2025 5th International Conference on Artificial Intelligence and Industrial Technology Applications (AIITA)

Research on the Application of Improved Algorithm Based on YOLO in the Industrial Field

Yinqiang Yan
Big Data Product Line
Changyang Technology (Beijing) Co. , Ltd.
Beijing, People’s Republic of China
E-mail: yqyan@hotmail.com

Xingren Yao
Big Data Product Line
Changyang Technology (Beijing) Co. , Ltd.
Chengdu, People’s Republic of China
E-mail: xingren.yao@cy-tech.net

Maoyao Xiong
Big Data Product Line
Changyang Technology (Beijing) Co. , Ltd.
Chengdu, People’s Republic of China
E-mail: maoyao.xiong@cy-tech.net

Haikun Jiang
Board of Directors Office
Changyang Technology (Beijing) Co. , Ltd.
Beijing, People’s Republic of China

Yu Fan
Board of Directors Office
Changyang Technology (Beijing) Co. , Ltd.
Beijing, People’s Republic of China

***Abstract*-To address the challenges of model detection accuracy and generalization in industrial visual AI applications, this paper presents the YOLOCy neural network framework based on the YOLO series. It optimizes network architecture and application levels, enhancing detection accuracy while maintaining efficiency. Industrial applications verify its optimization and effectiveness. Combined with the optimization engine, it improves usability in production safety. The YOLOCy - S model (improved from YOLOv5 - S) increases mAP by 6.3% with a 0.2ms inference time increase and has an accuracy rate over 0.94 in real - world data.**

Keywords-YOLOCy; Visual AI; Deep Learning; Object Detection; Progressive Analyzer; Industrial Field

I. Introduction

With the rapid development of information technology, artificial intelligence has become an important driving force for promoting innovation and development in various industries. As an important branch in the field of artificial intelligence, visual AI algorithms, with their powerful capabilities in image recognition, target detection and tracking, and scene understanding, have been widely applied in the industrial field and achieved remarkable results. However, there are still some challenges and difficulties. In response to the complexity of industrial scenes, challenges such as lighting changes and occlusion, and issues of real-time performance and robustness of visual AI algorithms, researchers have improved the adaptability of visual AI algorithms to complex scenes by introducing technologies such as attention mechanisms[1-2]; and improved the real-time performance and accuracy of visual AI algorithms by optimizing model structures and algorithm processes. In this paper, in-depth research has been conducted on algorithm structure, training methods, and application research. The algorithm framework YOLOCy is proposed and verified in practice in the industrial field, achieving certain research results.

II. Related Technologies

As a core branch in the field of artificial intelligence, visual AI technology has been widely studied and applied in recent years. Target detection algorithms based on convolutional neural networks (CNN) have gradually become the mainstream. Among them, the most representative algorithms include Faster R-CNN[3], YOLO[4-10], etc. The Faster R-CNN algorithm is the representative of two-stage detection algorithms, and the YOLO series algorithms are the most representative end-to-end target detection algorithms. With the continuous optimization and iteration of the YOLO series, the popularity and influence of this algorithm have been greatly improved, and it has become one of the most favored target detection algorithms in the industrial field.

The SpaceToDepth [11] technology optimizes the processing of image data by rearranging 2x2 pixel blocks into depth channels without increasing model complexity. The research of Raja et al. [12] shows that this technology can significantly improve the accuracy of image classification and object detection and improve the running efficiency of the model on GPU/TPU. RepVGG [13] is an efficient convolutional neural network architecture that simplifies the model structure through reparameterization while maintaining performance and improving inference speed. UniRepLKNet [14] is a large-core convolutional neural network architecture that enhances the large-core layer by fusing non-dilated and dilated small-core layers to optimize the spatial pattern hierarchy and network representation ability, showing its unique construction advantages. The AuxBranch [15] technology enhances the model's feature extraction ability by introducing auxiliary branches, promotes information flow and sharing, and at the same time maintains a lightweight structure to reduce computational overhead.

979-8-3315-0976-7/25/$31.00 ©2025 IEEE

III. YOLOCy Algorithm Framework

Based on the YOLO series object detection algorithm, this paper proposes the YOLOCy detection algorithm for the improved network structure based on the YOLO series in terms of network structure optimization. In terms of training method optimization, the AuxBranch auxiliary training method is introduced to reduce video memory occupation and increase training speed at the same time. At the application level, an alarm filter module is designed based on a large amount of practical application experience, which effectively improves the model detection effect in practical applications. This paper combines the YOLOCy detection algorithm, AuxBranch auxiliary training and alarm filter module, and proposes the YOLOCy algorithm framework.

