Lateral Feature Enhancement Network for Page Object Detection

Cao Shi, Canhui Xu, Hengyue Bi, Yuanzhi Cheng, Yuteng Li, and Honghong Zhang

***Abstract*— In this article, a lateral feature enhancement (LFE) backbone network is proposed to enrich feature representation effectively for page object detection (POD) across various scales. Our LFE backbone network has three feature enhancement modules. First, feature enhancement of large page object is a bottom-up feature pyramid, enhancing features of large page objects, which convey more important information to readers. Second, the LFE includes a top-down feature pyramid propagating representative semantical features to lower layers and a lateral connection for feature enhancement in each layer. Third, lateral skip connection is designed to retain the original feature details. The stacking strategies of bottom-up, top-down, and lateral connections are beneficial to overall object detection. Visualization of feature indicates that the proposed LFE backbone network enhances global semantic information as well as detailed features of small page objects. Comparative experiments on the two state-of-the-art datasets show that it achieves excellent results with 0.950 mean of AP (mAP) on PubLayNet and 0.892 mAP on POD with more strict metric intersection over union (IoU) = 0.8, respectively. Compared with both computer vision (CV)-based unimodal detectors and multimodal detectors, the proposed LFE network performs excellently. Visual effect experiments compare the performances of CV-based detectors. The results show that our detector outperforms others with strict metric, especially in the detection of small page objects.**

***Index Terms*— Deep convolutional neural network (CNN), document image, feature enhancement, page object detection (POD).**

I. Introduction

PAGE object detection is a crucial preceding step in automatic document analysis and understanding, which aims to classify the segmented regions semantically into tables, figures, formulas, texts, and other page parts, so as to let a machine understand content. Page objects have a large range of size variation. Different from natural scene images, document images have their own distribution. Comparatively, objects, such as figures and tables, belong to large objects. Formula and section title are comparatively small objects. Text blocks occupied majority of the distribution.

For natural scene image objects detection, there exist two categories of detectors to extract visual features, which are one-stage and two-stage detectors [1]. One-stage detectors include You Only Look Once (YOLO) [2], YOLO v4 [3], You Only Look At CoefficienTs ++ (YOLACT++) [4], Single Shot MultiBox Detector (SSD) [5], RetinaNet [6], Single-Shot Object Detector based on Multi-Level Feature Pyramid Network (M2Det) [7], RefineNet [8], and so on. One-stage detectors are suitable for real-time tasks [9], [10], [11], usually without region proposal network (RPN). In contrast with one-stage detectors, two-stage detectors achieve high accuracy by proposing regions for detected objects [12], [13], [14] for classification and localization. Due to the RPN module, two-stage detectors, such as regions with convolutional neural network (CNN) features (R-CNN) [15], Fast R-CNN [16], Faster R-CNN [17], Mask R-CNN [18], and so on, are able to filter out a large number of negative locations, which brings better accuracy but less efficiency when compared with one-stage detectors. According to the requirements of different applications, it suggested that the network can be designed to make a trade-off between accuracy and speed. Two kinds of detectors both need backbone network to extract features, which are generally based on convolutional neural networks (CNNs), such as AlexNet [19], Visual Geometry Group (VGG) [20], residual network (ResNet) [21], and so on. To further utilize features at various scales, a feature pyramid network (FPN) [22] is proposed. In visual detection task, FPN is integrated to backbone, such as ResNet, for extracting region of interest (RoI) features from various levels of feature pyramid. Lower resolution feature map from high level has richer semantic information. Meanwhile, the high-resolution low-level layers contain spatial localization information, since the extracted feature maps are decisive fundaments for network performance [23].

In the field of page object detection (POD), deep learning methods are introduced for different tasks, such as table detection [24], [25], formula detection [26], and various POD [27], [28], [29], [30]. Among these methods, most are two-stage detectors. The POD competition in the International Conference on Document Analysis and Recognition (ICDAR2017) [31], [32] summarizes top seven detectors, five of which are based on two-stage detector Faster R-CNN [17], and one of which is based on one-stage detector SSD [5]. More recent competition on scientific literature parsing (SLP) in ICDAR2021 [33], [34] concludes top nine detectors, which at least has five two-stage detectors, and the rest four teams did not provide their detector information. In competitions POD2017 [32] and SLP [34], backbone networks of two-stage detectors are mostly VGG [20], ResNet [21], or their variations. Apart from CNN-based backbone network, traditional

Manuscript received 29 March 2022; revised 17 July 2022; accepted 13 August 2022. Date of publication 25 August 2022; date of current version 8 September 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 61806107 and Grant 61702135, in part by the Shandong Key Laboratory of Wisdom Mine Information Technology, and in part by the Opening Project of State Key Laboratory of Digital Publishing Technology. The Associate Editor coordinating the review process was Dr. Jingyu Hua. *(Corresponding author: Canhui Xu.)*

The authors are with the School of Information Science and Technology, Qingdao University of Science and Technology, Qingdao 266061, China (e-mail: ccxu09@yeah.net).

Digital Object Identifier 10.1109/TIM.2022.3201546

methods are also integrated into deep learning architecture for feature extraction. For example, Li *et al.* [35] use conditional random field (CRF) to extract spatial feature for CNN, and Younas *et al.* [36] consider traditional computer vision (CV) representations (color, connection, and so on) as inputs for deep learning model.

Compared with the previous work, the main novelties of this article are the following.

1) Present feature enhancement of large page objects according to page object distribution caused by 2-D translations and zooming in/out of page object in the page layout process.
2) Propose lateral feature enhancement (LFE) with the aim to enhance feature representation of small objects as well as large objects. This enhancement is top-down with LFE in each layer, whereas enhancement in (1) is bottom-up. Two kinds of enhancement are designed for page objects detection across various scales. It is noted that the introduction of low-level spatial information to deeper layers can help large objects recognition. Meanwhile, the propagation of high-level semantic signals upsampled to higher resolution low-level feature layers enhances the performance semantically.
3) Design a lateral skip connection from backbone network to feature pyramids to enhance features in multiple scales.

The rest of this article is organized as follows. Related works are introduced in Section II. Our feature enhancement backbone network and two-stage detector are proposed in Section III. Experimental results and discussion are presented at Section IV. The conclusion is given in Section V.

II. Related Works

To deal with scale variation task, input image pyramid methods were initial attempts. As rapid development of deep learning architectures, feature pyramid becomes more practical for object detection. It is well known that the feature pyramid module can be easily fit into deep learning networks. Both one-stage and two-stage detectors apply feature pyramids.

As a typical one-stage detector, SSD [5] constructs feature pyramid by selecting two layers from backbone VGG16 and four layers from stride 2 convolution. Deconvolutional single shot detector (DSSD) [37] uses deconvolution layers from a single layer of backbone ResNet network. Deconvolution layers are to aggregate context and explore high-level semantics for shallow features.

The well-known FPN [22] utilizes lateral connections to fuse feature maps in a top-down manner. Recently, M2Det [7] develops thinned U-shape modules (TUMs) and exploits the decoder layers of each TUM for detecting objects of different scales. Mixture feature pyramid network (MFPN) [38] assembles top-down, bottom-up, and fusing-splitting FPN in parallel manner to enhance small-, large-, and medium-sized object detection, respectively. Path aggregation network (PAnet) [39] adds bottom-up path augmentation upon FPN to boost localization information from lower layers. It is claimed that stacking multiple feature pyramids proposed by neural architecture search (NAS)-FPN [40] increases detection accuracy.

