Sensors and Materials, Vol. 36, No. # (2024) 1–20
MYU Tokyo

S & M ####

Flexible Temporal Correlation Learning for Human, Animal, and Interactor Detection in Videos

Yanjun Feng[1] and Jun Liu[2*]

[1]School of Information Science and Engineering, Shenyang Ligong University, Shenyang 110159, China
[2]School of Automation and Electrical Engineering, Shenyang Ligong University, Shenyang 110159, China

(Received April 8, 2023; accepted December 15, 2023)

Keywords: object detection, video understanding, attention, temporal learning

Video object detection is a key technology for detecting and tracking humans and animals in behavior-understanding tasks. Furthermore, detecting small-scale interactors involved in human activities is challenging. Exploiting the temporal context relationship is important for continuous understanding. Temporal object detection has been the subject of significant attention, but most commonly used detection methods fail to fully leverage the abundant temporal information in videos. In the paper, we propose a novel approach to detect humans and animals in videos, called attentional temporal You Only Look Once (ATYOLO), which exploits the attention mechanism and convolutional long short-term memory. We use the proposed attentional module to integrate a pyramidal feature hierarchy temporally and design a unique structure that includes a low-level temporal unit and a high-level unit for multiscale feature maps. We have developed an innovative temporal analysis group with a temporal attention mechanism tailored for background and scale suppression. This attentional group integrates attention-aware features over time. Extensive comparisons are conducted to evaluate the detection capability of the proposed approach, and its superiority has been confirmed. As a result, the developed ATYOLO achieves fast speed and overall competitive performance in video detection, including ImageNet Video (VID) and Stanford Drone Dataset (SDD).

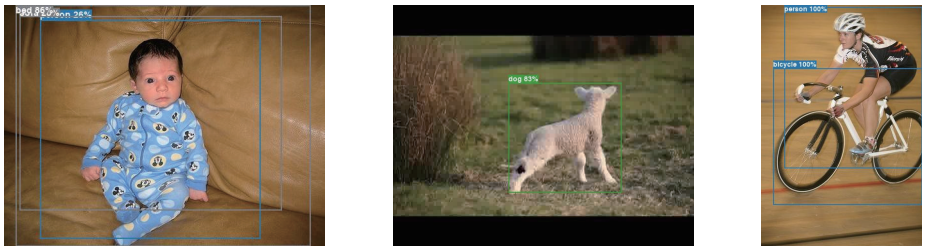
1. Introduction

To understand human activities, detection and tracking of human bodies and interacting objects are often required, especially in the case of understanding continuous behavior. Therefore, video detection technology is crucial, as shown in Fig. 1. Current methods aim to take advantage of the time–space relationships in video data. Traditional approaches rely heavily on manual design, resulting in low accuracy and limited robustness to noise sources. In recent years, deep learning solutions have been developed to overcome these limitations.[1,2] These methods can be classified on the basis of temporal information and feature aggregation.

Several methods have been proposed for video object detection, some of which only consider either local or global temporal information, while others use both. For example, relation

*Corresponding author: e-mail: lj_mail_sut@163.com

ISSN 0914-4935
https://myukk.org/



(a) (b) (c)

Fig. 1. (Color online) Detection in human, animal, and interactor understanding tasks. The proposed method can detect the baby, lamb, bicycle, and person effectively. (a) Human detection, (b) animal detection, and (c) interactor detection.

distillation networks use multistage reasoning to distill the relation between object proposals in videos.[3] Sequence Level Semantics Aggregation (SELSA) clusters and transforms the features of proposals extracted on different frames to generate more robust features for detection.[4] The object-guided external memory network uses object-guided external memory to store pixel and instance-level features for further global aggregation.[5] Memory-enhanced global-local aggregation (MEGA) utilizes global and local information for video object detection.[6] It integrates global information into local frames and uses a relation module to aggregate features of candidate proposals on global frames into that of local frames. Progressive sparse local attention (PSLA) establishes correspondence by propagating features in a local region with a gradually sparser stride according to the spatial information across frames.[7] Recursive feature updating and dense feature transforming based on PSLA were also proposed to model the temporal relationship and enhance the features.[8]