A. YOLOCy Detection Algorithm

In this paper, the YOLOCy model is proposed based on YOLO. The main structure of the model is shown in Figure 1(a). 1) SpaceMerge module. In order to alleviate the loss of a large amount of shallow feature information caused by downsampling using the stem module in CNN models, this paper proposes the SpaceMerge module by referring to SpaceToDepth[11]. Firstly, a pointwise convolution is added in SpaceToDepth to avoid the defect of limited output channel number of this module, increase universality and promote the fusion of feature information between channels to achieve dynamic channel output. Secondly, group convolution and fixed weight methods are adopted to optimize the parameter characterization ability and improve the inference efficiency on domestic equipment at the same time. The structure of the SpaceMerge module is shown in Figure 1(b) below. 2) RepC3 module. Referring to the idea of reparameterization of RepVGG, this paper replaces one Conv and Block in the C3 structure with RepConv and RepBlock respectively to construct the RepC3 module and enhance the feature extraction ability of the bottleneck. The structure of the RepC3 module is shown in Figure 1(c). 3) UniRepC3 module. In this paper, UniRepBlock is used to replace RepConv in the lightweight branch RepC3 structure to construct the UniRepC3 module, and UniRepC3 is used to replace RepC3 in the stage-3 part of the Backbone to expect to find a balance between speed, memory, and accuracy. The structure diagram of the UniRepC3 module is shown in Figure 1(d).

