Laplacian Feature Pyramid Network for Object Detection in VHR Optical Remote Sensing Images

Wenhua Zhang, Licheng Jiao, *Fellow, IEEE*, Yuxuan Li, *Student Member, IEEE*, Zhongjian Huang, *Student Member, IEEE*, and Haoran Wang, *Graduate Student Member, IEEE*

***Abstract*— Except for multiscale features, high-frequency features are also crucial for the identification of many objects in object detection for very high resolution optical remote sensing (VHR-ORS) images but have not been considered yet. Due to the fact that the Laplacian pyramid consists of high-frequency information at each level, we propose a Laplacian feature pyramid (FP) network (LFPN) considering both low-frequency features and high-frequency features based on FP structure to improve the object detection performance of VHR-ORS images. FP-based structures are efficient to represent multiscale features. But, in general, FP-based structures, high-frequency features are not specially considered. Such high-frequency features are important to distinguish many ground objects with sufficient details. For example, texture features are critical to distinguish basketball_court and tennis_court. The construction of LFPN consists of a bottom-up pathway, Laplacian pathway, and a fusion pathway, which generate low-frequency pyramid, high-frequency pyramid, and compound pyramid, respectively. The bottom-up pathway follows the computation flow of the backbone convolutional neural networks (CNNs) which is similar to general FP-based structures. The Laplacian pathway extracts the high-frequency features of objects through a trainable Laplacian operator. Finally, the low-frequency and high-frequency FPs are fused to generate the compound pyramid in efficient ways. To evaluate the performance of LFPN, we embed LFPN into both two-stage object detection (T-LFPN) systems and single-stage object detection (S-LFPN) systems to conduct experiments. Experiments on a public challenging ten-class data set NWPU VHR-10 demonstrate the superior performance of LFPN in both T-LFPN and S-LFPN systems and state-of-the-art performance of LFPN-based detectors.**

Manuscript received November 17, 2020; revised February 9, 2021; accepted April 6, 2021. This work was supported in part by the Key Scientific Technological Innovation Research Project by Ministry of Education; in part by the Project supported the Foundation for Innovative Research Groups of the National Natural Science Foundation of China under Grant 61621005; in part by the National Natural Science Foundation of China under Grant U1701267, Grant 61906093, Grant 61573267, and Grant 61906150; in part by the Fund for Foreign Scholars in University Research and Teaching Program's 111 Project under Grant B07048; in part by the Major Research Plan of the National Natural Science Foundation of China under Grant 91438201 and Grant 91438103; in part by the Fundamental Research Funds for the Central Universities under Grant 30919011279 and Grant JBF201905; and in part by the CAAI-Huawei MindSpore Open Fund. *(Corresponding author: Licheng Jiao.)*

The authors are with the Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education, School of Artificial Intelligence, Xidian University, Xi'an 710071, China (e-mail: lchjiao@mail.xidian.edu.cn; zhangwenhua_nuc@163.com).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TGRS.2021.3072488.

Digital Object Identifier 10.1109/TGRS.2021.3072488

***Index Terms*— Convolutional neural networks (CNNs), feature pyramid (FP) networks, Laplacian FP, object detection, very high resolution optical remote sensing (VHR-ORS) images.**

I. Introduction

IN RECENT years, very high resolution optical remote sensing (VHR-ORS) images provide a convenient way to monitor the earth's surface and the analysis of them has attracted increasing attentions [1]–[8]. In this article, we focus on the problem of object detection in VHR-ORS images. Object detection usually can be regarded as a classification problem where the labels of targets and background are assigned. Many machine learning methods are applied and have obtained significant success on some specific issues [1]–[3], [9]–[17]. In these methods, object detection is performed via classifiers, such as support vector machine (SVM) [2], [11], [15], [17], AdaBoost [3], [9], k-nearest neighbors [14], conditional random field [10], and Softmax [1], which are learned to capture the distinct appearance and views of different objects from a set of training data in supervised [1], [3], [9], [10], [12], [13], [15]–[17], semi-supervised [18], or weakly supervised ways [2], [11]. Since in object detection, the objects are usually discriminated in feature space with discriminative features, efficient representation of features is of great importance for object detection systems. In the past few decades, various feature representation methods have been developed for the detection of different types of objects in remote sensing images. The feature extractors can be manually designed or learned by feature learning methods or hierarchical architectures.

The bag-of-words (BoW) [19] is one of the most popular feature representation methods. It represents image regions with histograms computed by sets of visual words which are quantized from collections of local descriptors [20], [21]. The BoW is simple but effective for detectors that result in robustness to viewpoint changes and background blur. Therefore, BoW and its variants have been widely used in remote sensing image detection tasks [11]–[13]. Histogram of oriented gradients (HOG) [22], another popular feature representation method, has been widely used in remote sensing image analysis [4], [23]. In object detection, to represent the objects, HOG estimates the distribution of local gradient intensities and orientations for each region. After

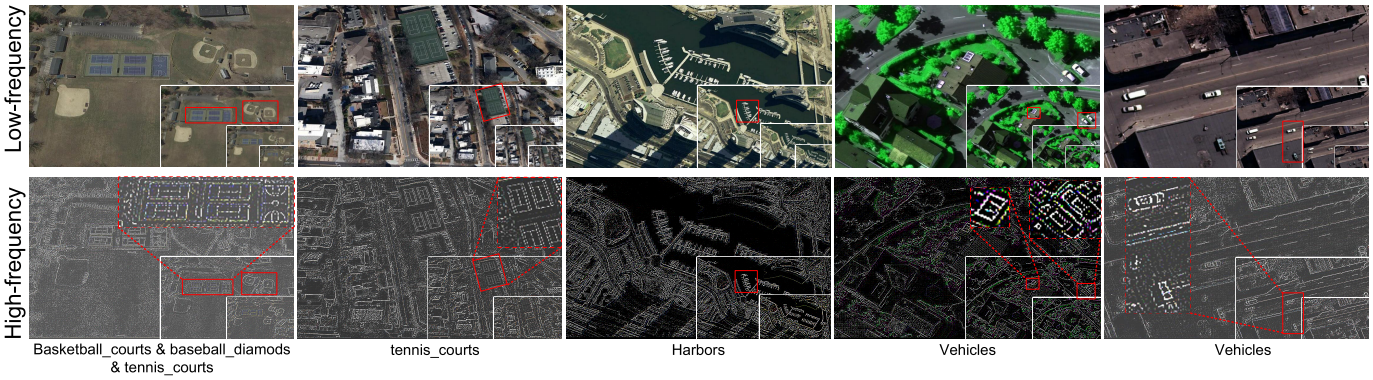

Fig. 1. Pyramids of small-scale and large-sale objects in VHR-ORS images.

that, the edge or shape of objects in the region can be extracted [23], [24]. There are also some variants based on HOG showing impressive performance, such as part-based methods and the sparselets works [4], [25]. Furthermore, some texture features including local binary pattern (LBP) feature, Gabor feature, and shape-based invariant texture feature [26] were also developed for identifying textural objects [9], [10]. Sparse-coding, a learning-based method, was widely used in remote sensing image analysis [14], [27], [28]. The mechanism of sparse coding is to generate sparse code from high dimensional original signals via many structural primitives in low dimensional manifolds [29]. With the excellent representation performance of convolutional neural networks (CNNs), it has been successfully used to object detection in remote sensing images [1], [7]. In [1], a rotation-invariant CNN (RICNN) [1] was proposed to solve the object rotation variation problem in VHR-ORS images. It is based on the architecture of existing CNN, i.e. AlexNet [30], and then a rotation-invariant layer was introduced to deal with rotation variation. Moreover, to obtain the rotation invariance network parameters, a regularization term was designed for the objective function which restrains the similarity between features of training samples before and after rotating. Later, the same group further improved the performance by a local-contextual feature fusion network which can learn local and contextual properties along two independent pathways. The two types of features are then combined in order to form a powerful joint representation [7].

Due to the scale variance of objects, multiscale features are able to efficiently represent useful information which can be extracted via feature pyramid (FP)-based architectures [31]–[38]. FP network (FPN) [39] is a classic FP-based architecture for object detection. FPN adopts a top-down architecture with lateral connections developed for building high-level semantic feature maps at multiple scales that results in robustness to scale variance among different objects. In this article, we focus on detecting objects in VHR-ORS images that are faced with scale variance as well. In such images, except for multiscale features, some high-frequency components with dramatically changed pixels such as textures, lines, and edges in the images are also crucial to distinguish many ground objects. However, due to the hierarchical convolution and downsampling operators in CNN, those components are smoothed out in low-scale feature maps and not specially considered during the following operators in FP-based architectures. Fig. 1 exhibits low-frequency and high-frequency components of different scenarios which are decomposed through Gaussian and Laplacian pyramids. respectively. As shown in Fig. 1, the texture information is the key feature to distinguish different playgrounds, such as basketball_court and tennis_court. The shape, color, and background of them are almost the same. In other types of objects, such components are also of great importance. For example, the stripe of the harbor is the critical feature that distinguishes them from the margin of shade. With the useful features contained in the high-frequency components, we consider the high-frequency features that encode such components to facilitate FP-based architectures. As a consequence, to better improve the performance of object detection in VHR-ORS images by incorporating the high-frequency features, we propose the Laplacian FP network (LFPN) considering both the high-frequency and low-frequency features based on FP structure. LFPN mainly consists of three pathways: the bottom-up pathway, the Laplacian pathway, and the fusion pathway. The bottom-up pathway in LFPN is the forward computation of CNN which is similar to general FP-based structures. Laplacian operator could well generate the high-frequency features of objects [40]–[45]. The Laplacian pathway in LFPN generates the high-frequency features through trainable Laplacian operator on feature maps. The fusion pathway efficiently fuses these two types of features. High-frequency components in an image denote the regions with dramatically changed pixels including those useful components as well as useless ones, such as noise. With the trainable Laplacian operator, useful high-frequency features can be specially learned for accurate object detection. To evaluate the performance of LFPN, we embed LFPN into both two-stage object detection (T-LFPN) and single-stage object detection (S-LFPN) systems and compare them with the corresponding FP-based structure embedded object detection systems in which only the low-frequency features generated by the bottom-up pathway are used. We also compare the proposed LFPN embedded detection systems with several state-of-the-art object detectors designed for remote sensing images. Experimental results on a public ten-class data set NWPU VHR-10 demonstrate the

superiority of LFPN in both T-LFPN and S-LFPN systems, and state-of-the-art performance of LFPN-based systems among compared detectors. LFPN especially benefits in categories of baskerball_court, tennis_court, base_diamond, vehicle, harbor, and bridge which contain sufficient detail features. Moreover, LFPN achieves better performance with almost equivalent running time with FP-based structure and less running time than state-of-the-art detectors.