Typically, FPN architectures use top-down and bottom-up to detect objects with various sizes. By stacking feature pyramids in parallel or sequentially, the multiscale FPN can significantly enhance feature representation. Top-down FPN integrates high-level semantic to low-level features for small objects representation. In contrast, bottom-up FPN introduces low-level spatial information to high-level features for enriching large objects description. There are other proposal-based methods focusing on anchor adjustment to deal with scale variation. Connectionist text proposal network (CTPN) [41] develops vertical anchors and connects fine-scale text proposals so as to detect text with various scales in natural images.

In the field of document image analysis, deep learning gains its popularity to detect, segment, and recognize document page objects. Some CNNs work for end-to-end pixel level analysis, whereas others aim to detect and classify regions with bounding boxes. In ICDAR2017 POD competition, almost all the participated teams used deep learning for object detection, including popular SSD, Faster R-CNN-based models aiming to detect tables, mathematical equations, and figures [32]. Also, in ICDAR2021 SLP competition, half of the detectors are based on Mask R-CNN, which generally apply ResNet and FPN as a backbone network [34].

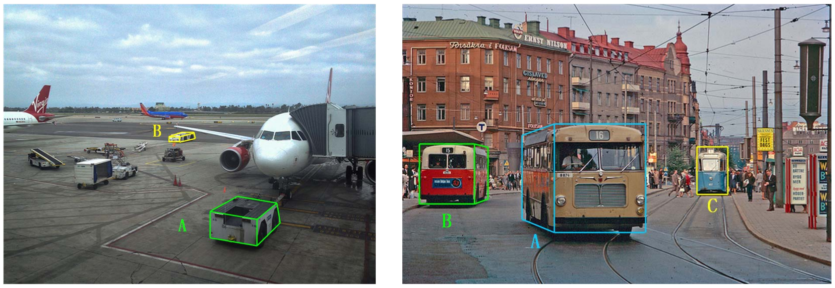
The research motivation behind our work is to consider inherent characteristics of document image. We analyze 3-D projection in natural image and 2-D transformation in layout design of page image, so as to design an effective backbone network for feature extraction from document image. Also, a two-stage detector for page object is realized based on the proposed backbone network. The backbone network is designed in Section III.

III. LFE Network

A. Analyzing 3-D Projection in Natural Image and 2-D Transformation in Layout Design of Page Image

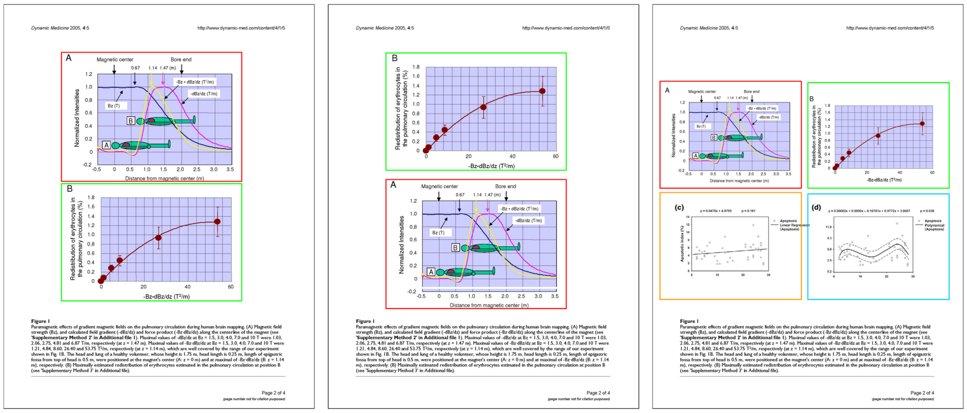
In natural scene, objects are projected on an image with perspective projection. As shown in Fig. 1(a), suppose aircraft tractors A moves to the position of B with the same orientation, the size in image decreases dramatically, and different sides of it appear in image. In this case, the same object shows different sizes and appearances because of 3-D projection. In the field of object detection, aircraft tractor B is called "small object" [22]. Fig. 1(b) shows another case of 3-D projection of three buses: front view of bus A, and back view of bus B and C. If bus A was in position B or C with the same orientation of B or C, its different part was projected in the 2-D image. Without considering zooming in/out in image, front view, back view, and side view of a bus have different image features for CV. Therefore, 3-D projection changes feature of the same object in 2-D image.

On the other hand, 2-D transformation of page objects is essentially different from 3-D projection in natural scene. As shown in Fig. 2(a), there are two tables A and B, of which positions are swapped in Fig. 2(b) or zooming out two tables to add tables C and D in Fig. 2(c). As for page object, swapping position, zooming in/out, or others 2-D transformation is used to communicate information clearly and effectively. To design a good page layout, 2-D transformation of page object rarely involves shearing, not to mention projection transformation.



(a) (b)

Fig. 1. 3-D projection of objects in natural scene (Microsoft COCO (MSCOCO) dataset [42]). (a) Two aircraft tractors. (b) Two aircraft tractors.


Dynamic Medicine 2005, 4:5
A
B
C
Magnetic center
Bore end
Normalized Intensities
Distance from magnetic center (m)
Redistribution of erythrocytes in the pulmonary circulation (%)
-Bz·dBz/dz (T²/m)
Figure 1
Page 2 of 4

(a) (b) (c)

Fig. 2. 2-D transformation of page objects (PubLayNet dataset [43]). (a) Two tables. (b) Exchanging positions. (c) Zooming in/out.

Through analyzing 3-D projection in natural image and 2-D transformation of page object, we get the following conclusions.

1) "Small object" in natural image is caused by perspective projection. It does not mean "small object" in natural image is not important.
2) Page object is not concerned with perspective projection. "Small object" in document image is designed to let large object show more important information. For example, figure is larger than footer. Their sizes will not change.
3) Different orientations of object in natural scene result in different image features.
4) Orientation of page object will not change in the process of layout design.

B. LFE Backbone Network

According to the abovementioned four conclusions, we design a backbone network, and its purpose is to enhance lateral feature in a feature pyramid. The design principles include the following.

1) ResNet [21] is employed to extract fine/coarse features, as shown in Fig. 3. From $\mathbf{C}_2$ to $\mathbf{C}_5$, feature resolution is reduced.
2) Feature enhancement of large page object follows ResNet as shown in Fig. 3 with three up arrows, because large object features in bottom layer remain unchanged when it passes three up arrows. According to conclusions 1) and 2), in document image, large object conveys more important information; therefore, we design this path to enhance feature, which can be denoted as follows:

$$\mathbf{U}_i(1) = \begin{cases} \mathbf{C}_i & i = 2 \\ \mathbf{U}_{i-1}(1) \oplus \mathbf{C}_i & 2 < i \leq 5 \end{cases} \tag{1}$$

where i represents the ith layer, "1" of $\mathbf{U}_i(1)$ means the first block of feature enhancement, and $\oplus$ is feature enhancement operation.
3) LFE follows 2) in Fig. 3 with three down arrows. First, large object features in top layer are enhanced along the top-down path. Second, image feature of page object will not change according to conclusions 3) and 4); hence, the features of small page objects are in lower layers with relatively high resolution. LFE are the following:

$$\mathbf{D}_i(1) = \begin{cases} \mathbf{U}_i(1) & i = 5 \\ \mathbf{D}_{i+1}(1) \oplus \mathbf{U}_i(1) & 2 \leq i < 5 \end{cases} \tag{2}$$

as for $\mathbf{D}_{i+1}(1)$, and it is enhanced by $\mathbf{U}_i(1)$ through a lateral path.
4) Feature enhancement with lateral skip runs after 3), inspired by deep residual learning [21], [44]. Feature can be enhanced by original feature in the same layer (the same resolution)

$$\mathbf{F}_i(1) = \mathbf{D}_i(1) \oplus \mathbf{C}_i \quad 2 \leq i \leq 5. \tag{3}$$