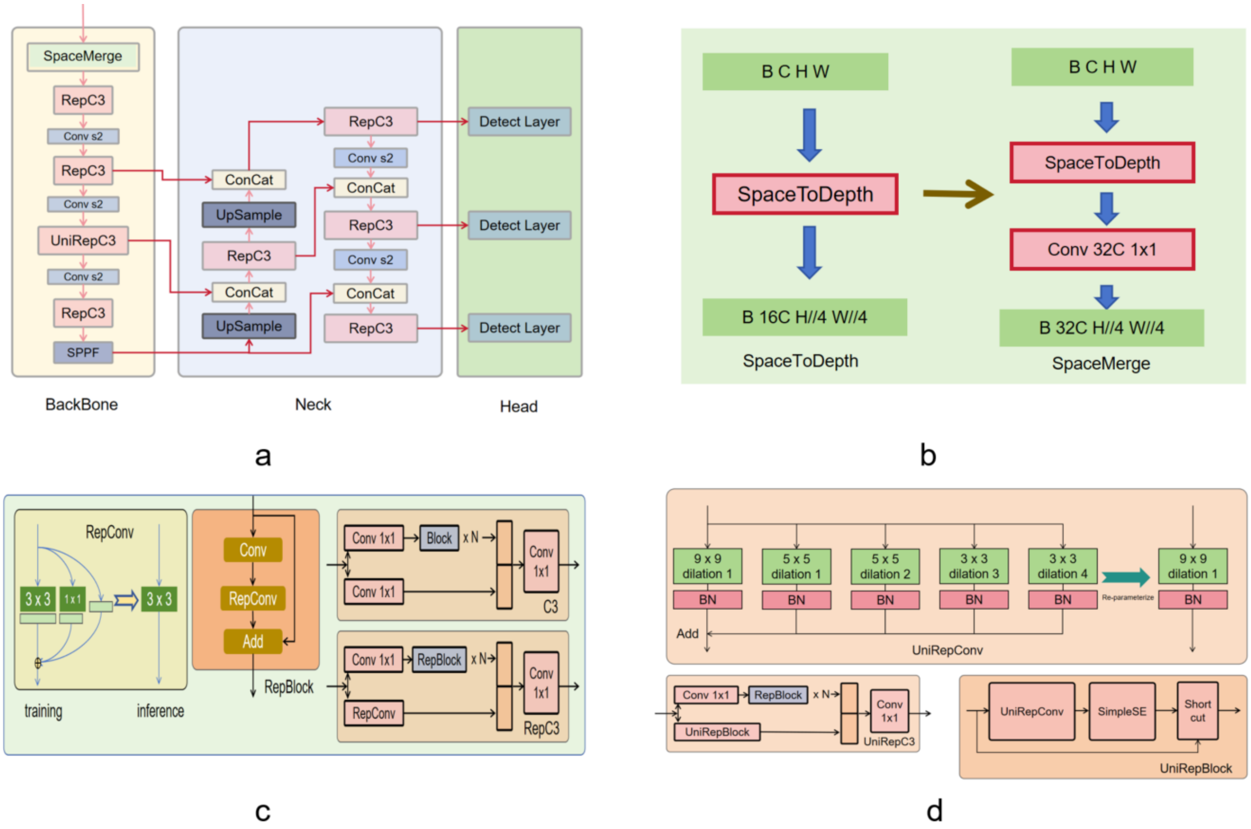

Figure 1. Structure Design

In addition, in order to deal with the problem of a sharp increase in the number of parameters and a significant drop in inference performance caused by the use of the more complex RepC3 module and UniRepC3 module, this paper prunes the small-size network structure, reduces the number of channels and network depth, and reduces network parameters. The parameter comparison with YOLOv5-S is shown in Table I below.

TABLE I. Comparison of YOLOCy-S and YOLOv5-S Parameters

Model	Param	GFLOPS	Forward (ms)	GMem (GB)
Yolov5-S	7235389	16.63	35.07	0.690
YoloCy-S	15035309	35.6	59.88	0.709
YoloCy-S-Prune	10907805	24.4	38.76	0.659

As shown in Table I, the parameter count and computational load of YOLOCy-S are more than twice those of YOLOv5-S, with significantly higher inference speed and memory usage. The pruned version of YOLOCy-S shows a substantial reduction in parameter count and computational load, with inference speed and memory usage comparable to YOLOv5-S.

B. *AuxBranch Auxiliary Training*

The AuxBranch framework can automatically generate auxiliary training branches for all models. This branch is only used in the training stage and will be removed during inference, so it will not increase the inference burden. In the Multi-Aux scheme, we use the custom Hybrid Distribution operator to perform information mixing processing on feature maps of different sizes to enhance the ability of the auxiliary training head to capture high-frequency and low-frequency information, thereby improving the perception ability of the model. In addition, we construct AuxBranch and AuxHead by deeply copying the last three stages and the Head part of the original model. During training, we innovatively concatenate the prediction results of AuxHead and Head to jointly calculate Loss, which not only reduces memory usage and improves training efficiency but also accelerates model convergence. The Multi-Aux structure is shown in Figure 2 (L). This paper draws on the practice of CBNetV2[16] and adds additional auxiliary modules to Multi-Aux to obtain Triple-Aux. A new AuxBackBone and AuxHead are copied to perform deep supervision on the Backbone and increase the backpropagation gradient of the Backbone during Backward to achieve the purpose of accelerating fitting. The Triple-Aux structure is shown in Figure 2 (R).

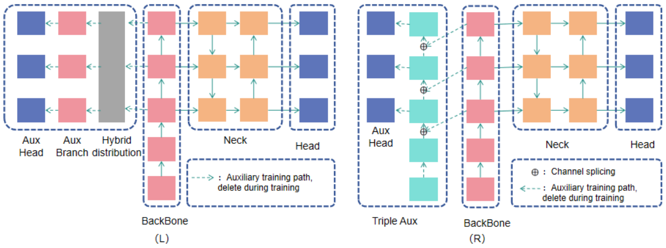

Figure 2. AuxBranch training structure diagram

In addition, since the two newly added auxiliary modules occupy too much video memory, we use the Gradient Checkpoint[17] technology to achieve the purpose of saving video memory.

C. *Alarm Progressive Analyzer Module*

To cope with the complex and changeable application environment of industrial scenes, this paper designs an alarm filter module to assist model detection. In the entire detection process, operations such as correction, cropping, and filtering are performed on the detected image, and rule filtering is performed on the model detection results to achieve the goal of optimizing the overall implementation effect. The filter module consists of three parts: filter operator library, filter configuration, and filter pipeline. The structure is shown in Figure 3 below:

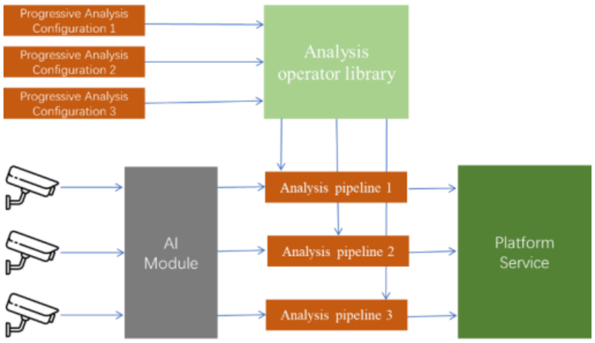
Progressive Analysis Configuration 1
Progressive Analysis Configuration 2
Progressive Analysis Configuration 3
Analysis operator library
AI Module
Analysis pipeline 1
Analysis pipeline 2
Analysis pipeline 3
Platform Service

Figure 3. Analyzer Module Structure Design

As shown in Figure 3, the filter module is deployed after the AI model detection and performs rule filtering on the AI model detection results. The partial operator function list is shown in Table II below.