The contributions of this work are listed as follows.

1) *Laplacian FP Network:* We propose a novel FP-based structure named LFPN via incorporating low-frequency and high-frequency features in order to increase the detecting performance of objects with abundant details.
2) *Laplacian Pathway:* We propose a trainable Laplacian pathway to extract the high-frequency features by using trainable Laplacian operators. The output of the Laplacian pathway is a pyramid with high-frequency feature maps (HF-FMs) of multilevel.
3) *Compound Pathway:* We use two types of fusion ways, including linear and nonlinear fusion, to fuse both low-frequency and high-frequency features. Linear fusion (in linear LFPN, L-LFPN) uses a simple add operator which can improve the performance with almost the same computational complexity. Nonlinear fusion (in nonlinear LFPN, NL-LFPN) introduces a convolutional layer that can fully represent complementary information but with slightly increased computational complexity.

The rest of this article is organized as follows. Before the detailed discussion of the proposed LFPN and for completeness, we first introduce the FP-based structures for object detection in Section II-A and the Laplacian operator and its applications in Section II-B. Section III describes the proposed LFPN. The LFPN-based object detection systems are described in Section IV. Section V reports experimental results. Finally, a conclusion and further work are drawn in Section VI.

II. Related Work

As introduced above, many feature representation methods have been applied to object detection for remote sensing images including manual feature representations [13], [18], [46] and learned representations [1], [7]. Methods mentioned in [1] and [7] follow the two-stage object detection framework with specially designed feature representations for remote sensing objects, including rotation-invariance and local contexture features. In remote sensing, an important property is the scale variance of different ground objects such as vehicles and harbors. The scale variance can be sufficiently represented by FP architectures.

Nowadays, a variety of networks based on FP have emerged due to their excellent multiscale feature representation performance no matter in computer vision field [36]–[39], [47], [48] or in remote sensing field [31]–[35]. This has motivated us to use this architecture to effectively represent multiscale object features in VHR-ORS images. In this section, we first introduce the FP-based architectures which are closely related to the proposed network.

A. Feature Pyramid Based Architectures for Object Detection

Due to the multiscale learning of FP-based architectures, many object detection methods are proposed based on such architectures [31]–[38]. FP network (FPN) [39] is a widely used classical FP-based representation network for object detection. It is composed of bottom-up and top-down pathways and corresponding lateral connections. The bottom-up pathway follows the computation flow of the backbone CNN and the feature maps in each layer are regularized to form a pyramid. The top-down pathway generates multiscale feature maps for object detection. It combines feature maps of adjacent levels via upsampling and element-wise add. To improve the detection performance on small objects, an attention pathway in the deeper layer and an augmented bottom-up pathway were introduced in [31] to make shallow layer information easier to spread. A double FP structure was introduced in [34] in order to obtain more proposals. A parallel FP network was proposed in [38], where the FP was constructed by widening the network width instead of increasing the network depth. In order to improve the performance of rotating object detection, the method mentioned in [32] takes the mask branch in MaskR-CNN [49] for reference. Reinforcement learning through a dueling structure Q network [50] was used in [35]. Moreover, to strengthen the feature representation ability, a U-shape structure was specially considered in [36]. Spatial attention and context information were considered in [38] with the fact that not all features are useful for saliency detection and some even cause interferences.

The above methods have achieved excellent performance on small objects and rotating objects, but texture features are also important to recognize different objects and distinguish targets from the background, especially in remote sensing images, such as different playgrounds, vehicles, and harbor. Such texture information can be represented by high-frequency features. However, among the above methods, high-frequency features are smoothed out during the construction of FP while not specially considered in the following operators. Since the Laplacian operator could well generate the high-frequency features of objects [42], to intensify the high-frequency features of objects, we construct a Laplacian pathway in LFPN through a trainable Laplacian operator.

B. Laplacian Operator and Its Applications

Given an image, the Gaussian pyramid is generated through multilevel down-sample and filtering operators from the image. The forward process of CNN can be taken as a process generating Gaussian pyramid with trainable filtering kernels. Laplacian pyramid [42] is generated by Laplacian operators from a Gaussian pyramid which extracts the lost high-frequency information in each level. Laplacian operator in the current level can be divided into two steps: filtering the upsampled image in higher level in Gaussian pyramid and subtracting filtered image with the image in the current level.

Since the Laplacian operator could well generate high-frequency features of objects, it has been widely used in many image processing problems, such as image blending [42], texture synthesis [43], and edge-aware

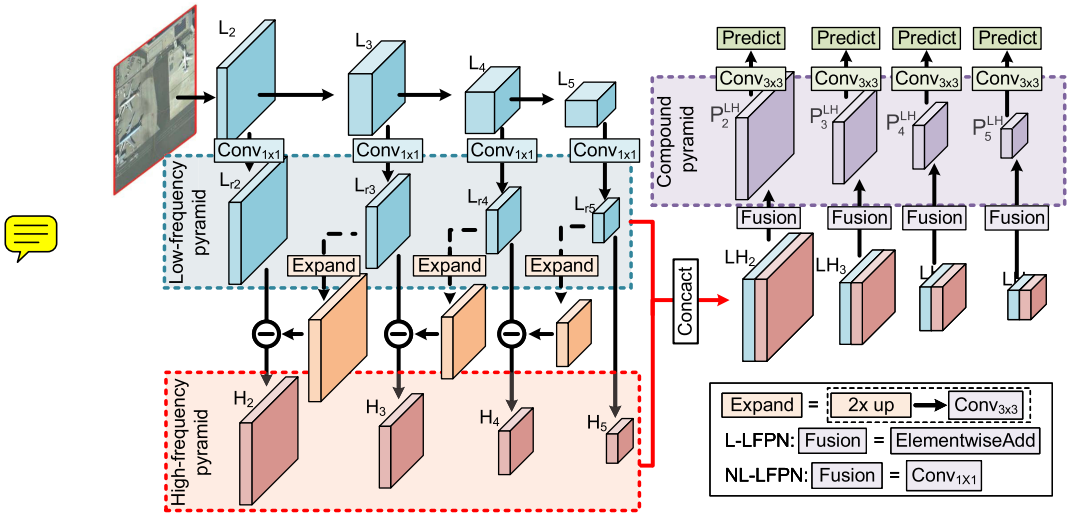

Fig. 2. Structure of the proposed LFPN. The construction of LFPN involves three parts: bottom-up pathway, Laplacian pathway, and fusion pathway.

filtering [44]. Recently, CNNs have been widely used but the convolutional kernels result in loss of image details in the forward CNN architectures. Many methods were proposed to increase the image details by high-frequency information generated via Laplacian operators. For instance, for image super-resolution, in [41] a deep Laplacian pyramid network was proposed by using the Laplacian operator to generate detail information in order to facilitate the upsampled image. It was later improved by an effective two-stage progressive training strategy with different components of high-frequency information in [40]. In [45], a low-frequency segmentation map was first reconstructed via CNN architecture, and then it was refined by adding higher frequency details derived by Laplacian operators for semantic segmentation. As shown in Fig. 1, high-frequency features are also important to identify and distinguish many ground objects in VHR-ORS images but have not been considered in object detection. Therefore, in LFPN, we consider both multiscale and high-frequency feature representations by integrating FP structure and Laplacian operators.

III. Laplacian Feature Pyramid Network

According to the above analysis, to highlight the high-frequency detail features, we propose a novel FP-based feature representation network, LFPN. LFPN takes both low-frequency features and high-frequency features into consideration. The input of LFPN can be images with arbitrary size, and the outputs are feature maps at proportionally increasing levels which are obtained in CNN fashion. This process is executed without obstructing the backbone architectures (e.g., ResNet [51], AlexNet [30], VGG [52]). Following we describe the network structure, optimization, and effectiveness and complexity analysis of LFPN.

A. Architecture of LFPN

As shown in Fig. 2, the construction of LFPN involves three parts: bottom-up pathway, Laplacian pathway, and fusion pathway which result in low-frequency pyramid, high-frequency pyramid, and compound pyramid, respectively. Below we introduce these three parts.

1) Bottom-Up Pathway: Similar to general FP-based architectures, the bottom-up pathway in LFPN is constructed by the feed-forward architecture of the CNNs, which generates multilevel feature maps at decreasing scales. There are often many feature maps of the same size in some layers of CNNs and they compose the network named as *stage*. The output of each *stage* is taken as the reference set of feature maps, which will be adopted to create LFPN for the consideration that the deepest layer of each *stage* deserves the strongest features. In Fig. 2, L_2, L_3, L_4 and L_5 are generated through the feed-forward computation of the backbone CNNs.

Through convolution and pooling operations, the high-frequency features are smoothed in the feed-forward computation of CNNs. Thus, the regularized bottom-up pathway in LFPN can be regarded as a low-frequency pathway. Here, we use ResNets ((ResNet50/ResNet101) [51]) as the backbone network. The feature maps generated by the last residual block in each *stage* are used. We use L_2, L_3, L_4, and L_5 to denote the output of these last residual blocks, and they have strides of $4, 8, 16, 32$ pixels in terms of the input image. The 1st convolutional layer (L_1) is not considered in the pyramid because of its large memory corruption. Since the number of feature maps (dimension) in each layer is not identical in bottom-up pathway from ResNet, the low-frequency feature maps (LF-FMs) $L_{r5} \rightarrow L_{r4} \rightarrow L_{r3} \rightarrow L_{r2}$ are generated by regularizing $L_5 \rightarrow L_4 \rightarrow L_3 \rightarrow L_2$ into the same feature dimension D via attached 1×1 convolutional layer which is formulated as follows:

$$\begin{aligned} L_{r2} &= L_2 \\ L_{ri} &= \mathrm{Conv}_{1\times1}(L_i), \quad i = 3, 4, 5 \end{aligned} \tag{1}$$

where $\mathrm{Conv}_{1\times1}(\cdot)$ denotes the 1×1 neural convolution operator. In this work, we set $D = 256$ which results in 256 channels in each extra convolutional layer. In the bottom-up pathway, the feature maps in higher layers contain proximate global semantic information and those in lower layers contains more

image details. More details are lost in higher layers due to hierarchical downsampling and nonlinear filtering. Then the low-frequency pyramid is generated as shown in Fig. 2.