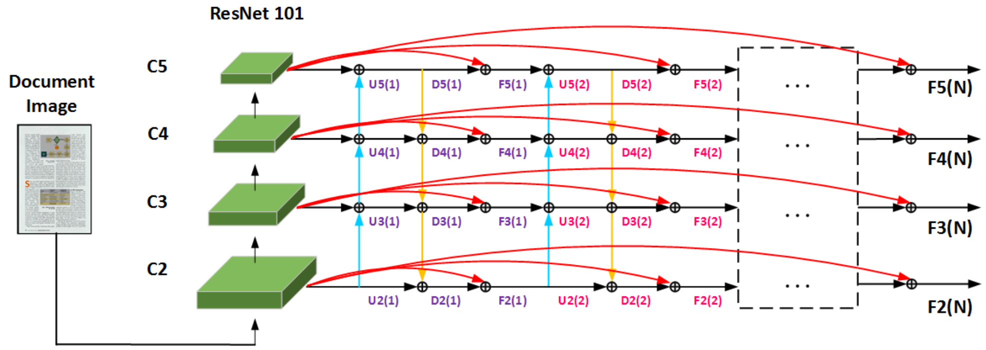


Fig. 3. LFE backbone network.

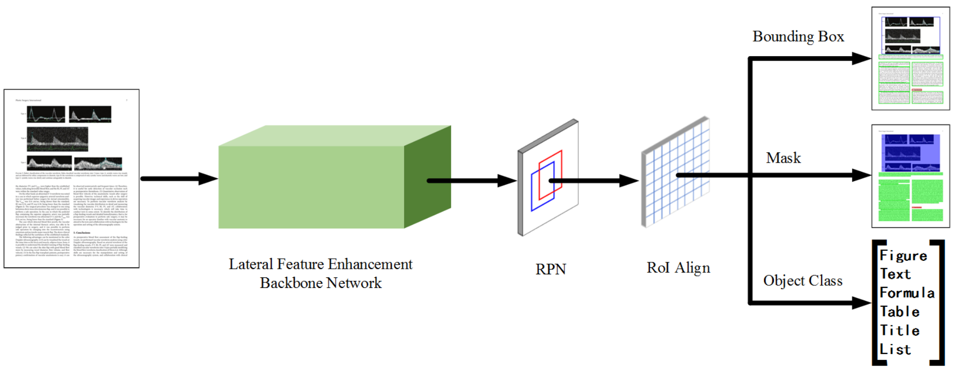


Fig. 4. Framework of LFE network.

5) A feature enhancement block consists of (1)–(3). Fig. 3 shows the first block denoted as "(1)," the second block and the Nth block.

C. Implementation of Backbone Network

The implementation of ResNet-101 follows configuration in [21]. The numbers of feature channel in layers $\mathbf{C}_2$, $\mathbf{C}_3$, $\mathbf{C}_4$, and $\mathbf{C}_5$ are 256, 512, 1024, and 2048, respectively. The resolution reduction rate is 0.5 from a layer to its above neighbor. For example, the width and the height of feature in $\mathbf{C}_3$ are 1/2 of feature in $\mathbf{C}_2$. The feature enhancement operation in (1)–(3) is defined as a sequence in the following:

1) resample resolution;
2) resample channels using 1 × 1 convolution;
3) element-wise addition;
4) 3 × 3 convolution.

When features in a layer are fused with another layer, resolution should be resampled to match the target layer. In step 1), resolution is downsampled from bottom to top, whereas resolution is upsampled along the top-down arrows. Step 2) fuses features using element-wise addition. To let the number of feature channels in a layer match another layer, 1 × 1 convolution is utilized in step 3). Moreover, 1 × 1 convolution extracts and fuses features from all feature channels. In step 4), 3 × 3 convolution is employed to extract and fuse features not only in feature slice but also from all feature channels.

Take $\mathbf{U}_3(1)$ in (1) for example; the resolution of $\mathbf{U}_2(1)$ is downsampled by 1/2, and the channels are upsampled by 2 using 1 × 1 convolution, and then, element-wise addition is performed of $\mathbf{U}_2(1)$ and $\mathbf{C}_3$. Finally, 3 × 3 convolution is implemented to output $\mathbf{U}_3(1)$. In (2), the resolution of $\mathbf{D}_5(1)$ is upsampled by 2; the channels are upsampled by 2 using 1 × 1 convolution, and then, it is added to $\mathbf{U}_4(1)$. Eventually, $\mathbf{D}_4(1)$ is generated using 3 × 3 convolution. There is a little difference in (3) that 2) is not executed. This example is in the first block of Fig. 3. The feature enhancement operation in the second block follows the same sequence 1)–4).

D. LFE Network Architecture

As shown in Fig. 4, the proposed LFE network consists of three parts: the proposed LFE backbone network described in Fig. 3, RPN and RoI alignment. The configuration of the last two parts follows the implementation in [18]. The input is a document image, in which there is a figure, several paragraphs, and a section title. The outputs have three components: bounding boxes indicating location of detected objects, pixel-wise masks predicting locations, and object classification.

IV. Experimental Results

A. Visualization of Feature Enhancement

To get insight into feature enhancement process of the proposed backbone network, features in LFE backbone network (Fig. 3) are visualized in Figs. 5–7. The LFE network (Fig. 4) is trained with eight GPUs(TITAN XP 12 G) for 90 000 iterations, using more than 300k document images in the training set of the PubLayNet dataset [43]. Two

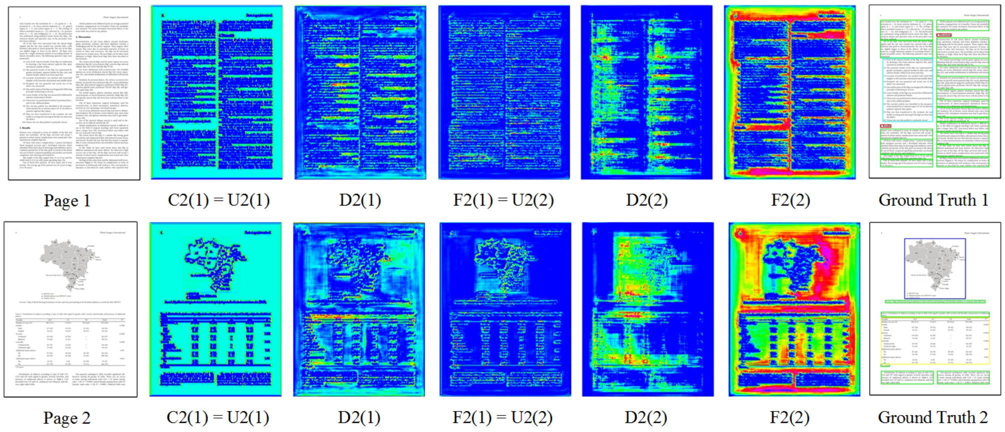


Fig. 5. Visualization of features in the bottom layer of backbone network.

images in development set are used to investigate feature enhancement.

Fig. 5 shows original images, ground truths, and features in the bottom layer of backbone network (Fig. 3). The feature $\mathbf{U}_2(1)$, which is directly from ResNet-101, has more detailed features. Especially, the edge of figure is enhanced in the second row, and so do the lines of table. After "feature enhancement of large page object" and "LFE" in Section III-B, $\mathbf{D}_2(1)$ loses local information, such as edge details in a figure; however, global information is enhanced, that is to say, pixels in foreground merge together to enhance contrast to merging pixels in background.