TABLE II. PROGRESSIVE ANALYSIS OPERATOR

Progressive Analysis Operator	Operator Function
OverLapUnit	When overlapping boxes of the same category are present, keep the box with the highest score and filter out the other overlapping boxes
BlurUnit	When the image is blurry, filter out all detection boxes
OverExposuUnit	When the image is overexposed or too dark, filter out all detection boxes
IomUnit	Category A must have at least one overlapping Category B. If not, filter out Category A
SequenceUnit	Require that the same type of detection at the same location appears consistently as the same category across multiple consecutive frames
dynamicUnit	Require that the same location in consecutive frames maintains the same detection category, and the pixel characteristics of the target area differ

With the flexible configuration of progressive analysis operators, the system can perform progressive fusion analysis on videos from different cameras, enhancing accuracy, reducing false alarms, aggregating alerts, and meeting the needs of production safety inspections, supervision, and rectifications.

IV. APPLICATION RESEARCH OF THE YOLOCY ALGORITHM FRAMEWORK IN THE INDUSTRIAL FIELD

A. *Application Architecture*

Based on the achievements of the enterprise's deep cultivation in the industrial field for many years, this paper integrates the algorithm framework based on YOLOCy into the visual AI product system to form a system architecture integrating data acquisition, algorithm construction, model training, service packaging, platform application, and industry field solutions. The application system architecture diagram is shown in Figure 4 below.

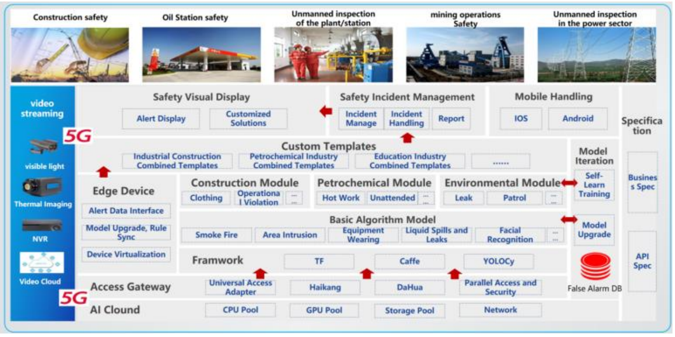

Figure 4. Application Architecture in the Industrial Field

As shown in Figure 4, in the application system architecture, the YOLOCy algorithm framework is applied to algorithm construction, foundational algorithm models, industry-specific algorithm models, as well as model upgrades and iterative management. It serves as the core technical module within the system architecture, functioning as the "engine" and "brain" of the entire application system architecture.

B. *Application of Visual AI in Industrial Production Safety Detection Scenarios*

In the process of application implementation, considering aspects such as application scope, universality, and representativeness comprehensively, this paper selects several general security scenarios in the industrial field such as smoke, open flames, labor protection wear, and boundary intrusion from numerous application scenarios. And algorithms mainly implemented by target detection of the YOLOCy framework are selected as typical scenarios for application research. The application effect of YOLOCy in the industrial field is shown in Figure 5 below.

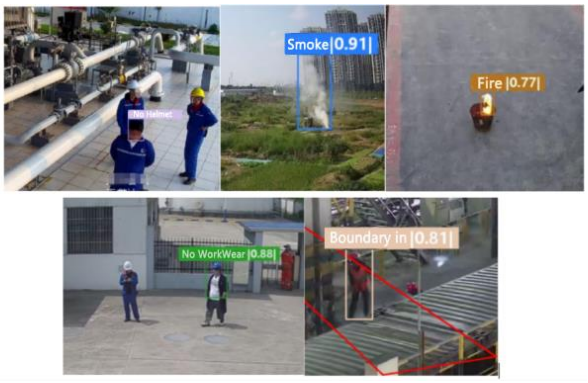

Figure 5. Application effect diagram in industrial field scenarios.