Video object detection presents a significant challenge owing to the degradation of features in video frames caused by camera jitter or fast motion. Applying detection algorithms designed for still images to video tasks is not optimal. However, videos contain valuable temporal information that can be utilized to detect objects that may appear in multiple frames within a specific time. In prior studies, the potential of this temporal information has been explored by using post-processing methods such as motion estimation and object tracking to assemble detection results from still-image detection on single frames. However, these methods do not operate end-to-end, and the weak detection results must be improved. Alternatively, attempts have been made to enhance the video detection performance by aggregating features using optical flow to model feature movement across frames and propagate temporal features to improve feature representation for detection. While this approach has significantly improved detection results, the lumping operation used to exploit temporal features must be simplified.

Hence, in this paper, we propose a novel flexible object detection method to detect and link objects across video frames. We introduce attentional temporal You Only Look Once (ATYOLO), which exploits the YOLO v5 architecture and integrates convolutional Long Short-Term Memory (LSTM) to incorporate temporal information effectively. We also develop a new structure, called the multilevel temporal unit, that enables the propagation of visual features across scales to enhance object detection accuracy. Furthermore, we address the background and

scale suppression challenge in multiscale feature maps by integrating an attention mechanism. We design an attentional module that can selectively focus on relevant visual features for object detection while suppressing irrelevant background information. The proposed ATYOLO significantly advances the field of object detection in videos by achieving high accuracy and efficiency in real-time applications. In addition to evaluating ATYOLO on the challenging ImageNet VID, we test it on small-sized object data collected using unmanned aerial vehicles (UAVs). Through experiments, we demonstrate that ATYOLO outperforms state-of-the-art methods in terms of accuracy and speed, making it a valuable tool for various applications, such as surveillance and gaming.

The work has contributed the following to object detection in videos.

(1) A novel structure and attentional module for the effective temporal propagation of pyramidal feature hierarchy has been proposed. The temporal attention mechanism incorporated in the framework has allowed for background and scale suppression.
(2) An attentional group with a low-level extractor has been employed to enable object correlation learning across frames. This allows for fast linking of detected objects and has improved the efficiency of the detector.
(3) The ATYOLO achieves improved results on ImageNet VID and SDD in terms of detection and tracking accuracy. These results demonstrate the effectiveness of improving state-of-the-art object detection in videos.

2. Related Work

2.1 Detecting humans, animals, and interactors in videos

Detection is a computer vision task that involves detecting objects in video data instead of conventional object detection in static images. It has played a significant role in developing autonomous driving and video surveillance applications. Video object detection is an active area of research, with various approaches being developed to address its associated challenges. Earlier attempts involved object detection on each image frame, leading to computational inefficiency and low accuracy.[9] However, more recent approaches use space and time information to reduce redundancy and improve detection efficiency and accuracy. Deep-learning-based models are more effective than conventional approaches for various computer vision, speech processing, and multimodality signal processing tasks.

Video object detection has numerous applications, including hand segmentation, human pose estimation, instance-level human parsing, and multiple people tracking. These applications draw on various approaches, including flow-based, LSTM-based, attention-based, and tracking-based methods.[10] Video object detection has significant value in numerous applications and is an active area of research. Various approaches have been developed to address the associated challenges, and deep-learning-based models are adequate for this task.[11]

2.2 Temporal information analysis

Recently, researchers have discussed various methods employed for video object detection, focusing on utilizing time–space relationships. Traditional methods heavily rely on manual design and suffer from low accuracy and lack of robustness to noise sources. The authors thus focused on LSTM-based solutions, which use convolutional LSTMs to process sequential data and select important information over a long duration. Offline and online LSTM-based solutions were discussed, with the former utilizing all frames in the video and the latter only using the current and previous frames.[12] A modified version of the convolutional LSTM was used with an image-based object detector to achieve good performance in model size and computational efficiency.[13] Then an improvement to this method that employs two feature extractors and a memory mechanism with a modified convolutional LSTM layer was discussed.[14] Finally, a recurrent causal method was proposed for online detection without succeeding frames, in which short-term and long-term temporal information is utilized to overcome challenges such as occlusion and motion blur.[15]

