledge synergy loss. From Table VII, we can make the following observations: (1) When using the weaker handcrafted feature Canny, if β is set greater than or equal to α, it makes the model pay more attention to the HAND branch in the backpropagation process, resulting in the

TABLE VII: Ablation experiments of auxiliary feature supervision unit. $\alpha, \beta = 0$ means remove the auxiliary supervision loss, () indicates the performance gain for $\alpha, \beta = 0$.

Dataset	Networks	Baseline	$\alpha, \beta = 0$	$\alpha, \beta = 1$	$\alpha = 0.2, \beta = 0.8$	$\alpha = 0.8, \beta = 0.2$	Ours
OpenSARShip	VGG16+Gabor	70.62±1.18	71.82±1.03	74.11±0.53(+2.29)	71.16±0.60(-0.66)	**74.21±0.82(+2.39)**	**77.97±0.94(+6.15)**
	AlexNet+Canny	70.47±1.45	74.43±0.59	73.77±0.67(-0.66)	73.11±0.43(-1.32)	**75.74±0.60(+1.31)**	**77.70±0.36(+3.27)**
FUSAR-Ship	VGG16+Gabor	80.30±0.19	82.86±0.17	**84.89±0.10(+2.03)**	83.61±0.16(+0.75)	84.39±0.22(+1.53)	**85.13±0.13(+2.27)**
	VGG16+Canny	80.30±0.19	**83.78±0.54**	70.54±0.62(-13.24)	69.69±0.83(-14.09)	82.10±0.55(-1.68)	**84.50±0.22(+0.72)**

TABLE VIII: Ablation experiments of knowledge collaboration learning unit. () indicates the performance gain for *Loss Removed*.

Dataset	Networks	Baseline	Loss Removed	**Ours**
OpenSAR	VGG16+Gabor	70.62±1.18	77.05±0.46	**77.97±0.94(+0.92)**
	AlexNet+Canny	70.47±1.45	74.10±0.43	**77.70±0.36(+3.60)**
FUSAR	VGG16+Gabor	80.30±0.19	84.97±0.31	**85.13±0.13(+0.16)**
	VGG16+Canny	80.30±0.19	82.90±0.44	**84.50±0.22(+1.60)**

model shaking violently and difficult to converge during the training process, which eventually leads to lower performance than Baseline; (2) When using stronger Gabor features, a larger gain can be produced for the deep features. The model performance gets better as α keeps increasing. (3) Our method uses feature contribution degree to set α and β, which can adaptively measure the relative importance of different deep features with handcrafted features, and finally achieve the maximum accuracy gain. In conclusion, the results in Table VII strongly prove that the auxiliary features supervision loss and feature contribution degree, which can balance the importance between features, make the handcrafted features and deep features complement each other.

C. Ablation Study on Knowledge Collaboration Learning Unit

We conduct some ablation experiments to verify the effectiveness of synergy loss in the knowledge collaborative learning unit. Here, we remove the knowledge synergy loss and keep the rest of MFCFNet in table VIII. Specifically, the best accuracy gain of 3.6% was achieved on OpenSARShip using AlexNet+Canny, and effective results were also achieved on FUSAR-Ship. These results demonstrate the significance of the knowledge synergy loss and the effectiveness of our approach, which leads deep knowledge and handcrafted knowledge to learn from each other, achieving a dynamic collaborative process for the same task.

VI. Limitation and Conclusion

In this paper, we proposed a novel collaborative multi-feature fusion network with deep supervision to better achieve handcrafted features to provide complementary information to deep features. An auxiliary feature supervision unit and a knowledge collaborative learning unit are designed, the former realized high quality extraction of feature maps for each branch, and the latter achieved collaborative learning of deep and handcrafted knowledge. In addition, a feature fusion and contribution assignment module based on the channel attention mechanism is designed, which can solve the problem of importance difference between deep features and handcrafted features and the unbalanced contribution of different features. Extensive experiments have shown that our proposed MFCFNet outperforms single deep features and multi-feature models based on Internal FC Layer and Terminal FC Layer fusion, and exhibited better performance than the current state-of-the-art related methods. Therefore, MFCFNet is indeed able to achieve better SAR ship classification reliably and realistically.

Our current study has some limitations. First, the model performance is improved when fusing one handcrafted feature with a deep feature, but sustained performance improvement cannot be achieved when more than two handcrafted features are fused with a deep feature at the same time. The main reason for our analysis of this phenomenon is the feature redundancy problem. Second, we observe that the size of MFCFNet is twice that of backbone, which is mainly related to the number and complexity of features auxiliary branches. Therefore, weighing the model size and the expected increase in accuracy, we believe that the current increase in model size is justified. More importantly, all auxiliary classifiers are discarded in the inference process, so there is no additional computational overhead.