V. Experimental Results and Analysis

This paper has conducted a large number of experimental tests on the application effects of each optimization module of the detection network, the AuxBranch training method, and the alarm filter in the visual detection scenario under the YOLOCy algorithm framework, and obtained data support for demonstrating the optimization and improvement of the YOLOCy algorithm framework and the effectiveness of practical application in the industrial field. This paper uses the MS COCO2017[18] dataset as test data. Among them, in the training task, the default batch-size is 128, the SGD optimizer, a learning rate of 0.01, 100 epochs, and the CIOU loss function. The Copy-Paste enhancement is not enabled in the experiment. For inference latency testing, a 3080TI and TensorRT with batch-size = 1 are used. In the scene detection application experiment, the actual data from the project site of a petrochemical group is used for testing.

First, experiments were conducted on the various optimization items of the YOLOCy algorithm using the MS COCO2017 dataset, with the small model as the validation. The results are shown in Table III.

TABLE III. Experimental results of YOLOcy algorithm framework

Methed	mAP.5:.95	Forward (ms)
Yolov5-S	37.4	2
- Copy-Paste	35.8	2
+ SpaceMerge	36.4	1.7
+ RepC3	41.2	2.7
+ UniRepC3	42.8	3.2
+ Layer Prune	41.6	2.2
+AuxBranch Train	43.1	2.2
+ Self-KD	43.8	2.2
Yolov8-S	44.9	4

As shown in Table III, starting from YOLOv5-S without using Copy-Paste, every time an optimization item is added, the mAP is improved to varying degrees. Finally, using AuxBranch training achieves the optimal effect. The test result shows that the mAP is slightly lower than that of YOLOv8-S, but the inference speed is nearly twice as fast as that of YOLOv8-S. From the experimental results, it can be seen that under the condition of keeping the inference speed unchanged, the YOLOCy algorithm in this paper has a certain improvement in model accuracy compared to the YOLOv5 model, proving the effectiveness and feasibility of the optimization measures in this paper.

After verifying the effectiveness and feasibility of the YOLOCy algorithm optimization scheme, this paper conducts experiments on the application practice of the YOLOCy algorithm in the industrial field to demonstrate the effectiveness of the alarm filter module on the application effect. The actual dataset of a privatized project is used for testing, which contains 15,549 image data, including 3,521 flame targets, 3,832 smoke targets, 3,358 safety helmet targets, 4,241 work clothes targets, 5,246 pedestrian targets, and a total of 6,342 negative sample targets of various categories. It contains shooting pictures under different camera angles and distances and various lighting conditions in more than 1,000 scenes. The experimental results are shown in Table IV below.

As shown in Table IV, in the five detection scenarios, different filtering operators are used for filtering according to the actual situation. After the initial model detection, through the superimposed filtering of filtering operators, there is a relatively obvious improvement in accuracy. It should be specially noted that the YOLOCy detection model has been fine-tuned and trained with actual project data.

TABLE IV. EXPERIMENTAL RESULTS OF PROGRESSIVE ANALYZER ACCURACY

Application Scenarios	Model Detection	OverLapUnit	BlurUnit	OverExposuUnit	IomUnit	SequenceUnit	dynamicUnit
Flame Detection	0.806	-	0.817	-	-	0.939	0.951
Smoke Detection	0.783	-	0.795	0.823	-	0.921	0.942
Helmet Detection	0.903	0.921	0.924	0.944	0.958	-	-
Workwear Detection	0.934	0.939	0.943	0.951	0.953	-	-
Boundary Intrusion	0.974	-	0.976	0.979	-	-	-

From the experiment, it can be seen that the YOLOCy algorithm framework proposed in this paper, including the YOLOCy network, the AuxBranch auxiliary training method, and the alarm filter module, is practical and feasible in the application practice in the industrial field, and has achieved good application effects on the ground. In terms of the optimization of the detection algorithm, each optimization item has a good improvement in accuracy, achieving the effect of optimizing the detection accuracy while maintaining the detection speed. In terms of application practice, due to the complexity of the actual environment, the initial model detection effect of each target detection task in the actual application scenario is not good. After the correction of each filtering operator of the alarm filter, a very good application effect can be achieved, reflecting the effectiveness and practical value of the alarm filter module for practical applications.

VI. CONCLUSIONS

In recent years, the application research of visual AI in the industrial field has made remarkable progress, providing strong technical support for industrial intelligence, quality control, and safety production. In this paper, the algorithm network of the YOLO series target detection algorithm is improved. The SpaceMerge module, RepC3 module, and UniRepC3 module are proposed. The auxiliary training optimization scheme is introduced to improve the model detection accuracy while maintaining performance after model pruning. In addition, an alarm filtering function module at the application level is proposed to form the algorithm detection framework YOLOCy. Taking the actual project business data as the research content, the challenges and innovative solutions of this framework in the practical application in the industrial field are discussed. Of course, the application of visual AI in the industrial field still has huge development potential. With the continuous progress of deep learning technologies such as large models and multimodal and the continuous expansion of industrial application scenarios, visual AI will inject new impetus into the intelligent and automated development of the industrial field.