Since high-frequency features are very important to identify many ground objects in remote sensing images. Different from general FP-based structures that directly use multiscale feature maps from the bottom-up pathway for subsequent operations, LFPN generates high-frequency features via a Laplacian pathway.

2) Laplacian Pathway: In Fig. 2, $H_5 \rightarrow H_4 \rightarrow H_3 \rightarrow H_2$ donates the high-frequency pyramid in LFPN generated by the Laplacian pathway. The Laplacian pyramid provides the high-frequency features of objects. Therefore, here we construct a high-frequency pyramid by trainable Laplacian operator using the low-frequency pyramid. Laplacian pyramid is constructed by the difference between feature maps and filtered upsampled feature maps of higher level. The detailed composition process of Laplacian pathway is listed as follows:

$$\begin{aligned} H_5 &= L_{r5} \\ H_i &= L_{ri} - \mathrm{Conv}_{3\times3}(2 \times up(L_{ri+1})), \quad i = 2, 3, 4 \end{aligned} \tag{2}$$

where $2 \times up(\cdot)$ denotes the 2 times upsampling operator (to improve efficiency, we use the same upsampling operation as in FPN [39]) and $\mathrm{Conv}_{3\times3}$ denotes 3×3 neural convolution operator. From the architecture perspective, the feature maps in regularized L_{r2}, L_{r3}, L_{r4}, and L_{r5} are first upsampled and then followed by a 3×3 convolution operator with trainable filtering kernels. L_{r5} contains the rich semantic features which is directly used as the top level of the high-frequency FP. Then the rest levels are obtained by subtracting corresponding upsampled feature maps from L_{r2}, L_{r3}, and L_{r4}. Laplacian pathway generates multilevel feature maps that contain the smoothed detail information during the construction of low-frequency pyramids. Such information provides edges, lines, textures, etc. of objects which are the key feature for many ground objects in remote sensing images. Moreover, the detailed information can facilitate the object detection system to better distinguish targets from the background.

The low-frequency pyramid provides the general overview of the input image and the high-frequency pyramid contains sufficient image details. It is crucial to efficiently use the two types of information. Therefore, we propose a fusion pathway to fuse the two types of feature maps.

3) Fusion Pathway: In Fig. 2, $P_5^{\mathrm{LH}} \rightarrow P_4^{\mathrm{LH}} \rightarrow P_3^{\mathrm{LH}} \rightarrow P_2^{\mathrm{LH}}$ donates the compound FP generated by the fusion pathway. The fusion pathway is designed to fuse low-frequency features and high-frequency features in an efficient way. Here, we provide two fusion ways, linear (L-LFPN) and nonlinear (NL-LFPN). For linear fusion, the feature maps in the fusion pathway are generated by element-wise add of corresponding feature maps from bottom-up and Laplacian pathways. The linear fusion is simple but efficiently fuses the two types of feature maps. However, it cannot sufficiently represent the relationship between the low-frequency and HF-FMs. Nonlinear fusion uses a convolutional layer with 1×1 trainable convolution kernels

$$\begin{aligned} P_5^{\mathrm{LH}} &= \mathrm{Conv}_{1\times1}(\mathrm{LH}_5), \ \mathrm{LH}_5 = H_5 \\ P_i^{\mathrm{LH}} &= \mathrm{Conv}_{1\times1}(\mathrm{LH}_i), \ \mathrm{LH}_i = \mathrm{Concat}(L_{ri}, H_i), \quad i = 2, 3, 4 \end{aligned} \tag{3}$$

where $\mathrm{Concat}(\cdot)$ denotes the concat operator that combines two tensors into one along the channel direction. Here, we set the output dimension of the convolutional layer as 256. Since the parameters in the convolutional layer are trainable according to both input data and objective function, the compound feature maps can be adaptively learned to fuse useful features from low-frequency and high-frequency pyramids. This will result in an increase of representation capability for remote sensing objects.

From the architecture perspective, feature maps in the compound pyramid are named as P_2^{LH}, P_3^{LH}, P_4^{LH}, and P_5^{LH}, which correspond to L_2, L_3, L_4, and L_5, respectively with the same spatial size. The whole forward process of LFPN is summarized in Fig. 2 where the input image propagates through the three pathways and a compound FP that contains multiscale as well as both LF-FMs and HF-FMs are generated. Then the FP is fed into object detection systems. LFPN not only inherits the multiscale representation learning of FP-based architectures but also especially considers the high-frequency information which is critical for detecting many objects in VHR-ORS images. The optimization follows the training process of general neural networks. With the forward process, it is also important to derive the backward process to train the network parameters.

B. Optimization

The architecture indicates the forward pathways for information flow with trainable convolutional kernels. To train the convolutional kernels, the gradients of output layer should back-propagate through the inverse pathways to compute the gradients of each layer. First, the gradients of the output of fusion pathway from ΔP_2^{LH} to ΔP_5^{LH} are computed from the prediction layer. Then the gradients backpropagate through the fusion pathway. For L-LFPN, gradients of both LF-FMs and HF-FMs are identical to that of compound feature maps: $\Delta H_i = \Delta L_{ri} = \Delta P_i^{\mathrm{LH}}, \quad i = 2, 3, 4, 5$. For NL-LFPN, the gradients should back-propagate through the convolutional layer

$$\begin{aligned} \Delta H_5 = \Delta \mathrm{LH}_5 &= \mathrm{deCon}_{1\times1}(\Delta P_5^{\mathrm{LH}}) \\ \left\{\Delta H_i, \Delta L_{ri}^P\right\} &= \mathrm{deCon}_{1\times1}(\Delta P_i^{\mathrm{LH}}), \quad i = 2, 3, 4 \end{aligned} \tag{4}$$

where $\mathrm{deCon}_{1\times1}()$ denotes the neural deconvolution operator which is inverse process of the 1×1 neural convolution. Here, ΔL_{ri}^P denotes the gradient from the fusion pathway. The gradient of L_{ri} should also contain the gradient back-propagated from the Laplacian pathway, i.e., ΔH_i

$$\begin{aligned} \Delta L_{ri}^H &= \Delta H_i - 2 \times \mathrm{down}(\mathrm{deCon}_{3\times3}(\Delta H_{i-1})), \quad i = 3, 4, 5 \\ \Delta L_{r2}^H &= \Delta H_2 \end{aligned} \tag{5}$$

where $2 \times \mathrm{down}(\cdot)$ denotes the two times downsampling operator. Then the gradient of the output of bottom-up pathway

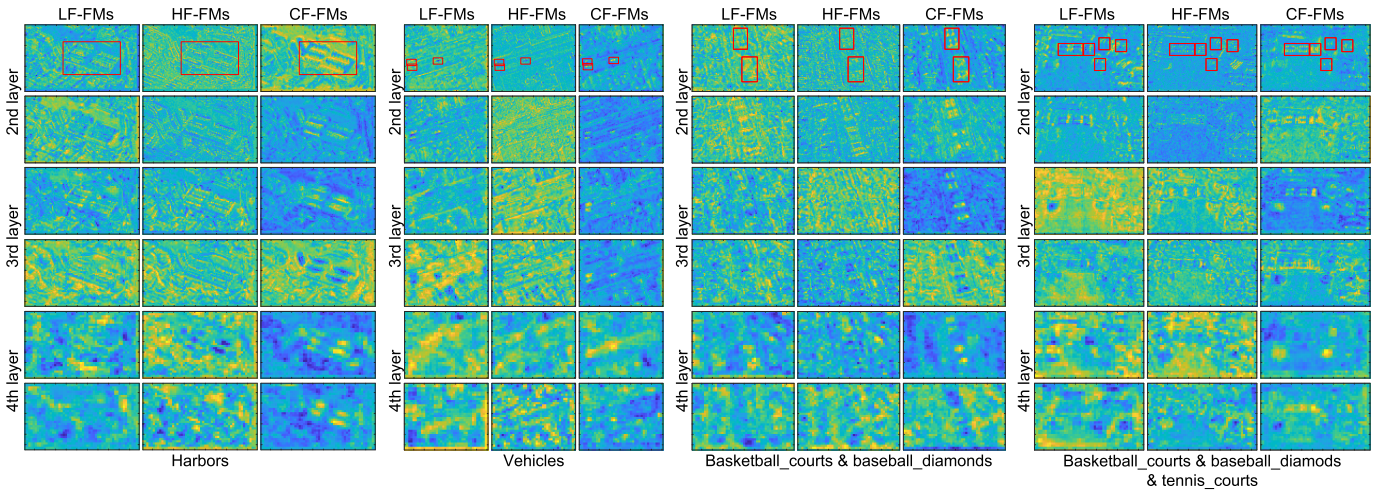

Fig. 3. Visualization of LF-FMs, HF-FMs, and CF-FMs of different layers. Sounded regions with red boxes are the locations of targets.

can be computed

$$\Delta L_{ri} = \Delta L_{ri}^{P} + \Delta L_{ri}^{H}, \quad i = 2, 3, 4, 5. \tag{6}$$

Finally, the gradient of each layer in CNN (L_2, L_3, L_4, and L_5) can be generated

$$\begin{aligned} \Delta L_5 &= \text{deCon}_{1\times1}(\Delta L_{r5}) \\ \Delta L_i &= \text{deCon}_{1\times1}(\Delta L_{ri}) + \text{Back}(\Delta L_{i+1}), \quad i = 2, 3, 4 \end{aligned} \tag{7}$$

where Back() denotes the backpropagation operator in CNN. With the gradients in each layer, the update gradient of convolutional kernels can be computed by derivation operator.