Using formula (3), $\mathbf{D}_2(1) \oplus \mathbf{C}_2$ to get $\mathbf{F}_2(1)$. Apparently, a detailed feature is enhanced from $\mathbf{C}_2$ to $\mathbf{F}_2(1)$. There is a sharp contrast between foreground and background in $\mathbf{D}_2(2)$. Furthermore, the contrast becomes sharper in $\mathbf{F}_2(2)$. Comparing $\mathbf{F}_2(2)$ with ground truth, in the first row, list object (in "cyan" bounding box), text object (in "green" bounding box), and title object (in "red" bounding box) in "ground truth 1" correspond to "blue" foreground, which is distinguished from background by "red"/"yellow" border in $\mathbf{F}_2(2)$. In the second row, the contrast between figure object and background is sharper.

The implementation of "LFE" leads to accurate detection of "small object," such as in Fig. 6; section title with "red" bounding box in ground truth 1 corresponds to a blue area with clear-cut boundary in $\mathbf{F}_2(2)$. Especially, although the PubLayNet dataset [43] does not provide annotations for page number and header, our backbone network is capable of extracting visual features of them, as shown in Fig. 6.

On the other hand, features in top layer of backbone are also enhanced. Take $\mathbf{C}_5$ of page 1 in Fig. 7 for example, and global information is enhanced from $\mathbf{C}_5$ to $\mathbf{D}_5(1)$ to obtain $\mathbf{F}_5(1)$. Comparing $\mathbf{F}_5(1)$ with ground truth 1 in Fig. 5, $\mathbf{F}_5(1)$ clearly indicates the positions of two columns in page 1. Therefore, the proposed backbone network (Fig. 3) is able to enhance features on both local information and global information.

B. POD Evaluation

The POD competition dataset in ICDAR2017 [32] is used to evaluate the proposed LFE network. This dataset is the most widely used to evaluate the page object detector. It is collected from scientific papers on the web CiteSeer. In total, there are more than 2000 document images with three types of page objects (formula, table, and figure) and various layouts (single column, two columns, and multicolumn). In the competition, 1600 images were used as training set, and test set had 817 images. We used the same training set and test set. Apart from eight detectors in the competition, three more recent detectors are used to evaluate the proposed LFE network, including Li *et al.* [35], YOLACT++ [4], and Adaptive Training Sample Selection (ATSS) [45]. YOLACT++, ATSS, and LFE network are trained with eight GPUs (TITAN XP 12 G) for 90000 iterations.

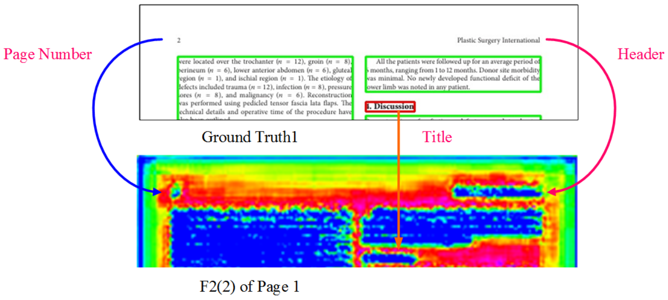


Fig. 6. Detection of unlabeled object and small object.

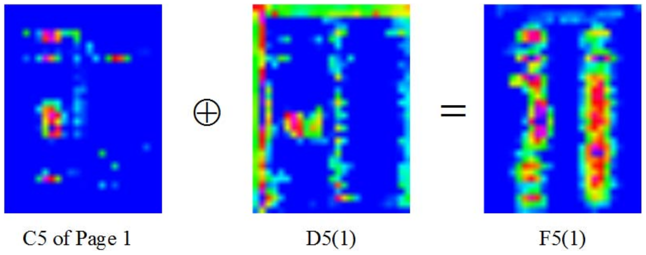


Fig. 7. Visualization of feature enhancement in the top layer.

Table I compares the performance of different detectors with the same configurations of intersection over union (IoU), average precision (AP), and mean of AP (mAP) in ICDAR2017 POD competition [32]. The last two rows show our LFE

TABLE I

EVALUATION OF THE PROPOSED LFE NETWORK ON THE ICDAR2017 POD DATASET [32]

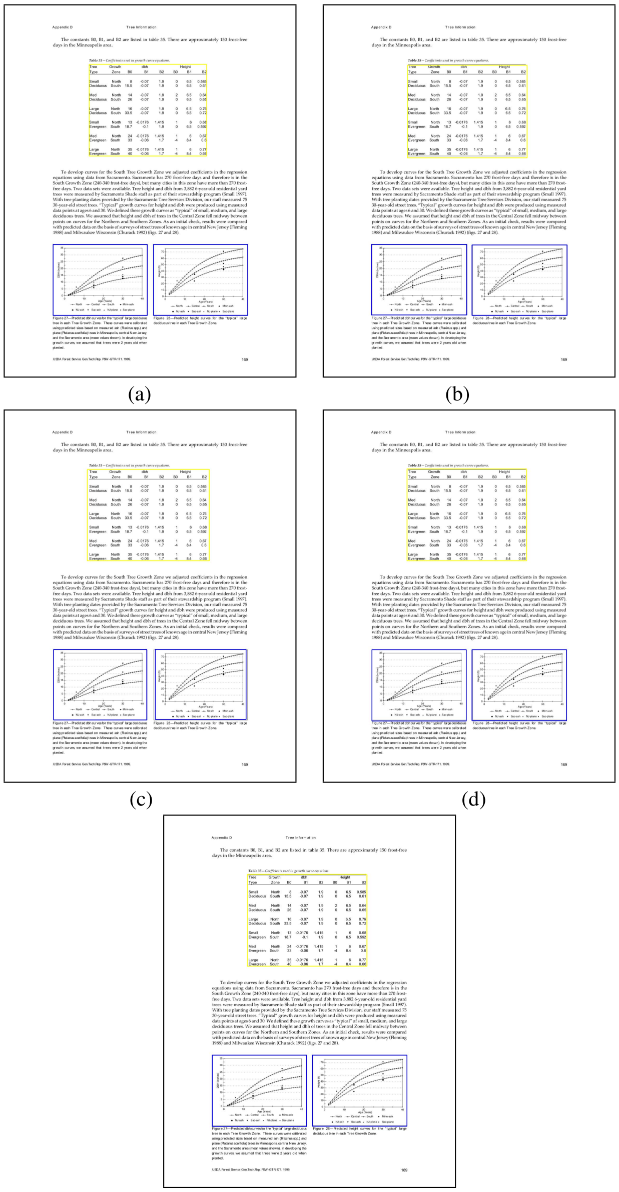
Method	AP(IoU=0.6)				AP(IoU=0.8)			
	Formula	Table	Figure	mAP	Formula	Table	Figure	mAP
NLPR-PAL	0.839	0.933	0.849	0.874	0.816	0.911	0.805	0.844
icstpku	0.849	0.753	0.679	0.760	0.815	0.697	0.597	0.703
FastDetectors	0.474	0.925	0.392	0.597	0.427	0.884	0.365	0.559
Vislnt	0.524	0.914	0.781	0.740	0.117	0.795	0.565	0.492
SOS	0.537	0.931	0.785	0.751	0.109	0.737	0.518	0.455
UITVN	0.193	0.924	0.786	0.634	0.061	0.695	0.554	0.437
Matiai-ee	0.116	0.781	0.325	0.407	0.005	0.626	0.134	0.255
HustVision	0.854	0.938	0.853	0.882	0.293	0.796	0.656	0.582
Li et al. [35]	0.878	0.946	**0.896**	0.907	0.863	**0.923**	**0.854**	0.880
YOLACT++ [4]	0.587	0.929	0.885	0.800	0.299	0.892	0.839	0.677
ATSS [45]	0.929	**0.971**	**0.886**	**0.929**	0.850	**0.944**	**0.853**	0.882
LFE-1	**0.950**	**0.961**	0.868	**0.927**	**0.923**	0.922	0.824	**0.890**
LFE-2	**0.957**	0.959	0.871	**0.929**	**0.926**	**0.923**	0.826	**0.892**

networks with "1" and "2" feature enhancement block(s), according to Fig. 3.