For example, a novel approach was proposed for pixelwise segmentation to locate foreground moving objects accurately. LSTM architecture was integrated into the encoder–decoder framework to represent the probability of each pixel being a foreground object. The LSTM network sequentially processed the input segmented video frames and learned to identify the moving objects by assigning weights to the relevant areas. The advantage of LSTM over RNNs is its ability to maintain long-term memory, thus ensuring temporal consistency across frame sequences. The perceptron network was employed to concentrate on the moving objects and consider their motion properties to determine the attention weight.[16] Detecting moving events in complex backgrounds is another challenging task in object detection, which Zhu *et al.* tackled by introducing Gaussian noise to simulate complicated backgrounds. Their approach involved using Mask R-CNN to localize the objects and VGG16 to extract features, followed by a bidirectional LSTM network that learns temporal information from past and future frames.[17] A weighted attention method was used to highlight the required features. The forward propagation method was used to feed previous frame features to the next node along with the input frame. In contrast, the background propagation was performed similarly to preserve the temporal information. The detected object output was obtained by averaging the features from both directions.[18]

2.3 Small-scale object detection

UAVs equipped with remote sensing equipment are gaining popularity in various fields, including security and surveillance, search and rescue, and sports analysis, owing to their high mobility, fast deployment, and enormous surveillance scope. UAV photography is a powerful supplement to satellite and airborne remote sensing. However, object detection in UAV images remains a core problem in computer vision. The small size and ambiguous boundaries of objects in UAV images, complex backgrounds, and changing illumination conditions pose significant challenges to accurate and efficient detection.

Traditional object detection methods in UAV images rely on hand-crafted features such as histogram of oriented gradient features, scale-invariant feature transform features, and Haar-like features, which are time-consuming and laborious to achieve the required robustness of feature representation. Recent advances in deep-learning-based methods, such as deep belief networks, convolutional neural networks (CNNs), generative adversarial networks, and deep transfer networks, show great promise.[19] However, these methods are computationally expensive in terms of time and network volume and have relatively low accuracy in detecting small objects. In addition, context is a vital factor for humans to recognize objects, and empirical studies in computer vision have shown that modeling spatial context can significantly improve algorithm performance. Therefore, modeling the spatial context can positively impact small object detection in UAV images. However, the current object detection methods are still a trade-off between speed and accuracy. The challenge of quickly and accurately detecting small objects in UAV images remains for real-time applications.

3. Method

As shown in Fig. 2, with the extended cross stage partial (CSP) darknet 53 as the backbone, we build a temporal architecture with four C3 blocks in the neck. The proposed ATYOLO is based on forward CNN and RNN, which generate pyramidal features for detection. The head is designed to be responsible for detecting the location and category of the object using the feature maps extracted from the backbone. In this process, the convolution operations are leveraged to predict the object's information with visual features. Then the multilevel temporal unit enables