Our future work is as follows:

1) Study the intrinsic relationship between deep and handcrafted features to implement recommending the best handcrafted features to different CNN networks.
2) Solve the feature redundancy problem and extract common features in deep and handcrafted features, thus improving model robustness and extending the MFCFNet framework to any number of feature fusions.
3) Study and evaluate various representational capabilities of deep and handcrafted features and build a feature capability matrix.
4) The existing feature fusion methods all use classic models as the backbone. In our future work, we will demonstrate the feasibility of our method on the latest models.

References

[1] H. Zhu, N. Lin, H. Leung, R. Leung, and S. Theodoidis, "Target classification from sar imagery based on the pixel grayscale decline by graph convolutional neural network," *IEEE Sensors Letters*, vol. 4, no. 6, pp. 1–4, 2020.

[2] A. Masjedi, M. J. V. Zoej, and Y. Maghsoudi, "Classification of polarimetric sar images based on modeling contextual information and using texture features," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 2, pp. 932–943, 2015.

[3] R. Shang, M. Liu, L. Jiao, J. Feng, Y. Li, and R. Stolkin, "Region-level sar image segmentation based on edge feature and label assistance," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–16, 2022.

[4] R. Luo, J. Xing, L. Chen, Z. Pan, X. Cai, Z. Li, J. Wang, and A. Ford, "Glassboxing deep learning to enhance aircraft detection from sar imagery," *Remote Sensing*, vol. 13, no. 18, p. 3650, 2021.

[5] L. Yun, Z. Lifeng, and Z. Shujun, "A hand gesture recognition method based on multi-feature fusion and template matching," *Procedia Engineering*, vol. 29, pp. 1678–1684, 2012.

[6] S. F. Chevtchenko, R. F. Vale, V. Macario, and F. R. Cordeiro, "A convolutional neural network with feature fusion for real-time hand posture recognition," *Applied Soft Computing*, vol. 73, pp. 748–766, 2018.

[7] J. Karvonen and M. Hallikainen, "Sea ice sar classification based on edge features," in *2009 IEEE International Geoscience and Remote Sensing Symposium*, vol. 3. IEEE, 2009, pp. III–129.

[8] M. Wang, J. Luo, and D. Ming, "Extract ship targets from high spatial resolution remote sensed imagery with shape feature," *Geomatics Inf Sci Wuhan Univ*, vol. 30, no. 8, pp. 685–688, 2005.

[9] W. Li, C. Chen, H. Su, and Q. Du, "Local binary patterns and extreme learning machine for hyperspectral imagery classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 7, pp. 3681–3693, 2015.

[10] F. Wu, C. Wang, S. Jiang, H. Zhang, and B. Zhang, "Classification of vessels in single-pol cosmo-skymed images based on statistical and structural features," *Remote Sensing*, vol. 7, no. 5, pp. 5511–5533, 2015.

[11] H. Lin, S. Song, and J. Yang, "Ship classification based on mshog feature and task-driven dictionary learning with structured incoherent constraints in sar images," *Remote Sensing*, vol. 10, no. 2, p. 190, 2018.

[12] Q. Shi, W. Li, R. Tao, X. Sun, and L. Gao, "Ship classification based on multifeature ensemble with convolutional neural network," *Remote Sensing*, vol. 11, no. 4, p. 419, 2019.

[13] C. Wang, J. Shi, Y. Zhou, X. Yang, Z. Zhou, S. Wei, and X. Zhang, "Semisupervised learning-based sar atr via self-consistent augmentation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, no. 6, pp. 4862–4873, 2020.

[14] Y. Dong, H. Zhang, C. Wang, and Y. Wang, "Fine-grained ship classification based on deep residual learning for high-resolution sar images," *Remote Sensing Letters*, vol. 10, no. 11, pp. 1095–1104, 2019.

[15] H. Zheng, Z. Hu, J. Liu, Y. Huang, and M. Zheng, "Metaboost: A novel heterogeneous dcnns ensemble network with two-stage filtration for sar ship classification," *IEEE Geoscience and Remote Sensing Letters*, 2022.

[16] G. Huang, X. Liu, J. Hui, Z. Wang, and Z. Zhang, "A novel group squeeze excitation sparsely connected convolutional networks for sar target classification," *International Journal of Remote Sensing*, vol. 40, no. 11, pp. 4346–4360, 2019.

[17] T. Zhang and X. Zhang, "Injection of traditional hand-crafted features into modern cnn-based models for sar ship classification: What, why, where, and how," *Remote Sensing*, vol. 13, no. 11, p. 2091, 2021.