As for IoU threshold of 0.6, our LFE-2 and ATSS [45] obtain the best mAP 0.929. ATSS adaptively selects positive and negative samples according to statistical characteristics of object, and it brings contribution to common object detection. Thus, it might achieve slight improvement on figure detection, which is similar to common object in texture diversity. Whereas, our LFE-2 enhances the features of an image itself. Also, our LFE-1 gets the second mAP of 0.927. Hence, the LFE network with "two" feature enhancement blocks shows better performance than LFE-1 with "one" block. For "small object" formula, our LFE-2 and LFE-1 acquire top two APs: 0.957 and 0.950. Due to "LFE," features of "small object" are enhanced in lateral direction in the bottom layer of backbone network (Fig. 3), so as to our detectors outperform others. When it comes to the table, ATSS and our LFE-1 gain the best two APs: 0.971 and 0.961. Li *et al.* [35] make well use of prior knowledge of line distribution in page, so the detector is able to clearly distinguish figure from other objects (AP for figure: 0.896). Meanwhile, this is why Li *et al.* performs well for all page objects. YOLACT++ [4] is not designed for document image processing; therefore, it is inferior to Li *et al.*

Given the IoU threshold of 0.8, it seems similar conclusions. Our LFE-1 and LFE-2 achieve the best two mAPs: 0.892 and 0.890. They overcome others to detect formula. ATSS gets the best score on table, and Li *et al.* achieve the best AP on figure. It is safe to conclude that our LFE-1 and LFE-2 show the best performance according to mAP and AP on "small object" formula. Li *et al.* performs well because of considering prior knowledge of page object distribution in document image. ATSS is good at detecting table and figure.

As shown in Table I, table detection result of LFE-2 achieves 0.959 mAP at 0.6 IoU threshold, and it is 0.002 lower than LFE-1. This issue might be attributed to slight learning degradation when we train a deeper LFE. However, we mainly focus on investigating the efficiency of the proposed LFE network on document object detection in this article. Also, it can be seen that when the baseline is equipped with single LFE block "LFE-1," we get comparable results on both of the POD dataset and the PubLayNet dataset.



Appendix D Tree Information

The constants B0, B1, and B2 are listed in table 35. There are approximately 150 frost-free days in the Minneapolis area.

Table 35—Coefficients used in growth curve equations.

Tree	Growth	dbh			Height		
Type	Zone	B0	B1	B2	B0	B1	B2
Small	North	8	-0.07	1.9	0	6.5	0.585
Deciduous	South	15.5	-0.07	1.9	0	6.5	0.61
Med	North	14	-0.07	1.9	2	6.5	0.64
Deciduous	South	26	-0.07	1.9	0	6.5	0.65
Large	North	16	-0.07	1.9	0	6.5	0.76
Deciduous	South	33.5	-0.07	1.9	0	6.5	0.72
Small	North	13	-0.0176	1.415	1	6	0.68
Evergreen	South	18.7	-0.1	1.9	0	6.5	0.592
Med	North	24	-0.0176	1.415	1	6	0.67
Evergreen	South	33	-0.06	1.7	-4	8.4	0.6
Large	North	35	-0.0176	1.415	1	6	0.77
Evergreen	South	40	-0.06	1.7	-4	8.4	0.66

To develop curves for the South Tree Growth Zone we adjusted coefficients in the regression equations using data from Sacramento. Sacramento has 270 frost-free days and therefore is in the South Growth Zone (240-340 frost-free days), but many cities in this zone have more than 270 frost-free days. Two data sets were available. Tree height and dbh from 3,882 6-year-old residential yard trees were measured by Sacramento Shade staff as part of their stewardship program (Small 1997). With tree planting dates provided by the Sacramento Tree Services Division, our staff measured 75 30-year-old street trees. "Typical" growth curves for height and dbh were produced using measured data points at ages 6 and 30. We defined these growth curves as "typical" of small, medium, and large deciduous trees. We assumed that height and dbh of trees in the Central Zone fell midway between points on curves for the Northern and Southern Zones. As an initial check, results were compared with predicted data on the basis of surveys of street trees of known age in central New Jersey (Fleming 1988) and Milwaukee Wisconsin (Churack 1992) (figs. 27 and 28).

Figure 27—Predicted dbh curves for the "typical" large deciduous tree in each Tree Growth Zone. These curves were calibrated using predicted sizes based on measured ash (Fraxinus spp.) and plane (Platanus acerifolia) trees in Minneapolis, central New Jersey, and the Sacramento area (mean values shown). In developing the growth curves, we assumed that trees were 2 years old when planted.

Figure 28—Predicted height curves for the "typical" large deciduous tree in each Tree Growth Zone.

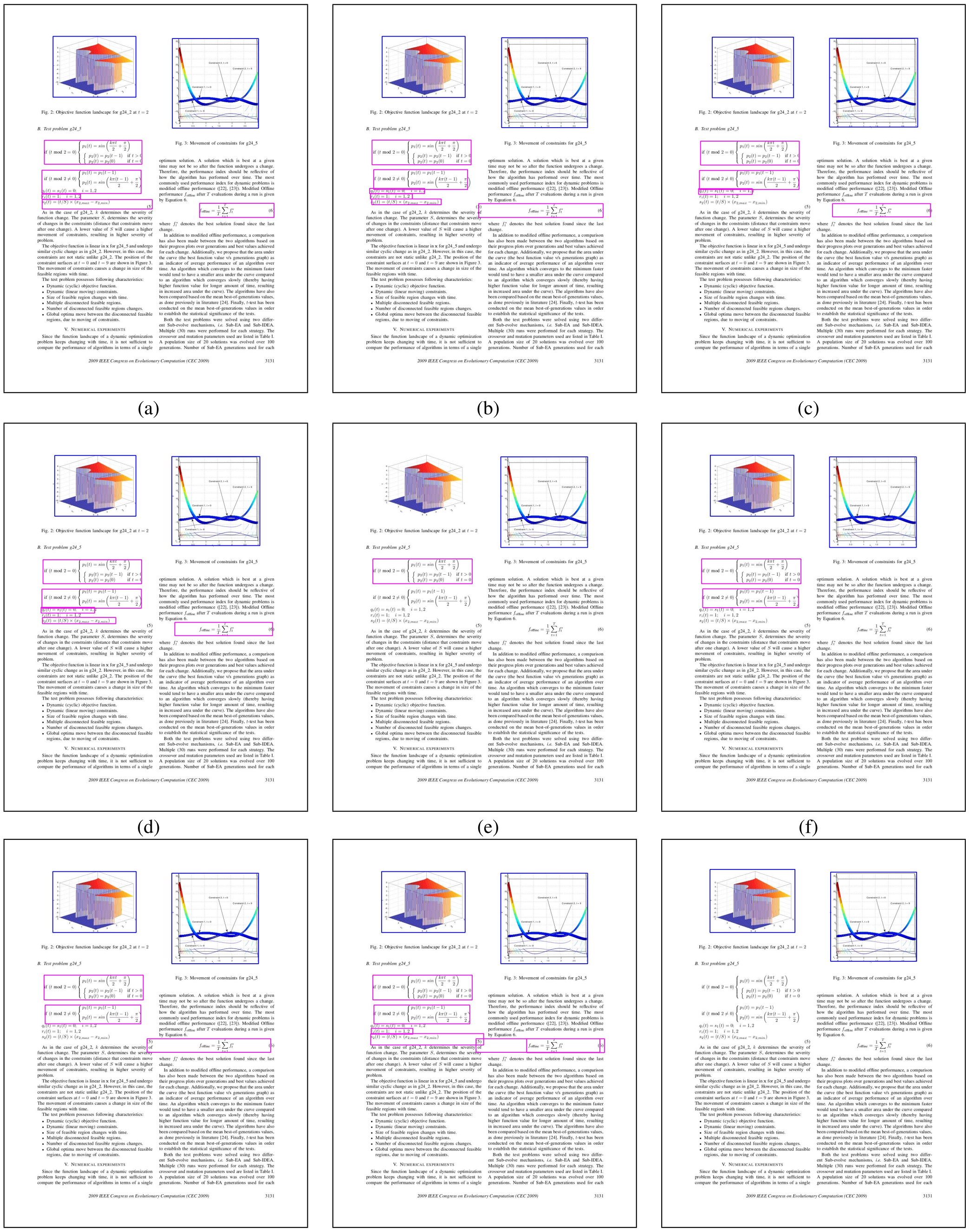
USDA Forest Service Gen.Tech.Rep. PSW-GTR-171. 1999. 169