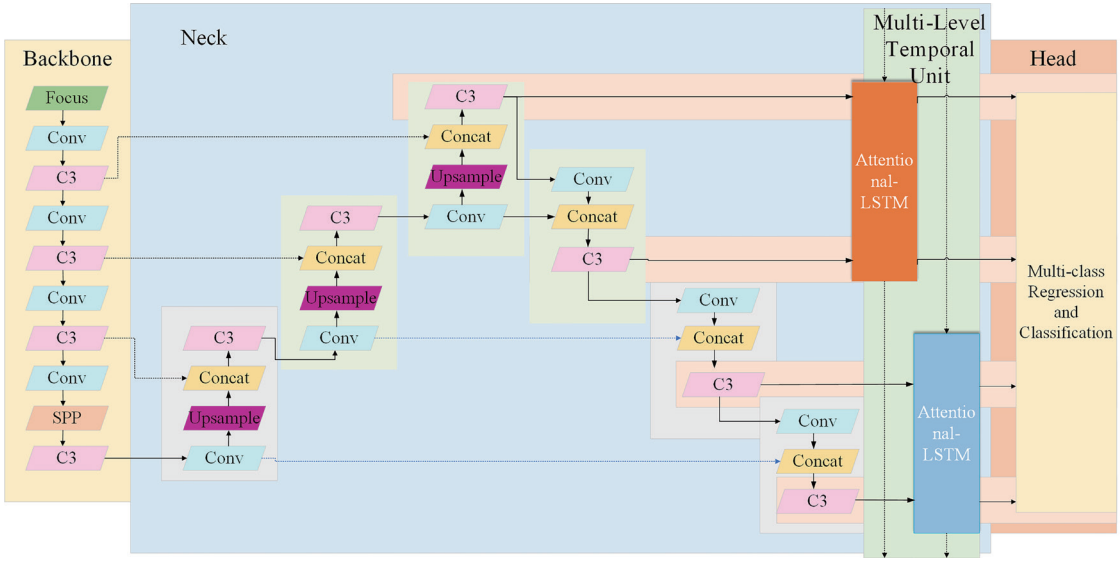


Fig. 2. (Color online) Framework of the proposed ATYOLO. The low-level features share an attentional LSTM, and high-level features do so in the neck. Next, the hidden states of convolutional LSTMs will be used for multibox regression and classification. Eventually, OTA is conducted for identification using the multiscale attention maps.

the propagation of visual features across time to enhance object detection accuracy. Finally, these bounding boxes and the category of objects on those boxes follow the nonmaximum suppression (NMS) to generate the final results. As for sequence learning, ATYOLO is equipped with multiscale feature-integration structures aimed at effectively producing temporal memory by using more helpful information.

3.1 Temporal correlation learning

Furthermore, we integrate the attentional mechanisms using the same two structures and integrated temporal features. Specifically, we divide the multiscale feature maps into two groups on the basis of their hierarchical relationships, as shown in Fig. 2. The module makes fuller use of the learning process since the low-level features (extracted by the first few convolution layers) contain more image details. In contrast, the high-level features (features extracted by the last layers of convolution) contain more semantic information in the process of visual feature extraction by convolutional computation. Consequently, low-level and high-level features need to share their respective time units to extract better features. Finally, we assign the two features extracted at the neck (shown in blue in Fig. 2) into one group and the next two extracted features (shown in green in Fig. 2) into another group; these groups share a one-time unit.

Not only that, even though only one frame of the target detection process is shown in Fig. 1, ATYOLO learns from all previous frames and uses the current pyramidal features and all previous memories to generate the current hidden state. This is a multilevel time unit, as shown in yellow in Fig. 2. In addition, the number of frames that need to be memorized is controlled by ATYOLO's forgetting gate.

3.2 Attentional module

Small target detection with a large-scale background is difficult in the detection task. Therefore, convolutional LSTM could be more efficient when dealing with background information, especially for multiscale feature maps. For example, if an object occupies too small a proportion of the whole image, this results in far fewer features associated with small objects than the background, which may cause false detection. In addition, the attention mechanism can selectively focus on relevant visual features for object detection while suppressing irrelevant background information. For this reason, we propose ATYOLO for background and scale suppression, where the temporal attention mechanism selects object-aware features for the convolutional LSTM. In turn, the convolutional LSTM provides temporal information to the attention mechanism to improve attention accuracy. As a temporal analysis unit, ATYOLO can be represented as

$$a_t = \mathrm{sigmoid}\left(W_a * \left[x, h_{t-1}\right]\right), \tag{1}$$

$$i_t = \mathrm{sigmoid}\left(W_i * \left[a_t \circ x, h_{t-1}\right] + b_i\right), \tag{2}$$

$$f_t = \text{sigmoid}\left(W_f * \left[a_t \circ x, h_{t-1}\right] + b_f\right), \tag{3}$$

$$o_t = \text{sigmoid}\left(W_o * \left[a_t \circ x, h_{t-1}\right] + b_o\right), \tag{4}$$

$$c_t = \text{sigmoid}\left(W_c * \left[a_t \circ x, h_{t-1}\right] + b_c\right), \tag{5}$$

$$s_t = \left(f_t \odot s_{t-1}\right) + \left(i_t \odot c_t\right), \tag{6}$$

$$h_t = o_t \odot \tanh(s_t), \tag{7}$$

where * stands for convolution, $\circ$ represents the multiplication of single channel mapping with each channel in a multichannel feature mapping one by one, and $\odot$ represents the element-by-element multiplication. a_t, i_t, f_t, o_t, c_t, s_t, and h_t in the above equation represent the attention map, input gate, forget gate, output gate, LSTM'S incoming information, memory, and the hidden state, respectively.