[18] Zhang, Tianwen and Zhang, Xiaoling, "Integrate traditional hand-crafted features into modern cnn-based models to further improve sar ship classification accuracy," in *2021 7th Asia-Pacific Conference on Synthetic Aperture Radar (APSAR)*. IEEE, 2021, pp. 1–6.

[19] T. Zhang, X. Zhang, X. Ke, C. Liu, X. Xu, X. Zhan, C. Wang, I. Ahmad, Y. Zhou, D. Pan *et al.*, "Hog-shipclsnet: A novel deep learning network with hog feature fusion for sar ship classification," *IEEE Transactions on Geoscience and Remote Sensing*, 2021.

[20] Y. Li, W. Chen, X. Huang, Z. Gao, S. Li, T. He, and Y. Zhang, "Mfvnet: Deep adaptive fusion network with multiple field-of-views for remote sensing image semantic segmentation," *Sci. China Inf. Sci*, 2022.

[21] Y. Li, Y. Zhou, Y. Zhang, L. Zhong, J. Wang, and J. Chen, "Dkdfn: Domain knowledge-guided deep collaborative fusion network for multimodal unitemporal remote sensing land cover classification," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 186, pp. 170–189, 2022.

[22] J. Tompson, R. Goroshin, A. Jain, Y. LeCun, and C. Bregler, "Efficient object localization using convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 648–656.

[23] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 6, pp. 679–698, 1986.

[24] E. Rosten, R. Porter, and T. Drummond, "Faster and better: A machine learning approach to corner detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 1, pp. 105–119, 2008.

[25] J. G. Daugman, "Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters," *JOSA A*, vol. 2, no. 7, pp. 1160–1169, 1985.

[26] D. A. Clausi and M. E. Jernigan, "Designing gabor filters for optimal texture separability," *Pattern Recognition*, vol. 33, no. 11, pp. 1835–1849, 2000.

[27] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971–987, 2002.

[28] D. Sun, A. Yao, A. Zhou, and H. Zhao, "Deeply-supervised knowledge synergy," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6997–7006.

[29] C.-Y. Lee, S. Xie, P. Gallagher, Z. Zhang, and Z. Tu,

"Deeply-supervised nets," in *Artificial intelligence and statistics*. PMLR, 2015, pp. 562–570.

[30] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[31] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *International Conference on Machine Learning*. PMLR, 2015, pp. 2048–2057.

[32] J. Li, C. Qu, and S. Peng, "Ship classification for unbalanced sar dataset based on convolutional neural network," *Journal of Applied Remote Sensing*, vol. 12, no. 3, p. 035010, 2018.

[33] Z. Huang, Z. Pan, and B. Lei, "What, where, and how to transfer in sar target recognition based on deep cnns," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, no. 4, pp. 2324–2336, 2019.

[34] Z. Huang, M. Datcu, Z. Pan, and B. Lei, "Deep sar-net: Learning objects from signals," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 161, pp. 179–193, 2020.

[35] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 25, pp. 1097–1105, 2012.

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

[37] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[38] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4700–4708.

[39] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[40] J. R. Quinlan, "Induction of decision trees," *Machine learning*, vol. 1, no. 1, pp. 81–106, 1986.

[41] T. K. Ho, "The random subspace method for constructing decision forests," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 8, pp. 832–844, 1998.

[42] M. W. Gardner and S. Dorling, "Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences," *Atmospheric environment*, vol. 32, no. 14-15, pp. 2627–2636, 1998.

[43] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1492–1500.

[44] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[45] W. Zhang, J. Li, and X. Qiu, "Sar image super-resolution using deep residual squeezenet," in *Proceedings of the International Conference on Artificial Intelligence, Information Processing and Cloud Computing*, 2019, pp. 1–5.

[46] Z. Ying, C. Xuan, Y. Zhai, B. Sun, J. Li, W. Deng, C. Mai, F. Wang, R. D. Labati, V. Piuri *et al.*, "Tai-sarnet: Deep transferred atrous-inception cnn for small samples sar atr," *Sensors*, vol. 20, no. 6, p. 1724, 2020.

[47] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1251–1258.

[48] J. Wang, H. Xiao, L. Chen, J. Xing, Z. Pan, R. Luo, and X. Cai, "Integrating weighted feature fusion and the spatial attention module with convolutional neural networks for automatic aircraft detection from sar images," *Remote Sensing*, vol. 13, no. 5, p. 910, 2021.

[49] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "Cbam: Convolutional block attention module," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 3–19.