(a) (b) (c) (d) (e)

Fig. 8. Visualization of table and figure detection on the POD2017 dataset with a CP of 0.9. (a) Ground truth. (b) LFE-2, CP = 0.9. (c) LFE-1, CP = 0.9. (d) ATSS, CP = 0.9. (e) YOLACT++, CP = 0.9.

C. Comparison With Multimodal Networks

In ICDAR 2021 SLP Competition (ICDAR2021-SLP) [33], [34], several top-level detectors are using multimodal networks. To compare performance with the multimodal detector, the same dataset PubLayNet dataset [43] in ICDAR2021-SLP is used to train our LFE-1 and LFE-2. Totally, more than 340k document images are randomly split into train, development, and test sets, and the ratio is 32:1:1. The dataset includes five types of document objects: text, title, table, figure, and list. The ICDAR2021-SLP competition uses the detection evaluation metrics of Common Objects in Context (COCO) [46]: AP and mAP averaged over multiple intersection over ten IoU from



(a) (b) (c) (d) (e) (f) (g) (h) (i)

Fig. 9. Visualization of formula detection on the POD2017 dataset with various CP values. (a) Ground truth. (b) LFE-2, CP = 0.9. (c) LFE-1, CP = 0.9. (d) LFE-1, CP = 0.8. (e) ATSS, CP = 0.9. (f) ATSS, CP = 0.8. (g) ATSS, CP = 0.7–0.5. (h) ATSS, CP = 0.4. (i) YOLACT++, CP = 0.9–0.4.

0.50 to 0.95 with a step size of 0.05. In Table II, YOLACT++, ATSS, and LFE network are trained with eight GPUs (TITAN XP 12 G) for 90 000 iterations.

As shown in Table II, vision, semantics, and relations (VSR) and SRK achieve two highest mAPs: 0.957 and 0.950. VSR takes advantage of portable document format (PDF) parsing to extract structured information, such as texts and their position for a natural language processing (NLP)-based flow in its network framework. Meanwhile, the other flow is CV-based to process document image. Different from VSR, SRK utilizes two models, all based on CV, in which one is designed for small page object: title, the other is for other objects. Our LFE-1 and LFE-2 get mAP: 0.950 and 0.948; therefore, they are comparable to VSR and SRK.

The aforementioned methods make full use of complementary information for page layout modeling. For instances, list objects have more indentations in page layout. Thus, both of SRK and VSR achieve better list detection performances than LFE with layout sensitiveness. On text, LFE-1 and LFE-2 outperform other methods. Generally, texts take up most of the

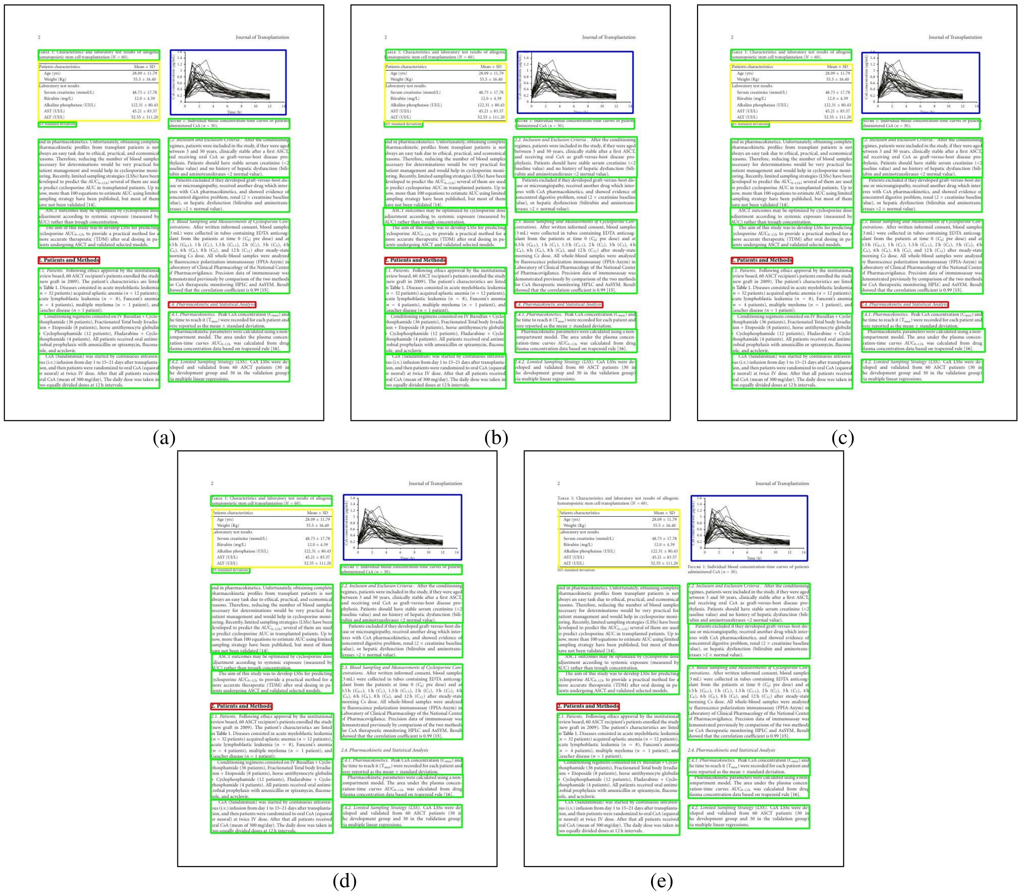

(a) (b) (c) (d) (e)

Fig. 10. Visualization of POD on the PubLayNet dataset with CP 0.7. (a) Ground truth. (b) LFE-2, CP = 0.7. (c) LFE-1, CP = 0.7. (d) ATSS, CP = 0.7. (e) YOLACT++, CP = 0.7.