As shown in Fig. 3, ATYOLO is designed with CNN and RNN. The current feature map x and previous hidden state h_{t-1} serve as the input of the attention module. After a three-layer convolution, a one-channel attention a_t containing pixelwise positions for object-aware features is generated and will be used to select useful features in ATYOLO. Note that each element of the graph takes a value within [0,1] in order to describe the object mass more effectively. For feature selection, each channel of the current feature map multiplies this attention map pixel-by-pixel, and the attention-aware feature ($a \bullet x$) can be obtained. The attention-aware feature and the previous hidden state are concatenated as the input of the convolutional LSTM. Different from the traditional LSTM, gates (i, f, o) and incoming information (c) will be computed by the convolution operation. Subsequently, several frames are memorized by the door control, the temporal memory(s) is updated, and the current hidden state is generated for regression.

3.3 Backbone

3.3.1 YOLO families

Object detection is a critical computer vision task, and YOLO is a widely used algorithm known for its fast and accurate object detection capabilities. YOLO was first introduced by Redmon *et al.* in 2015.[20] Since then, several subsequent versions of YOLO have been developed, including YOLO V2, YOLO V3, YOLO V4, and YOLO V5, along with a few revised limited versions like YOLO-LITE.[20–22] In this subsection, we aim to comprehensively compare the five main YOLO versions in terms of their conceptual designs and implementations, focusing on their primary motivations, feature development, limitations, and relationships. This comparison is particularly relevant as YOLO versions continue to evolve, making it essential to understand their similarities and differences.