ACKNOWLEDGMENT

In the process of completing this paper, I am filled with countless emotions. First of all, I would like to sincerely thank my enterprise and leaders and colleagues. They have given me full support and help in daily work and created a good working environment and development space for me. Secondly, I also want to thank my family. During the days when I am busy with work and papers, they have given me understanding, encouragement and love. You are my strongest backing, allowing me to be fully engaged in work and study. Finally, I want to thank all those who have given me help and support in my career. This paper is not only a manifestation of my academic achievements, but also a witness to my growth in the enterprise. I will continue to work hard and contribute my own strength to the development of the enterprise and live up to everyone's expectations.

REFERENCES

[1] Dosovitskiy, A., Borovinsky, A., Kolesnikov, I., Gelfand, J., & Bahdanau, D. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929.

[2] Touvron, C., Dehghani, M., Denil, M., Belanger, F., Larsson, E., Agarwal, N., ... & Bengio, Y. (2021). Train Transformers Longer. arXiv preprint arXiv:2103.17232.

[3] Girshick R . Fast R-CNN[J]. Computer Science, 2015.

[4] Redmon J , Divvala S , Girshick R , et al. You Only Look Once: Unified, Real-Time Object Detection[C]// Computer Vision & Pattern Recognition. IEEE, 2016.

[5] Redmon J , Farhadi A . YOLO9000: Better, Faster, Stronger[C]// IEEE. IEEE, 2017:6517-6525.

[6] Redmon J , Farhadi A . YOLOv3: An Incremental Improvement[J]. arXiv e-prints, 2018.

[7] Chuyi Li, Lulu Li, Yifei Geng, Hongliang Jiang, Meng Cheng, Bo Zhang, Zaidan Ke, Xiaoming Xu, & Xiangxiang Chu. (2023). YOLOv6 v3.0: A Full-Scale Reloading.

[8] Chien-Yao Wang, Alexey Bochkovskiy, & Hong-Yuan Mark Liao. (2022). YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors.

[9] Chien-Yao Wang, I-Hau Yeh, & Hong-Yuan Mark Liao. (2024). YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information.

[10] Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, & Jian Sun. (2021). YOLOX: Exceeding YOLO Series in 2021.

[11] Tal Ridnik, Hussam Lawen, Asaf Noy, Emanuel Ben Baruch, Gilad Sharir, & Itamar Friedman. (2020). TResNet: High Performance GPU-Dedicated Architecture.

[12] Raja Sunkara, & Tie Luo. (2022). No More Strided Convolutions or Pooling: A New CNN Building Block for Low-Resolution Images and Small Objects.

[13] Xiaohan Ding, Xiangyu Zhang, Ningning Ma, Jungong Han, Guiguang Ding, & Jian Sun. (2021). RepVGG: Making VGG-style ConvNets Great Again.

[14] Xiaohan Ding, Yiyuan Zhang, Yixiao Ge, Sijie Zhao, Lin Song, Xiangyu Yue, Ying Shan:UniRepLKNet: A Universal Perception Large-Kernel ConvNet for Audio, Video, Point Cloud, Time-Series and Image Recognition. CoRR abs/2311.15599 (2023)

[15] Siming FU, Huanpeng Chu, Lu Yu, Bo Peng, Zheyang Li, Wenming Tan, Haoji Hu. AuxBranch: Binarization residual-aware network design via auxiliary branch search. Pattern Recognition, 136:109263, April 2023.

[16] Liang, T., Chu, X., Liu, Y., Wang, Y., Tang, Z., Chu, W., Chen, J., & Ling, H. (2022). CBNet: A Composite Backbone Network Architecture for Object Detection. IEEE Transactions on Image Processing, 31, 6893–6906.

[17] CHEN Tianqi, XU Bing, ZHANG Chiyuan, et al. Training deep nets with sublinear memory cost[J]. arXiv preprint arXiv:1604.06174, 2016.

[18] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays,Pietro Perona, Deva Ramanan, Piotr Dollar, and C Lawrence ´Zitnick. Microsoft coco: Common objects in context. In European conference on computer vision, pages 740–755.Springer, 2014.