TABLE II

EVALUATION OF THE PROPOSED LFE NETWORK ON THE PUBLAYNET DATASET [43]

Method	AP (IoU from 0.50 to 0.95 with a step size of 0.05)					
	Text	Title	List	Table	Figure	mAP
SRK [47]	0.947	0.900	**0.951**	0.972	**0.980**	**0.950**
VSR [47]	0.967	**0.923**	**0.946**	0.970	**0.979**	**0.957**
YOLACT++ [4]	0.915	0.620	0.849	0.897	0.892	0.835
ATSS [45]	0.973	0.850	0.944	**0.978**	0.932	0.935
LFE-1	**0.980**	**0.902**	0.926	**0.984**	0.950	0.948
LFE-2	**0.982**	**0.902**	0.928	**0.984**	0.954	**0.950**

space of the page. Features of texts are effectively enhanced by our backbone network (Fig. 3), so that our method gets good performance.

As for small page object title, VSR achieves the best AP 0.923, and our LFE method performs well with an AP of 0.902. This result verifies the effectiveness of that: “LFE” leads to accurate detection of “small object.”

The proposed LFE-1 and LFE-2 achieve the best AP 0.984 on table, whereas VSR and SRK obtain better APs on list and figure.

Overall, in spite of being unimodal, using feature enhancement strategies 2)–5) in Section III-B, our method is comparable to multimodal methods VSR and SRK. For text, title, and table, our LFE method performs well, but for list and figure, multimodal methods perform better.

D. Visualization of POD With Various CPs

For an insight into the detection abilities of our LFE-1 and LFE-2, detection results with various confidence probabilities (CPs) are visualized in Figs. 8–10. The CP is the most widely used as an output representing the likelihood for each predicted class in the state-of-the-art detectors, such as YOLACT++ [4], ATSS [45], Faster R-CNN [17], Mask R-CNN [18], and so on, and is used in the detection evaluation metrics of COCO [46].

As can be seen from Fig. 8, yellow and blue bounding boxes are used to locate table and figure on the POD2017 dataset. LFE-2, LFE-1, ATSS, and YOLACT++ produce the same visual effect with the CP of 0.9 from Fig. 8(b)–(e). That is to say, for a strict metric CP of 0.9, four detectors achieve well visual effect according to ground truth in Fig. 8(a).

Fig. 9 shows different visual effects of four detectors with various CP values. Fuchsia bounding boxes locate formula objects on the POD2017 dataset. Compared with ground truth in Fig. 9(a), LFE-2 finds out all figures and formulas, although the bounding boxes of formulas in the middle of the left column and the right column do not perfectly match the ground truth, with the strict CP of 0.9 in Fig. 9(b).

As shown in Fig. 9(c), there is a small part of formulas in the middle of the left column that cannot be detected by LFE-1 with a CP of 0.9. When the CP value decreases to 0.8, there is still a formula LFE-1 that cannot find out. The number of blocks of feature enhancement has a significant impact on the detection of "small page object" formula, as described in Section III-B: "LFE" results in better detection of "small object," because LFE-2 has one more block of feature enhancement than LFE-1.

ATSS does not perform good enough with a CP of 0.9, as shown in Fig. 9(e). This detector only finds out a figure and a small part of formulas. This situation improves when the metric becomes weaker, CP from 0.8 [Fig. 9(f)] to 0.4 [Fig. 9(h)], two figures and all formula are detected gradually. In Fig. 9(i), YOLACT++ detects two figures when CP ranges from 0.9 to 0.4.

Fig. 10 compares visual effects of four detectors on the PubLayNet dataset with a CP of 0.7. Yellow, blue, green, and red bounding boxes locate table, figure, text, and title objects, respectively. As can be seen from Fig. 10(b) and (c), LFE-2 and LFE-1 find out all page objects. ATSS misses a title in the right column, in Fig. 10(d). YOLACT++ performs worse in Fig. 10(e). It cannot detect both the title in the right column and texts above and under the table in the left column.

Based on the preceding analysis of visual effect, we see that the proposed LFE-2 and LFE-1 outperform ATSS and YOLACT++ on "small page objects," such as formulas on the POD2017 dataset, titles on the PubLayNet dataset, and one or two lines of texts on the PubLayNet dataset.

Additionally, the proposed detector can be applied in the SparkFun Jetbot [48], which is a popular platform and is capable of deep learning inference. It is powered by NVIDIA Jetson Nano [49], which has 128 NVIDIA cores. Besides, the Jetson Nano Developer Kit provides software support for parallel implementation of different applications. Images are captured by a Universal Serial Bus (USB) Web camera, and the High-Definition Multimedia Interface (HDMI) interface is used for visualizing inference results. Our detector can be employed in the Pytorch framework of the SparkFun Jetbot to detect scene text.

V. Conclusion

In this work, we propose the LFE network by analyzing page object distribution in the page layout process. The LFE backbone network aggregates bottom-up and top-down feature pyramids sequentially as an enhancement block, and the lateral skip connection is added after each enhancement block for retaining the output feature details. In the LFE backbone network, the introduction of low-level spatial information to deeper layers can help large objects recognition. Meanwhile, the propagation of high-level semantic signals upsampled to higher resolution low-level feature layers enhances the performance semantically. The LFE network achieves great results with the mAP of 0.950 on PubLayNet, and the mAP of 0.892 on POD with strict metric IoU = 0.8. Extensive experimental results on the two state-of-the-art databases demonstrate that our LFE network retains more original feature information and enhances feature extraction and feature representation ability on document images. It is capable of improving the document object detection accuracy.

References

[1] L. Jiao *et al.*, "A survey of deep learning-based object detection," *IEEE Access*, vol. 7, pp. 128837–128868, 2019.

[2] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.

[3] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," 2020, *arXiv:2004.10934*.

[4] D. Bolya, C. Zhou, F. Xiao, and Y. J. Lee, "YOLACT++: Better real-time instance segmentation," 2019, *arXiv:1912.06218*.

[5] W. Liu *et al.*, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 21–37.

[6] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.

[7] Q. Zhao *et al.*, "M2Det: A single-shot object detector based on multi-level feature pyramid network," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 9259–9266.

[8] S. Zhang, L. Wen, X. Bian, Z. Lei, and S. Z. Li, "Single-shot refinement neural network for object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4203–4212.

[9] Y. Cai *et al.*, "YOLOv4–5D: An effective and efficient object detector for autonomous driving," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–13, 2021.

[10] L. Guan, L. Jia, Z. Xie, and C. Yin, "A lightweight framework for obstacle detection in the railway image based on fast region proposal and improved YOLO-tiny network," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–16, 2022.

[11] X. Lu, J. Ji, Z. Xing, and Q. Miao, "Attention and feature fusion SSD for remote sensing object detection," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–9, 2021.

[12] R. Gao *et al.*, "Small foreign metal objects detection in X-ray images of clothing products using faster R-CNN and feature pyramid network," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–11, 2021.

[13] Z. Liu, Y. Lyu, L. Wang, and Z. Han, "Detection approach based on an improved faster RCNN for brace sleeve screws in high-speed railways," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 7, pp. 4395–4403, Jul. 2020.

[14] H. Bi *et al.*, "SRRV: A novel document object detector based on spatial-related relation and vision," *IEEE Trans. Multimedia*, early access, Apr. 7, 2022, doi: 10.1109/TMM.2022.3165717.

[15] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 580–587.

[16] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.