C. Analysis

1) Effectiveness Analysis: Fig. 3 shows some feature maps of different layers in different pathways, including those in 2nd, 3rd, and 4th layers of low-frequency (LF-FMs), high-frequency (HF-FMs), and compound-frequency feature maps (CF-FMs) pyramids. Four types of scenarios shown in Fig. 1 are exhibited. The feature maps in Fig. 1 are extracted by the manually designed Laplacian pyramid which shows all the high-frequency components in images, such as textures, lines, edges, and even noise. While in Fig. 3, the compound features maps show the learned features, especially for the targets. The 2nd layer shows more details of objects but LF-FMs are not able to fully represent the details of objects. As shown in Fig. 3, it is difficult to recognize harbors and vehicles and to distinguish different playgrounds in the same scenario. For example, in the LF-FM, the features of the harbor and margin of shade show a similar appearance. HF-FMs provide complementary information. Therefore, in the compound feature maps, the objects show specific features. For example, the harbor and vehicles in the compound feature map show recognizable appearance. While in Fig. 1, all the high-frequency components are highlighted without considering the object detection problem. Higher layers are more close to the discriminative layer so that the feature maps are semantically related to objects. Compared with LF-FMs, high-frequency facilitated compound feature maps are able to highlight the targets as well as distinguish different targets. For example, in Fig. 3, harbors and vehicles can be highlighted in higher layer feature maps and different playgrounds can be highlighted in different feature maps. By comparing Figs. 1 and 3, the special feature learning for object detection with trainable Laplacian operator and fusion pathway can be demonstrated.

2) Complexity Analysis: LFPN inherits the multiscale feature representation from the FP structure and introduces the Laplacian pathway to fully consider the smoothed image details. This will increase the computational time. For CNN in each layer, the computational complexity of convolution operator is $O(N_{ker}^2 \times N_{in} \times N_{\text{out}} \times N_{\text{wid}} \times N_{\text{hei}})$. N_{ker}, N_{in}, N_{out}, N_{wid}, and N_{hei}, respectively, denote the kernel size, number of input feature maps, number of output feature maps, width, and height of feature maps. Laplacian operator is additionally introduced in LFPN and the computational complexity of it is $O(N_{ker}^2 \times N_{\text{wid}} \times N_{\text{hei}} \times N_{\text{out}})$. The computational time of the convolution operator is N_{in} times that of the Laplacian operator. With numerous feature maps in each layer, the increased computational time by Laplacian operator can be negligible. Then in the fusion pathway, for linear fusion, the computational complexity is $O(N_{\text{wid}} \times N_{\text{Hei}} \times N_{\text{out}})$ which is much smaller than that of convolution and Laplacian operators. For nonlinear fusion, the computational complexity is $O(N_{\text{wid}} \times N_{\text{hei}} \times N_{\text{out}}^2)$. Because the kernel size in the nonlinear fusion is 1, the computational time is several times lower than that of the convolution operator in CNN. The increased time is obvious in training due to numerous forward and backward processes. But for the prediction process, only one forward process is implemented and the increased computational time is not that large. The comparison of the computational time is implemented in Section V.

IV. LFPN for Object Detection

LFPN is a hierarchical feature learning architecture focusing on representing remote sensing objects of variant scales with discriminable low-frequency features and high-frequency

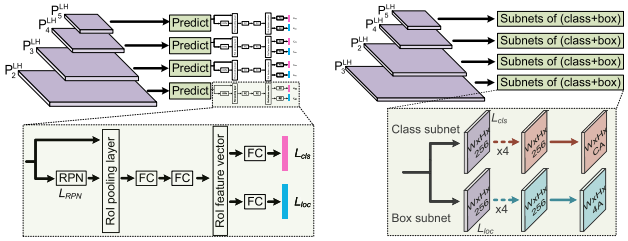

Fig. 4. Overview of the LFPN-based object detection systems. (a) T-LFPN (b) S-LFPN.

features. Then we embed it into object detection systems. The object detection process can be roughly divided into three steps: proposal generation, proposal feature extraction, and discrimination. The current popular object detection systems include two types according to whether it is necessary to train a specialized proposal generation network: two-stage object detection system and single-stage object detection system. The two-stage object detection system needs to specially train the object proposal generation network while the single-stage object detection system needs not. In order to better test the generalization performance of the proposed network, we embed LFPN into both systems (T-LFPN and S-LFPN) to accomplish the object detection in VHR-ORS images.

A. LFPN for Faster R-CNN

Faster R-CNN [53] and Mask R-CNN [54] are two typical two-stage object detection systems. Here, we introduce the LFPN-based system with the sample of Faster R-CNN. Fig. 4(a) shows the structure of the LFPN for two-stage object detection system (T-LFPN). The feature maps in the output pyramid of LFPN are fed into the proposal generation network and detection network to generate the final detection result. Since the region proposal network (RPN)-based proposal generation network [53] embedded in the detection system has achieved excellent performance in many object detection systems [39], [47], [53] as well as those for remote sensing images [7], we use RPN as the proposal generation network in T-LFPN. Next, we mainly introduce the following two aspects: RPN and detection network in T-LFPN.

1) Proposal Generation Network (RPN): RPN is a CNN-based structure used for proposal generation. For a given scene with an arbitrary size, it generates a set of scored rectangular proposals via a small network sliding over the feature maps produced by the last shared convolutional layer (single-scale feature map). RPN generates K proposals for each sliding window. Moreover, RPN generates a binary label indicating the presence/absence of an object for each proposal/anchor. RPN is trained end-to-end by back-propagation and stochastic gradient descent (SGD) [55].

In our work, we adjust the RPN by replacing the single-scale feature map with multiscale feature maps generated by the output pyramid of LFPN. We use an architecture containing a 3×3 convolutional layer and two sibling 1×1 convolutional layers for each level in the FP and anchors of a single scale to each level. We define the areas of those anchors as $\{32^2, 64^2, 128^2, 256^2, 512^2\}$ pixels on P_2^{LH}, P_3^{LH}, P_4^{LH}, P_5^{LH}, and an additionally introduced P_6^{LH}, respectively. The multiple aspect ratios are $\{0.5, 1, 2\}$ for anchors at each level.

The training labels of the anchors are assigned based on their Intersection-over-Union (IoU) ratios compared with ground-truth bounding boxes. If an anchor has the highest IoU with a given ground-truth box or it achieves an IoU value over 0.7, then it is assigned a positive label. If for all ground-truth boxes, it achieves IoU value lower than 0.3, then the negative label is assigned. We do not use the scales of ground-truth boxes explicitly to assign them to the pyramid levels. Instead, the ground-truth boxes have been assigned to pyramid levels which are associated with anchors.

2) Detection Network: After RPN, the attention regions (ARs) on LFPN are obtained. Then, we send the ARs to the region of interest (RoI) pooling layer and send the features after pooling to two fully connected layers to obtain the RoI feature vectors. The RoI feature vectors are then fed into two branches: the classification branch and the bounding box regression branch to accomplish the final detection. The classification branch consists of a fully connected layer and a Softmax classification layer. The bounding box regression branch consists of a fully connected layer and a Smooth$_{L1}$ [53] regression layer.

In addition to being embedded in the two-stage object detection system, LFPN can be also embedded into the single-stage object detection system.

B. LFPN for Retinanet

Retinanet [47] is a typical single-stage object detection framework. An LFPN-based single-stage object detection system (S-LFPN) is an integrated network based on Retinanet which consists of embedded LFPN and two task-specific subnets as shown in Fig. 4(b). The first subnet is used to classify the object from the output of the LFPN. Then the second subnet regresses the bounding box by convolution. The following paragraphs in this subsection mainly introduce the settings of anchors, the classification subnetwork (class subnet), and the regression subnetwork (box subnet).

1) Settings of Anchors: The anchors are with areas from 32^2 to 512^2 on pyramid levels from P_2^{LF} to P_6^{LH}, respectively. At each pyramid level, anchors at three aspect ratios $\{0.5, 1, 2\}$ are used in this article. For denser scale coverage than that in [53], at each level, anchors with the sizes $\{2^0, 2^{1/3}, 2^{2/3}\}$ of the original set of 3 aspect ratio anchors are added. For each anchor, a one-hot vector for classifying objects with a length of C, where C is the number of classes, and a 4-D vector of box regression are assigned. We use the same rule of assignment to that in RPN but it is modified with adjusted thresholds for multiclass detection. Specifically, anchors are assigned according to the IoU. If an anchor's IoU is over 0.5, then it is assigned to the ground-truth object box. It is assigned to the background if its IoU is in $[0, 0.4)$. Then the offsets between anchors and their object boxes are computed to generate the box regression targets. If an object box is not assigned, it will be omitted.

2) Class Subnet: At each spatial position, the class subnet predicts the probability of the corresponding object for each anchor (among A anchors) and object class (C). This subnet is a small full convolutional network (FCN) attached to each LFPN level and the parameters are shared by subnets across all levels. With a D channel input feature map given a pyramid level, four 3×3 convolutional layers are constructed in the subnet, each with D filters and followed by the ReLU activation. Then it is followed by a 3×3 convolutional layer with $D \times A$ filters. Finally, the output is computed via $D \times A$ binary predictions per spatial location with the activation of Sigmoid. We use $A = 9$ in our experiments.

3) Box Subnet: In addition to the subnet of the object classification, another small FCN is attached in each pyramid level in order to offset the variance between an anchor box and a ground-truth object nearby, if there is one. The attached subnet is designed to be identical to the subnet of object classification, in addition to the difference that at each spatial location, the attached subnet ends at the $4 \times A$ linear outputs, as Fig. 4(b) shows. In the attached subnet, those $4 \times A$ outputs generate the relative variance between the anchor box and the ground-truth box.

V. Experimental Verification

A. Data Set

To verify the performance of the proposed LFPN, we evaluate the LFPN-based object detection systems and compared methods on a widely used data set composed of VHR-ORS images with different scenes: NWPU VHR-10 [1], [18]. There are ten classes of objects for detection in the data set including vehicle, bridge, harbor, ground_track_field, basketball_court, tennis_court, baseball_diamond, storage_tank, ship, and airplane. These objects exist in 650 images composing the positive image set where at least one object in each image. Meanwhile, there are also 150 negative images that contain no objects belonging to the ten classes. The negative images could be used as negative samples for semi-supervised learning systems [2], [18] which are not used in this work. The positive image set is used in our work. The ground-truth is hand-crafted annotated positive images with bounding boxes around each object belonging to the ten classes. In our experiments, the training data set and the test data set are divided exactly the same as in [7]. All experiments are conducted on a computer with a 2.4 GHz Intel Xeon CPU, 64GB RAM, and an NVIDA Titan X GPU.

B. Evaluation Metrics

In this article, we use the widely used metrics, i.e., the precision-recall curve (PRC), average precision (AP), and mean AP (mAP) to evaluate the performance of object detection systems [1], [2], [18].

1) Precision-Recall Curve: The PRC is plotted according to the Precision and Recall of a result. The Precision is computed as the proportion of true positive detections. The Recall is defined as the proportion of correctly identified positives. They are formulated as follows:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{8}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{9}$$

where TP, FP, and FN denote the number of true-positives, the number of false-positives, and the number of false-negatives, respectively. In addition, true-negatives are donated as TN.

For a detection result, if the area overlap ratio (IoU) between the predicted bounding box and the ground-truth bounding box is over 0.5, then it is considered to be a true positive. Otherwise, it is assigned as a false-positive. In addition, if several detections overlap with the same ground-truth bounding box simultaneously, only one is considered as a true-positive, and others are considered as false-positives.

2) Average Precision and Mean Average Precision: The AP is computed as the integral of Precision values over all Recall values which is the area under the PRC. Then the mAP metric measures the average value of AP over all categories

$$\text{mAP} = \frac{1}{C} \sum_{c=1}^{C} \text{AP}_c \tag{10}$$

where c represents the category index, and C is the number of categories.

C. Effectiveness Verification of High-Frequency Features

We embed LFPN into both a two-stage object detection system (Faster R-CNN and Mask R-CNN) and a single-stage object detection system (Retinanet) to verify the performance of LFPN embedded detectors. First, we compare LFPN with the classic and widely used FP-based structure, FPN [39] which only uses the low-frequency features (generated by the bottom-up pathway) to demonstrate the effectiveness of feature learning by integrating high-frequency features. The backbone networks are models pre-trained on ImageNet. The learning rate is set 0.005 at the beginning, decays by 0.1 every 10k iterations, and the maximum iterations are 30k. Except for parameters in the construction of LFPN, the other parameter settings for LFPN keep the same with FPN as defined in the corresponding systems for a fair comparison.

1) LFPN Versus FPN in Two-Stage Object Detection Systems: In order to evaluate the performance of LFPN in a two-stage object detection system, we embed LFPN into two classic two-stage object detection systems, including Faster-RCNN [53] and Mask R-CNN [54] to conduct experiments. Both of them use two types of residual structures, i.e. ResNet50 (R50) and ResNet101 (R101) as the backbone networks. Then, eight detectors are formed which are L-LFPN with Faster R-CNN and R50 (L-LFPN-FR50), NL-LFPN with Faster R-CNN and R50 (NL-LFPN-FR50), L-LFPN with Faster R-CNN and R101 (L-LFPN-FR101), NL-LFPN with Faster R-CNN and R101 (NL-LFPN-FR101), L-LFPN with Mask R-CNN and R50 (L-LFPN-MR50), NL-LFPN with Mask R-CNN and R50 (NL-LFPN-MR50), L-LFPN with Mask R-CNN and R101 (L-LFPN-MR101), and NL-LFPN with Mask R-CNN and R101 (NL-LFPN-MR101). We compare these eight detectors with corresponding compared methods: FPN with Faster R-CNN and R50 (FPN-FR50), FPN with Faster R-CNN and R101 (FPN-FR101), FPN with Mask

TABLE I

PERFORMANCE COMPARISON IN TERMS OF AP AND MAP BETWEEN FPN AND LFPN ON DATA SET NWPU VHR-10 [1] IN T-LFPN SYSTEMS

	mAP	airplane	ship	storage_ tank	baseball_ diamond	tennis_ court	basketball_ court	ground_track_ field	harbor	bridge	vehicle
FPN-FR50 (baseline)	0.8867 ±0.0051	1.0000 ±0.0000	0.8723 ±0.0081	0.9091 ±0.0000	0.8849 ±0.0073	0.9065 ±0.0011	0.9089 ±0.0007	0.9998 ±0.0007	0.8332 ±0.0311	0.6861 ±0.0320	0.8658 ±0.0353
L-LFPN-FR50	**0.9027 ±0.0059**	1.0000 ±0.0000	0.8910 ±0.0064	0.9091 ±0.0000	0.9220 ±0.0325	0.9127 ±0.0195	0.9552 ±0.0423	1.0000 ±0.0000	0.8710 ±0.0123	0.6763 ±0.0351	0.8901 ±0.0047
NL-LFPN-FR50	**0.9187 ±0.0027**	1.0000 ±0.0000	0.8941 ±0.0052	0.9091 ±0.0000	0.9653 ±0.0043	0.9362 ±0.0040	0.9875 ±0.0005	1.0000 ±0.0000	0.8738 ±0.0032	0.7299 ±0.0018	0.8912 ±0.0035
FPN-FR101 (baseline)	0.8969 ±0.0037	1.0000 ±0.0000	0.8794 ±0.0143	0.9091 ±0.0000	0.8810 ±0.0099	0.9070 ±0.0015	0.9077 ±0.0044	1.0000 ±0.0000	0.8900 ±0.0266	0.7081 ±0.0103	0.8868 ±0.0036
L-LFPN-FR101	**0.9054 ±0.0033**	0.9988 ±0.0022	0.8826 ±0.0082	0.9091 ±0.0000	0.9102 ±0.0194	0.9074 ±0.0024	0.9647 ±0.0387	1.0000 ±0.0000	0.8915 ±0.0135	0.6944 ±0.0221	0.8954 ±0.0056
NL-LFPN-FR101	**0.9252 ±0.0022**	1.0000 ±0.0000	0.8947 ±0.0023	0.9091 ±0.0000	0.9669 ±0.0017	0.9461 ±0.0037	0.9945 ±0.0035	1.0000 ±0.0000	0.8980 ±0.0078	0.7449 ±0.0057	0.8973 ±0.0032
FPN-MR50 (baseline)	0.8940 ±0.0071	1.0000 ±0.0000	0.8843 ±0.0144	0.9090 ±0.0005	0.8931 ±0.0061	0.9053 ±0.0025	0.9086 ±0.0001	0.9989 ±0.0035	0.8486 ±0.0184	0.7362 ±0.0445	0.8566 ±0.0427
L-LFPN-MR50	**0.9056 ±0.0050**	0.9998 ±0.0004	0.8907 ±0.0080	0.9089 ±0.0008	0.9059 ±0.0041	0.9239 ±0.0336	0.9385 ±0.0380	1.0000 ±0.0000	0.8754 ±0.0137	0.7196 ±0.0347	0.8935 ±0.0031
NL-LFPN-MR50	**0.9264 ±0.0025**	1.0000 ±0.0000	0.8972 ±0.0038	0.9091 ±0.0000	0.9583 ±0.0021	0.9539 ±0.0021	0.9836 ±0.0102	1.0000 ±0.0000	0.8994 ±0.0022	0.7661 ±0.0060	0.8962 ±0.0012
FPN-MR101 (baseline)	0.8972 ±0.0090	1.0000 ±0.0000	0.8909 ±0.0113	0.9091 ±0.0000	0.8873 ±0.0096	0.9070 ±0.0022	0.9062 ±0.0061	0.9898 ±0.0324	0.8728 ±0.0269	0.7293 ±0.0528	0.8794 ±0.0274
L-LFPN-MR101	**0.9063 ±0.0089**	0.9999 ±0.0003	0.8910 ±0.0108	0.9091 ±0.0000	0.8963 ±0.0071	0.9151 ±0.0240	0.9225 ±0.0351	1.0000 ±0.0000	0.8753 ±0.0138	0.7563 ±0.0610	0.8970 ±0.0030
NL-LFPN-MR101	**0.9323 ±0.0026**	1.0000 ±0.0000	0.8947 ±0.0013	0.9091 ±0.0000	0.9680 ±0.0041	0.9665 ±0.0032	0.9919 ±0.0126	1.0000 ±0.0000	0.9009 ±0.0046	0.7905 ±0.0169	0.9016 ±0.0027

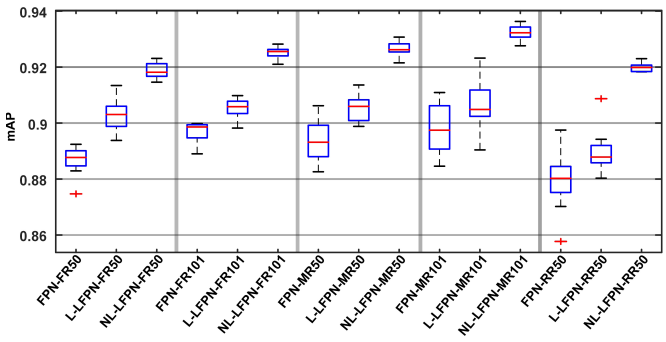

Fig. 5. Performance comparison in terms of stability assessment on mAP between FPN and LFPN on data set NWPU VHR-10 [1].

R-CNN and R50 (FPN-MR50), and FPN with Mask R-CNN and R101 (FPN-MR101) on data set NWPU VHR-10 to verify the performance of LFPN in two-stage object detection systems.

Table I shows the comparison results of FPN and LFPN in two-stage object detection systems. Since we train the systems by stochastic gradient descent where the training batches are randomly selected, the results are not identical in each independent run. Therefore, we run the methods 10 times independently and the mean value and standard deviation (in the way of (mean ± std)) are exhibited in Table I. In general, in these four groups of experiments, the performance of LFPN-based detectors exceeds the performance of corresponding FPN-based detectors (Fig. 5). When the detection framework is Faster R-CNN with R50, the L-LFPN-based detector is generally 1.60% higher than FPN-based detector in terms of mAP and the performance of NL-LFPN-based detector is generally 3.20% higher than FPN-based detector, and the larger performance improvements are in categories of baseball_diamond, basketball_court, harbor and bridge which exceed FPN 8.04%, 7.86%, 4.06%, and 4.38% respectively. When the detection framework is Faster R-CNN with R101, the L-LFPN-based detector is generally 0.85% higher than the FPN-based detector in terms of mAP and the NL-LFPN-based detector is generally 2.83% higher than the FPN-based detector, and the larger performance improvements are in categories of baseball_diamond, tennis_court, basketball_court and bridge which exceed FPN 8.59%, 3.91%, 8.68%, and 3.68% respectively. This demonstrates that the Laplacian high-frequency feature in LFPN not only works on large-scale objects (e.g., baseball_diamond, harbor) but also achieves better performance for small-scale objects with abundant detail information (e.g., vehicle).