[17] 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.*, vol. 28, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds. Curran Associates, 2015, pp. 91–99.

[18] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 2961–2969.

[19] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 25. Stateline, NV, USA, Dec. 2012, pp. 1097–1105.

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

[21] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[22] T.-Y. Lin, P. Dollar, 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.

[23] A. K. Gupta, A. Seal, P. Khanna, E. Herrera-Viedma, and O. Krejcar, "ALMNet: Adjacent layer driven multiscale features for salient object detection," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–14, 2021.

[24] S. Schreiber, S. Agne, I. Wolf, A. Dengel, and S. Ahmed, "DeepDeSRT: Deep learning for detection and structure recognition of tables in document images," in *Proc. 14th IAPR Int. Conf. Document Anal. Recognit. (ICDAR)*, Nov. 2017, pp. 1162–1167.

[25] A. Gilani, S. R. Qasim, I. Malik, and F. Shafait, "Table detection using deep learning," in *Proc. 14th IAPR Int. Conf. Document Anal. Recognit. (ICDAR)*, Nov. 2017, pp. 771–776.

[26] L. Gao, X. Yi, Y. Liao, Z. Jiang, Z. Yan, and Z. Tang, "A deep learning-based formula detection method for PDF documents," in *Proc. 14th IAPR Int. Conf. Document Anal. Recognit. (ICDAR)*, vol. 1, Nov. 2017, pp. 553–558.

[27] C.-H. Xu, C. Shi, and Y.-N. Chen, "End-to-end dilated convolution network for document image semantic segmentation," *J. Central South Univ.*, vol. 28, no. 6, pp. 1765–1774, Jun. 2021.

[28] C. Xu *et al.*, "A page object detection method based on mask R-CNN," *IEEE Access*, vol. 9, pp. 143448–143457, 2021.

[29] X. Yi, L. Gao, Y. Liao, X. Zhang, R. Liu, and Z. Jiang, "CNN based page object detection in document images," in *Proc. 14th IAPR Int. Conf. Document Anal. Recognit. (ICDAR)*, Nov. 2017, pp. 230–235.

[30] R. Mondal, S. Bhowmik, and R. Sarkar, "TsegGAN: A generative adversarial network for segmenting touching nontext components from text ones in handwriting," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–10, 2021.

[31] ICDAR2017-Organizing-Committee. (2017). *ICDAR2017 Competition on Page Object Detection*. Website. [Online]. Available: http://u-pat.org/ICDAR2017/program_competitions.php

[32] L. Gao, X. Yi, Z. Jiang, L. Hao, and Z. Tang, "ICDAR2017 competition on page object detection," in *Proc. 14th IAPR Int. Conf. Document Anal. Recognit. (ICDAR)*, Nov. 2017, pp. 1417–1422.

[33] ICDAR2021-Organizing-Committee. (2021). *ICDAR 2021 Competition on Scientific Literature Parsing*. Website. [Online]. Available: https://icdar2021.org/program-2/competitions/competition-on-scientific-literature-parsing/

[34] A. J. Yepes, X. Zhong, and D. Burdick, "ICDAR 2021 competition on scientific literature parsing," 2021, *arXiv:2106.14616*.

[35] X.-H. Li, F. Yin, and C.-L. Liu, "Page object detection from PDF document images by deep structured prediction and supervised clustering," in *Proc. 24th Int. Conf. Pattern Recognit. (ICPR)*, Aug. 2018, pp. 3627–3632.

[36] J. Younas *et al.*, "Fi-Fo detector: Figure and formula detection using deformable networks," *Appl. Sci.*, vol. 10, no. 18, p. 6460, Sep. 2020.

[37] C.-Y. Fu, W. Liu, A. Ranga, A. Tyagi, and A. C. Berg, "DSSD: Deconvolutional single shot detector," 2017, *arXiv:1701.06659*.

[38] T. Liang, Y. Wang, Q. Zhao, H. Zhang, Z. Tang, and H. Ling, "MFPN: A novel mixture feature pyramid network of multiple architectures for object detection," 2019, *arXiv:1912.09748*.

[39] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *Proc. IEEE Conf. Comput. Vis. pattern Recognit.*, 2018, pp. 8759–8768.

[40] G. Ghiasi, T.-Y. Lin, and Q. V. Le, "NAS-FPN: Learning scalable feature pyramid architecture for object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7036–7045.

[41] Z. Tian, W. Huang, T. He, P. He, and Y. Qiao, "Detecting text in natural image with connectionist text proposal network," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 56–72.

[42] T.-Y. Lin *et al.*, "Microsoft coco: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 740–755.

[43] X. Zhong, J. Tang, and A. J. Yepes, "PubLayNet: Largest dataset ever for document layout analysis," in *Proc. Int. Conf. Document Anal. Recognit. (ICDAR)*, Sep. 2019, pp. 1015–1022.

[44] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 630–645.

[45] S. Zhang, C. Chi, Y. Yao, Z. Lei, and S. Z. Li, "Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 9759–9768.

[46] C.-O. in Context. (2021). *COCO Detection Evaluation*. Website. [Online]. Available: https://cocodataset.org/#detection-eval

[47] P. Zhang *et al.*, "VSR: A unified framework for document layout analysis combining vision, semantics and relations," 2021, *arXiv:2105.06220*.

[48] *SparkFun-Jetbot*. Website. Accessed: Aug. 31, 2022. [Online]. Available: https://www.sparkfun.com/Jetson

[49] *NVIDIA-Jetson-Nano*. Website. Accessed: Aug. 31, 2022. [Online]. Available: https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-nano/

Cao Shi received the Ph.D. degree from Central South University, Changsha, China, in 2011.

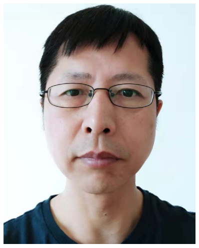
He was a Post-Doctoral Research Fellow with Peking University, Beijing, China, from 2011 to 2013. He is currently with the School of Information Science and Technology, Qingdao University of Science and Technology, Qingdao, China. His research interests include image, video processing, and artificial intelligence.

Canhui Xu received the Ph.D. degree from Central South University, Changsha, China, in 2011.

She was a Visiting Ph.D. Student with the Imperial College London, London, U.K., from 2009 to 2010. She was a Post-Doctoral Research Fellow with Peking University, Beijing, China, from 2012 to 2014. She was a Visiting Scholar with Arizona State University, Tempe, AZ, USA, from 2019 to 2020. She is currently with the School of Information Science and Technology, Qingdao University of Science and Technology, Qingdao, China. Her research interests include deep learning, document layout analysis, and image understanding.

Hengyue Bi is currently pursuing the M.S. degree with the School of Information Science and Technology, Qingdao University of Science and Technology, Qingdao, China.

His research interests include computer vision and machine learning.



Yuanzhi Cheng received the Ph.D. degree from the Harbin Institute of Technology, Harbin, China.

He was with the School of Computer Science and Technology, Harbin Institute of Technology, until 2020. He is currently a Professor with the School of Information Science and Technology, Qingdao University of Science and Technology, Qingdao, China. His research interests include pattern recognition, image processing, and computer-assisted surgical system.

Yuteng Li is currently pursuing the M.S. degree with the School of Information Science and Technology, Qingdao University of Science and Technology, Qingdao, China.

His research interests include deep learning, computer vision, and image processing.

Honghong Zhang is currently pursuing the M.S. degree with the School of Information Science and Technology, Qingdao University of Science and Technology, Qingdao, China.

Her research interests include artificial intelligence, computer vision, and image processing.