When the detection framework is Mask R-CNN with R50, the L-LFPN-based detector is generally 1.16% higher than the FPN-based detector in terms of mAP and the NL-LFPN-based detector is 3.24% higher than the FPN-based detector, and the larger performance improvements are in categories of baseball_diamond, tennis_court, basketball_court, harbor, and vehicle which exceed FPN 6.54%, 4.86%, 7.50%, 5.08%, and 3.96%, respectively. When the detection framework is Mask R-CNN with R101, the L-LFPN-based detector is generally 0.91% higher than the FPN-based detector in terms of mAP and the NL-LFPN-based detector is 3.51% higher than the

TABLE II
PERFORMANCE COMPARISON IN TERMS OF DETECTION SPEED BETWEEN FPN AND LFPN ON DATA SET NWPU VHR-10 [1]

	Average running time per image (second)
FPN-FR50	0.065 ± 0.001
L-LFPN-FR50	0.066 ± 0.001
NL-LFPN-FR50	0.067 ± 0.001
FPN-FR101	0.073 ± 0.001
L-LFPN-FR101	0.074 ± 0.001
NL-LFPN-FR101	0.075 ± 0.001
FPN-MR50	0.065 ± 0.001
L-LFPN-MR50	0.066 ± 0.001
NL-LFPN-MR50	0.067 ± 0.001
FPN-MR101	0.070 ± 0.001
L-LFPN-MR101	0.071 ± 0.001
NL-LFPN-MR101	0.072 ± 0.001
FPN-RR50	0.051 ± 0.001
L-LFPN-RR50	0.052 ± 0.001
NL-LFPN-RR50	0.053 ± 0.001

FPN-based detector, and the larger performance improvements are in categories of baseball_diamond, tennis_court, basketball_court and bridge which exceed FPN 8.07%, 5.95%, 8.57% and 6.12%, respectively. This further demonstrates the overall superior performance of LFPN over FPN in the two-stage object detection system. Moreover, compared to FPN, LFPN almost does not affect detection efficiency as shown in Table II where the average running time (mean ± std) per image is listed.

2) LFPN Versus FPN in Single-Stage Object Detection System: To further verify the generalization performance of LFPN, we embed LFPN into the classic single-stage object detection system, i.e. Retinanet [47] with R50 as the backbone network to conduct experiments on data set NWPU VHR-10 [1]. Then the proposed method is formed as L-LFPN with Retinanet and R50 (L-LFPN-RR50) and NL-LFPN with Retinanet and R50 (NL-LFPN-RR50) and the corresponding compared method is FPN with Retinanet and R50 (FPN-RR50).

Table III shows the comparison results. Overall, L-LFPN-RR50 is higher than FPN-RR50 with 1.00% in terms of mAP and NL-LFPN-RR50 is 4.03% higher than FPN-based detector. Larger performance improvements of NL-LFPN-RR50 are in the categories of baseball_diamond, tennis_court, basketball_court, bridge, and vehicle which exceed FPN 6.13%, 4.42%, 5.72% 12.60%, and 5.42% respectively. This demonstrates the superior performance of LFPN over FPN in a single-stage object detection system. Moreover, LFPN almost does not increase computational burden (Table II).

All the results in terms of mAP over 10 runs are illustrated by box plots in Fig 5. Whether in the two-stage object detection system or the single-stage object detection system, the performance of LFPN is overall better than the performance of FPN. This is mainly due to the introduction of the Laplacian typological high-frequency FP to the learning of texture features. L-LFPN uses a simple element-wise add operator to combine the low-frequency and high-frequency features. It can improve the overall performance without introducing much computational complexity. Due to the trainable fusion layer in NL-LFPN that fully considers the relationship between the two types of features, it significantly outperforms FPN and L-LFPN. Moreover, NL-LFPN decreases the deviation of results which demonstrates its stability in applications. Since NL-LFPN introduces another layer for fusion, the computational complexity has increased slightly. But for prediction, due to that the overall system is composed of numerous layers, the increased computational time can be omitted as listed in Table II.

Compared with two-stage object detection systems, in general, a single-stage achieves lower accuracy. But the improvement of Laplacian FP in a single-stage system is more significant than that in two-stage ones. The Laplacian FP is especially effective for objects with confused appearances in a single-stage system, such as ships and bridges. With the confused appearances, it is difficult for detectors to correctly distinguish the targets and other objects. Moreover, without a regional proposal module, a single-stage system performs poorly in locating them. As a consequence, FPN in two-stage systems significantly outperforms that in single-stage one on objects of ships and bridges. However, with the Laplacian FP, those targets are more recognizable via high-frequency features. Therefore, LFPN in a single-stage system achieves equivalent performance to and even outperforms that in two-stage ones on the two objects.

D. High-Frequency Features on Different Layers

To evaluate the impact of high-frequency features on different layers, we conduct experiments on architectures without, with H_2 layer, (H_2, H_3) layers, and (H_2, H_3, H_4) layers of high-frequency features. Here we use FPN as the FP architecture without high-frequency features. The results are listed in Table IV. In general, better results can be achieved by using more layers of high-frequency features. Nonlinear fusion significantly improves the performance which demonstrates that it can sufficiently fuse the two types of features. For objects with obvious discriminative appearance, such as airplanes, existing methods have achieved perfect performance. The introduced Laplacian FP can keep the perfect performance with both linear and nonlinear fusion. For some objects with confused appearance and simplex features, such as storage tanks, ships, and bridges, the effect of high-frequency features is not obvious. The features even decrease the performance of detecting bridges when they are linearly fused. This is because linear fusion cannot fully consider the importance of different types of features which results in the decrease of effect of low-frequency features, such as color and geometry. But with nonlinear fusion, the high-frequency features can still improve the performance for detecting them even though the improvement is not significant. As the motivation of the proposed LFPN and analysis in the above sections, high-frequency features are specially designed for objects with many details, such as various playgrounds, harbors, and vehicles. Therefore, significant improvement is achieved for such objects.

TABLE III
PERFORMANCE COMPARISON IN TERMS OF AP AND MAP BETWEEN FPN AND LFPN ON DATA SET NWPU VHR-10 [1] IN S-LFPN SYSTEM

	mAP	airplane	ship	storage_ tank	baseball_ diamond	tennis_ court	basketball_ court	ground_track_ field	harbor	bridge	vehicle
FPN-RR50 (baseline)	0.8797 ±0.0111	0.9995 ±0.0015	0.8723 ±0.0183	0.9089 ±0.0007	0.8902 ±0.0125	0.8902 ±0.0106	0.9083 ±0.0233	0.9996 ±0.0009	0.8600 ±0.0162	0.6435 ±0.0562	0.8245 ±0.0438
L-LFPN-RR50	**0.8897 ±0.0078**	0.9999 ±0.0002	0.8780 ±0.0120	0.9089 ±0.0007	0.8985 ±0.0076	0.8909 ±0.0109	0.9176 ±0.0314	0.9998 ±0.0007	0.8723 ±0.0158	0.6788 ±0.0413	0.8528 ±0.0337
NL-LFPN-RR50	**0.9200 ±0.0015**	1.0000 ±0.0000	0.9062 ±0.0043	0.9091 ±0.0000	0.9515 ±0.0021	0.9344 ±0.0060	0.9655 ±0.0040	1.0000 ±0.0000	0.8848 ±0.0045	0.7695 ±0.0094	0.8787 ±0.0023

TABLE IV
PERFORMANCE COMPARISON IN TERMS OF AP AND MAP ON DIFFERENT LAYERS OF L-LFPN AND NL-LFPN ON DATA SET NWPU VHR-10 [1]

	mAP	airplane	ship	storage_ tank	baseball_ diamond	tennis_ court	basketball_ court	ground_track_ field	harbor	bridge	vehicle
FPN-FR50 (baseline)	0.8867 ±0.0051	1.0000 ±0.0000	0.8723 ±0.0081	0.9091 ±0.0000	0.8849 ±0.0073	0.9065 ±0.0011	0.9089 ±0.0007	0.9998 ±0.0007	0.8332 ±0.0311	0.6861 ±0.0320	0.8658 ±0.0353
L-LFPN-FR50 (H_2)	0.8935 ±0.0048	1.0000 ±0.0000	0.8885 ±0.0082	0.9091 ±0.0000	0.9080 ±0.0276	0.9074 ±0.0079	0.9378 ±0.0104	1.0000 ±0.0007	0.8522 ±0.0105	0.6572 ±0.0341	0.8744 ±0.0033
L-LFPN-FR50 (H_2, H_3)	0.8993 ±0.0057	1.0000 ±0.0000	0.8874 ±0.0072	0.9091 ±0.0000	0.9180 ±0.0266	0.9109 ±0.0109	0.9494 ±0.0211	1.0000 ±0.0007	0.8698 ±0.0118	0.6663 ±0.0313	0.8822 ±0.0037
L-LFPN-FR50 (H_2, H_3, H_4)	0.9027 ±0.0059	1.0000 ±0.0000	0.8910 ±0.0064	0.9091 ±0.0000	0.9220 ±0.0325	0.9127 ±0.0195	0.9552 ±0.0423	1.0000 ±0.0000	0.8710 ±0.0123	0.6763 ±0.0351	0.8901 ±0.0047
NL-LFPN-FR50 (H_2)	0.8995 ±0.0021	1.0000 ±0.0000	0.8891 ±0.0053	0.9091 ±0.0000	0.9080 ±0.0045	0.9165 ±0.0036	0.9391 ±0.0003	1.0000 ±0.0009	0.8539 ±0.0032	0.7028 ±0.0012	0.8765 ±0.0038
NL-LFPN-FR50 (H_2, H_3)	0.9160 ±0.0028	1.0000 ±0.0000	0.8923 ±0.0052	0.9091 ±0.0000	0.9635 ±0.0046	0.9266 ±0.0039	0.9856 ±0.0004	1.0000 ±0.0007	0.8677 ±0.0038	0.7245 ±0.0013	0.8902 ±0.0033
NL-LFPN-FR50 (H_2, H_3, H_4)	0.9187 ±0.0027	1.0000 ±0.0000	0.8941 ±0.0052	0.9091 ±0.0000	0.9653 ±0.0043	0.9362 ±0.0040	0.9875 ±0.0005	1.0000 ±0.0000	0.8738 ±0.0032	0.7299 ±0.0018	0.8912 ±0.0035

TABLE V
PERFORMANCE COMPARISON AMONG STATE-OF-THE-ART METHODS IN TERMS OF AP AND MAP ON DATA SET NWPU VHR-10 [1]

	FDDL [47]	SSCBoW [13]	COPD [18]	RICNN [1]	RICAOD [7]	FPN-MR101	L-LFPN-MR101	NL-TLFPN-(MR101)
airplane	0.2915	0.5061	0.6225	0.8835	0.9970	**1.0000**	0.9999	**1.0000**
ship	0.3764	0.5084	0.6887	0.7734	**0.9080**	0.8909	0.8910	0.8947
storage_tank	0.7700	0.3337	0.6371	0.8527	0.9061	**0.9091**	**0.9091**	**0.9091**
baseball_diamond	0.2576	0.4349	0.8327	0.8812	0.9291	0.8873	0.8963	**0.9680**
tennis_court	0.0275	0.0033	0.3208	0.4083	0.9029	0.9070	0.9151	**0.9665**
basketball_court	0.0358	0.1496	0.3625	0.5845	0.8013	0.9062	0.9225	**0.9919**
ground_track_field	0.2010	0.1007	0.8531	0.8673	0.9081	0.9898	**1.0000**	**1.0000**
harbor	0.2539	0.5833	0.5527	0.6860	0.8029	0.8728	0.8753	**0.9009**
bridge	0.2154	0.1249	0.1479	0.6151	0.6853	0.7293	0.7563	**0.7905**
vehicle	0.0447	0.3361	0.4403	0.7110	0.8714	0.8794	0.8970	**0.9016**
mAP	0.2474	0.3081	0.5458	0.7263	0.8712	0.8972	0.9063	**0.9323**

With only one layer of high-frequency features, the improvement is not significant, both for linear and nonlinear fusions. But for some objects, such as different playgrounds and bridges, the two layers of high-frequency features begin to work effectively. Moreover, nonlinear fusion shows more superiority over linear one. But there is an exception where linear fusion achieves better result than nonlinear fusion with two layers of high-frequency features on harbor. Since high-frequency features are more important than low-frequency features for harbors, simple linear fusion can significantly improve the performance. Thus the effect of nonlinear fusion is not that large. From mAP, better performance is achieved with more layers of high-frequency features and nonlinear fusion. This demonstrates the effectiveness of the trainable high-frequency features for object detection in remote sensing images.

E. State-of-the-Art Comparisons

To put the detection performance into perspective, we select one group best performed LFPN detectors and the corresponding FPN-based detector, FPN-MR101, L-LFPN-MR101, and NL-LFPN-MR101, and compare them to several popular object detectors designed for remote sensing images RICAOD [7], RICNN [1], COPD [18], FDDL [46], and SSCBoW [13] on the challenging data set NWPU VHR-10 [1]. Among comparison methods, RICAOD and RICNN use the CNN structure to extract features and follow the two-stage object detection framework. SSCBoW uses BoW method to

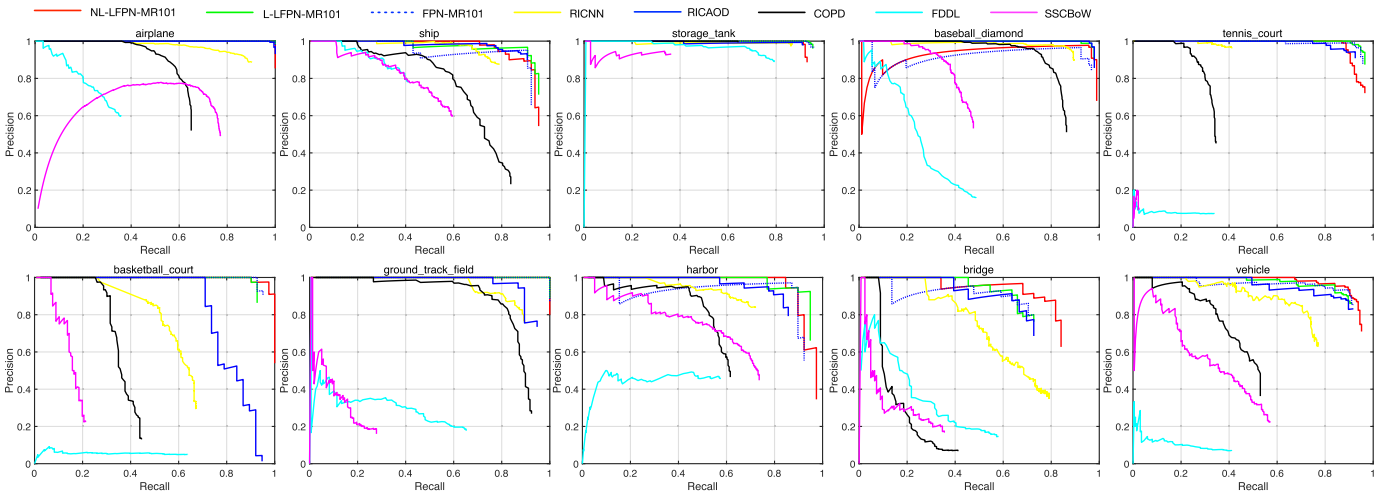

Fig. 6. PRCs of two proposed methods in the correspondence frameworks and other state-of-the-art approaches for all ten classes, respectively.

Fig. 7. Detection results of L-LFPN-FR101 on the test data set. The red boxes donate true positives. The cyan boxes donate false positives. The green boxes donate false negatives.

TABLE VI
PERFORMANCE COMPARISON IN TERMS OF DETECTION SPEED AMONG STATE-OF-THE-ART APPROACHES

	Average running time per image (second)
SSCBoW [13]	38.13
FDDL [47]	6.09
COPD [18]	1.12
RICNN [1]	8.30
RICAOD [7]	2.89

represent features. FDDL uses a sparse coding-based feature representation method and COPD adopts HOG to represent features of objects.

Fig. 6 and Table V show the quantitative comparison results of SSCBoW, FDDL, COPD, RICNN, RICAOD, FPN-MR101, L-LFPN-MR101, and NL-LFPN-MR101 in terms of AP, mAP, and PRCs, respectively. Overall, the proposed methods (L-LFPN-MR101 and NL-LFPN-MR101) outperform all other comparison approaches in terms of mAP. Specifically, L-LFPN-MR101 obtains 65.89%, 59.82%, 36.05%, 18.00%, 3.51%, and 0.91% performance gains in terms of mAP compared with the FDDL [46], SSCBoW [13], COPD [18], RICNN [1], RICAOD [7], and FPN-MR101, respectively. NL-LFPN-MR101 outperforms FDDL [46], SSCBoW [13], COPD [18], RICNN [1], RICAOD [7], and FPN-MR101 with 68.49%, 62.42%, 38.65%, 20.60%, 6.11%, and 3.51%, respectively. This demonstrates the superiority of the proposed LFPN-based object detection systems over compared methods, especially for categories of basketball_court, ground_track_field, harbor, bridge, and vehicle. But for ships, RICAOD has the highest values in terms of AP. This is due to the use of multiangle proposals in RPN in RICAOD and the introduction of local-contextual feature fusion network in RICAOD which lead to robustness to object rotation and appearance ambiguity of objects. Compared with state-of-the-art methods for remote sensing images, FPN benefits from the multiscale feature representation which is critical for scale variance of different types of objects in VHR-ORS images. Therefore, the FPN-based detector shows promising performance on remote sensing images. The proposed LFPN further complements the high-frequency information which is also critical information for many objects. Therefore, LFPN achieves significant improvements over FPN in VHR-ORS images. Fig. 7 shows some detection results on the test data

set with L-LFPN-FR101. As shown in Fig. 7, even with linear fusion, L-LFPN-FR101 detects most of the objects. Table VI summarizes the detection speed of these compared methods where LFPN-based detectors are efficient among state-of-the-art methods.

VI. Conclusion

In this article, we propose an LFPN by incorporating both low-frequency features and high-frequency features based on FP structure to improve the object detection performance of VHR-ORS images. The FP-based structure could well represent multiscale features and benefits in multiscale object detection. High-frequency could well represent the texture features of objects. Such high-frequency features are important to identify many objects in VHR-ORS images but are smoothed out during the forward computation of CNNs. In LFPN, high-frequency features are adaptively learned via trainable convolutional kernels. Then the two types of features are fused in the fusion pathway and two fusion ways including linear and nonlinear are provided. We embed LFPN into both two-stage object detection and single-stage object detection systems on various network backbones. The experiments are conducted to compare LFPN with an FP architecture, FPN in those object detection systems on a widely used public challenging ten-class data set NWPU VHR-10. We also compare the constructed detectors with several popular object detectors designed for remote sensing images to evaluate the advancement of the proposed methods. Experimental results demonstrate that LFPN significantly improves the performance of FPN with almost the same computational efficiency in both two-stage and single-stage object detection systems. Moreover, the proposed methods achieve the state-of-the-art performance among compared detectors.

In future work, focusing on indistinguishable (small objects such as vehicles and ships) and confused objects (such as bridges and roads), we intend to combine the existing network with the attention mechanism to improve the performance of the relatively weak detection categories without decreasing the current performance. Moreover, we will explore more applications with the Laplacian FP based on different baselines.

Acknowledgment

The authors would like to thank J. Han, G. Cheng, P. Zhou, and L. Guo, who generously provided their data set NWPU VHR-10 with ground-truth.

References

[1] G. Cheng, P. Zhou, and J. Han, "Learning rotation-invariant convolutional neural networks for object detection in VHR optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 12, pp. 7405–7415, Dec. 2016.

[2] J. Han, D. Zhang, G. Cheng, L. Guo, and J. Ren, "Object detection in optical remote sensing images based on weakly supervised learning and high-level feature learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 6, pp. 3325–3337, Jun. 2015.

[3] J. Leitloff, S. Hinz, and U. Stilla, "Vehicle detection in very high resolution satellite images of city areas," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 7, pp. 2795–2806, Jul. 2010.

[4] G. Cheng, J. Han, L. Guo, Z. Liu, S. Bu, and J. Ren, "Effective and efficient midlevel visual elements-oriented land-use classification using VHR remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 8, pp. 4238–4249, Aug. 2015.

[5] X. Yao, J. Han, C. Gong, X. Qian, and L. Guo, "Semantic annotation of high-resolution satellite images via weakly supervised learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 6, pp. 3660–3671, Jun. 2016.

[6] Z. Deng, H. Sun, S. Zhou, J. Zhao, L. Lei, and H. Zou, "Multi-scale object detection in remote sensing imagery with convolutional neural networks," *ISPRS J. Photogramm. Remote Sens.*, vol. 145, pp. 3–22, Nov. 2018.

[7] K. Li, G. Cheng, S. Bu, and X. You, "Rotation-insensitive and context-augmented object detection in remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 4, pp. 2337–2348, Apr. 2018.

[8] W. Zhang, L. Jiao, X. Liu, and J. Liu, "Multi-scale feature fusion network for object detection in VHR optical remote sensing images," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2019, pp. 330–333.

[9] Ö. Aytekin, U. Zongur, and U. Halici, "Texture-based airport runway detection," *IEEE Geosci. Remote Sens. Lett.*, vol. 10, no. 3, pp. 471–475, May 2013.

[10] P. Zhong and R. Wang, "A multiple conditional random fields ensemble model for urban area detection in remote sensing optical images," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 12, pp. 3978–3988, Dec. 2007.

[11] D. Zhang, J. Han, G. Cheng, Z. Liu, S. Bu, and L. Guo, "Weakly supervised learning for target detection in remote sensing images," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 4, pp. 701–705, Apr. 2015.

[12] S. Xu, T. Fang, D. Li, and S. Wang, "Object classification of aerial images with bag-of-visual words," *IEEE Geosci. Remote Sens. Lett.*, vol. 7, no. 2, pp. 366–370, Apr. 2010.

[13] H. Sun, X. Sun, H. Wang, Y. Li, and X. Li, "Automatic target detection in high-resolution remote sensing images using spatial sparse coding bag-of-words model," *IEEE Geosci. Remote Sens. Lett.*, vol. 9, no. 1, pp. 109–113, Jan. 2012.

[14] L. Zhang, L. Zhang, D. Tao, and X. Huang, "Sparse transfer manifold embedding for hyperspectral target detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 2, pp. 1030–1043, Feb. 2014.

[15] S. Das, T. T. Mirnalinee, and K. Varghese, "Use of salient features for the design of a multistage framework to extract roads from high-resolution multispectral satellite images," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 10, pp. 3906–3931, Oct. 2011.

[16] F. Bi, B. Zhu, L. Gao, and M. Bian, "A visual search inspired computational model for ship detection in optical satellite images," *IEEE Geosci. Remote Sens. Lett.*, vol. 9, no. 4, pp. 749–753, Jul. 2012.

[17] C. Zhu, H. Zhou, R. Wang, and J. Guo, "A novel hierarchical method of ship detection from spaceborne optical image based on shape and texture features," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 9, pp. 3446–3456, Sep. 2010.

[18] G. Cheng, J. Han, P. Zhou, and L. Guo, "Multi-class geospatial object detection and geographic image classification based on collection of part detectors," *ISPRS J. Photogramm. Remote Sens.*, vol. 98, pp. 119–132, Dec. 2014.

[19] F.-F. Li and P. Perona, "A Bayesian hierarchical model for learning natural scene categories," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2005, pp. 524–531.

[20] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, Nov. 2004.

[21] G.-S. Xia, J. Delon, and Y. Gousseau, "Accurate junction detection and characterization in natural images," *Int. J. Comput. Vis.*, vol. 106, no. 1, pp. 31–56, Jan. 2014.

[22] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, San Diego, CA, USA, Jun. 2005, pp. 886–893.

[23] S. Tuermer, F. Kurz, P. Reinartz, and U. Stilla, "Airborne vehicle detection in dense urban areas using HoG features and disparity maps," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 6, no. 6, pp. 2327–2337, Dec. 2013.

[24] C. Yao and G. Cheng, "Approximative bayes optimality linear discriminant analysis for Chinese handwriting character recognition," *Neurocomputing*, vol. 207, pp. 346–353, Sep. 2016.

[25] G. Cheng, J. Han, L. Guo, and T. Liu, "Learning coarse-to-fine sparselets for efficient object detection and scene classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1173–1181.

[26] G.-S. Xia, J. Delon, and Y. Gousseau, "Shape-based invariant texture indexing," *Int. J. Comput. Vis.*, vol. 88, no. 3, pp. 382–403, Jul. 2010.

[27] Y. Zhong, R. Feng, and L. Zhang, "Non-local sparse unmixing for hyperspectral remote sensing imagery," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 6, pp. 1889–1909, Jun. 2014.
[28] B. Song, P. Li, J. Li, and A. Plaza, "One-class classification of remote sensing images using kernel sparse representation," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 9, no. 4, pp. 1613–1623, Apr. 2016.
[29] W. Zhang, L. Jiao, Y. Li, and J. Liu, "Sparse learning-based correlation filter for robust tracking," *IEEE Trans. Image Process.*, vol. 30, pp. 878–891, 2021.
[30] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.
[31] W. Guo, W. Li, W. Gong, and J. Cui, "Extended feature pyramid network with adaptive scale training strategy and anchors for object detection in aerial images," *Remote Sens.*, vol. 12, no. 5, p. 784, Mar. 2020.
[32] Y. Li, Q. Huang, X. Pei, L. Jiao, and R. Shang, "RADet: Refine feature pyramid network and multi-layer attention network for arbitrary-oriented object detection of remote sensing images," *Remote Sens.*, vol. 12, no. 3, p. 389, Jan. 2020.
[33] C. Chen, W. Gong, Y. Chen, and W. Li, "Object detection in remote sensing images based on a scene-contextual feature pyramid network," *Remote Sens.*, vol. 11, no. 3, p. 339, Feb. 2019.
[34] X. Zhang *et al.*, "Geospatial object detection on high resolution remote sensing imagery based on double multi-scale feature pyramid network," *Remote Sens.*, vol. 11, no. 7, p. 755, Mar. 2019.
[35] K. Fu *et al.*, "A ship rotation detection model in remote sensing images based on feature fusion pyramid network and deep reinforcement learning," *Remote Sens.*, vol. 10, no. 12, p. 1922, Nov. 2018.
[36] Q. Zhao *et al.*, "M2Det: A single-shot object detector based on multi-level feature pyramid network," in *Proc. AAAI*, 2019, pp. 9259–9266.
[37] T. Zhao and X. Wu, "Pyramid feature attention network for saliency detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3085–3094.
[38] S. Kim, H. Kook, J. Sun, M. Kang, and S. Ko, "Parallel feature pyramid network for object detection," in *Proc. ECCV*, 2018, pp. 239–256.
[39] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2117–2125.
[40] Y. Tang, W. Gong, X. Chen, and W. Li, "Deep inception-residual Laplacian pyramid networks for accurate single-image super-resolution," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 5, pp. 1514–1528, May 2020.
[41] W.-S. Lai, J.-B. Huang, N. Ahuja, and M.-H. Yang, "Fast and accurate image super-resolution with deep Laplacian pyramid networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 11, pp. 2599–2613, Nov. 2019.
[42] P. Burt and E. Adelson, "The Laplacian pyramid as a compact image code," *IEEE Trans. Commun.*, vol. COM-31, no. 4, pp. 532–540, Apr. 1983.
[43] D. J. Heeger and J. R. Bergen, "Pyramid-based texture analysis/synthesis," in *Proc. SIGGRAPH*, 1995, pp. 229–238.
[44] S. Paris, S. W. Hasinoff, and J. Kautz, "Local Laplacian filters: Edge-aware image processing with a Laplacian pyramid," *Commun. ACM*, vol. 58, no. 3, pp. 81–91, Feb. 2015.
[45] G. Ghiasi and C. C. Fowlkes, "Laplacian pyramid reconstruction and refinement for semantic segmentation," in *Proc. ECCV*, 2016, pp. 519–534.
[46] J. Han *et al.*, "Efficient, simultaneous detection of multi-class geospatial targets based on visual saliency modeling and discriminative learning of sparse coding," *ISPRS J. Photogramm. Remote Sens.*, vol. 89, pp. 37–48, Mar. 2014.
[47] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.
[48] S.-W. Kim, H.-K. Kook, J.-Y. Sun, M.-C. Kang, and S.-J. Ko, "Parallel feature pyramid network for object detection," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 234–250.
[49] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 386–397, Feb. 2020.
[50] V. Mnih *et al.*, "Playing atari with deep reinforcement learning," 2013, *arXiv:1312.5602*. [Online]. Available: http://arxiv.org/abs/1312.5602
[51] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.
[52] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Represent.*, 2015, pp. 1–14.
[53] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 91–99.
[54] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 2961–2969.
[55] Y. LeCun *et al.*, "Backpropagation applied to handwritten zip code recognition," *Neural Comput.*, vol. 1, no. 4, pp. 541–551, Dec. 1989.

Wenhua Zhang received the B.S. degree in communication engineering from the North University of China, Taiyuan, China, in 2015. She is pursuing the Ph.D. degree with the School of Artificial Intelligence, Xidian University, Xi'an, China.

Her research interests include machine learning and image processing.

Licheng Jiao (Fellow, IEEE) received the B.S. degree from Shanghai Jiaotong University, Shanghai, China, in 1982, and the M.S. and Ph.D. degrees from Xi'an Jiaotong University, Xi'an, China, in 1984 and 1990, respectively.

Since 1992, he has been a Professor with the School of Artificial Intelligence, Xidian University, Xi'an, where he is the Director of Key Laboratory of Intelligent Perception and Image Understanding of the Ministry of Education of China. His research interests include image processing, natural computation, machine learning, and intelligent information processing.

Dr. Jiao is a member of the IEEE Xi'an Section Execution Committee, the Chairman of the Awards and Recognition Committee, the Vice Board Chairperson of the Chinese Association of Artificial Intelligence, a Councilor of the Chinese Institute of Electronics, a committee member of the Chinese Committee of Neural Networks, and an expert of the Academic Degrees Committee of the State Council.

Yuxuan Li (Student Member, IEEE) received the B.S. degree in computer science from Henan University, Kaifeng, China, in 2015. He is pursuing the Ph.D. degree with the School of Artificial Intelligence, Xidian University, Xi'an, China.

His research interests include deep learning and video analysis and processing.

Zhongjian Huang (Student Member, IEEE) received the B.S. degree in intelligent science and technology from Xidian University, Xian, China, in 2018, where he is pursuing the Ph.D. degree in computer science and technology.

He is a member of the Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education, International Research Center for Intelligent Perception and Computation, and Joint International Research Laboratory of Intelligent Perception and Computation, Xidian University. His research interests include video tracking and satellite video analysis.

Haoran Wang (Graduate Student Member, IEEE) received the B.S. degree in electronic information engineering from Xidian University, Xian, China, in 2017, where he is pursuing the Ph.D. degree in circuit and system.

He is a member of the Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education, International Research Center for Intelligent Perception and Computation, and Joint International Research Laboratory of Intelligent Perception and Computation, Xidian University. His research interests include person reidentification and video analysis.