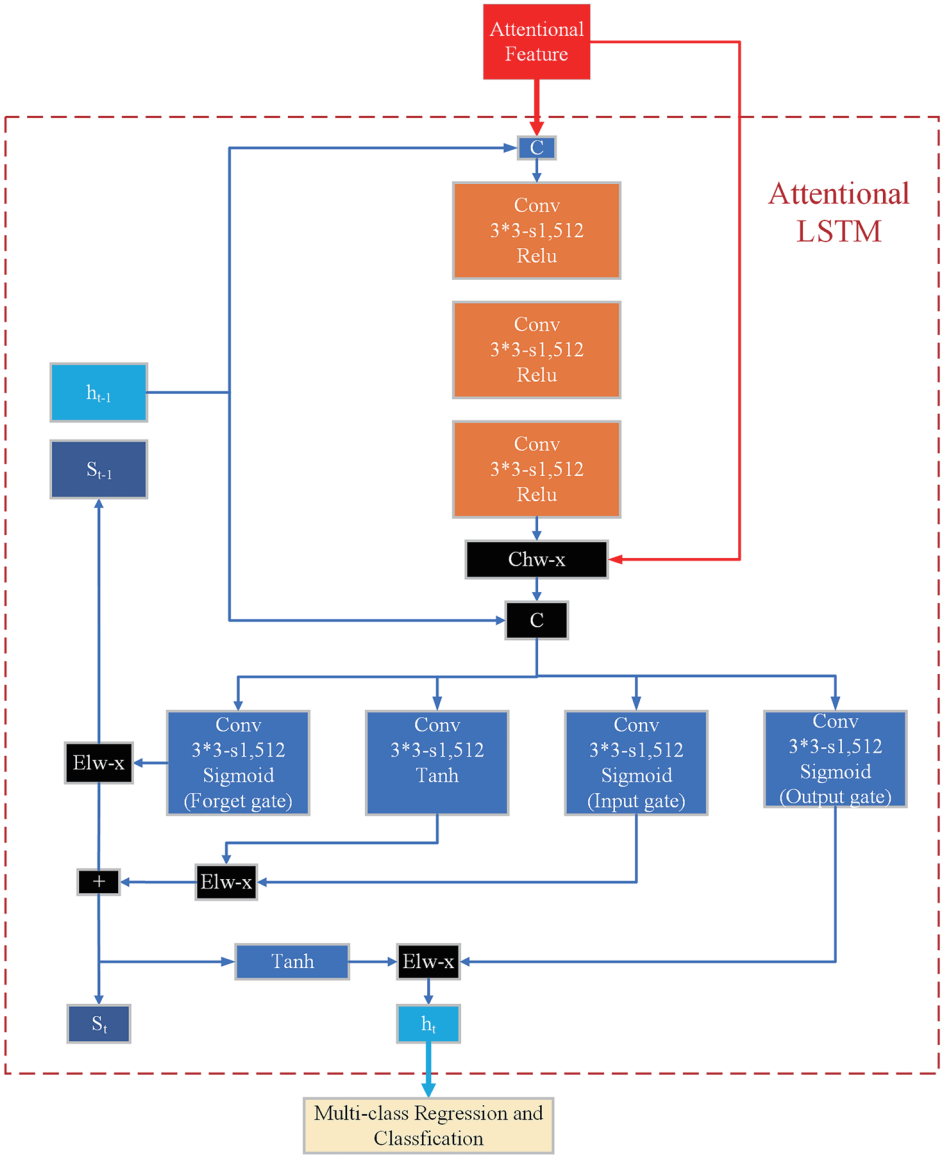


Fig. 3. (Color online) Architecture of the proposed attentional LSTM. "c" denotes the concatenation; "Chw-x" and "Elw-x" represent the channelwise and elementwise multiplications, respectively.

Object detectors typically comprise a backbone network and a head for predicting object categories and bounding boxes, with a neck layer in between to aggregate feature maps. The backbone network is usually pretrained on the ImageNet dataset, and YOLO V4 has been used in extensive experiments on the neck layer. YOLO V1 uses a backbone similar to GoogleNet, with 24 convolutional layers and two fully connected layers, and was validated on the Pascal visual object classes dataset. However, YOLO V1 was not suitable for recognizing small and densely packed objects, as it only predicts two bounding boxes per grid in a 7 × 7 grid. YOLO V2 is based on V1 and uses the VGG network to create a new backbone network called Darknet-19. In

YOLO V2, anchor boxes were introduced to replace the fully connected layers, which resulted in more accurate localization. The input was also resized to 416 × 416 and a 13 × 13 feature map with an odd dimension and exactly one center was obtained, making it easier to predict objects with their center points falling into that position. YOLO V3 uses Darknet-53 as the backbone network, where the residual structure of ResNet was borrowed to deepen the network structure while preventing gradient explosion. It also uses tensor concatenation to extract more information by concatenating the middle layer of Darknet-53 with a later layer after upsampling. As a result, YOLO V3 has more than ten times the number of predicted boxes as YOLO V2, and they are performed at different scales, greatly improving detection accuracy, particularly for small objects. YOLO V4 amalgamates various improvement methods after V3 and is divided into free and discounted packages. The former indicates modules that improve training without affecting inference speed, and the latter indicates modules that have little impact on inference time but have higher performance returns. For example, the local CSP structure used in the backbone network maintains high inference speed while retaining high accuracy. At the same time, YOLO V4 is more suitable for training on a single GPU. YOLO V5 has a similar basic structure to YOLO V4 but builds models based on different channel scales, from small to large, depending on the model. YOLOX is based on YOLO V3 and YOLO V5 and uses CSPNet, the sigmoid-weighted linear unit, and a path aggregation network. YOLO is a milestone algorithm in single-stage detection.

3.3.2 YOLOv5-based backbone

YOLO V5 boasts multiple network architectures, making it a highly versatile and lightweight option that matches the accuracy of its predecessor, YOLO V4. Despite criticism for being less innovative, YOLO V5 has numerous advantages, such as the PyTorch framework, which is user-friendly and easy to use. Its code is also easy to read and includes various computer vision technologies that facilitate learning and reference. One of the most significant advantages of YOLO V5 is its simplified training process. A data loader can enhance training data in three ways: scaling, color space adjustment, and mosaic enhancement. Mosaic enhancement has been beneficial in accurately detecting small objects, a persistent issue in model training. Although the naming of YOLO V5 has been controversial, and its implementation is still evolving, it currently provides greater flexibility in controlling model size, applying the Hardswish activation function, and utilizing data enhancement techniques. In summary, YOLO V5 is a promising solution for real-time object detection and offers a convenient platform for further research and development in this field.

We adopt the YOLO V5 framework for object detection, which comprises three main components: the backbone, neck, and prediction layers. This framework is widely used among deep learning enthusiasts and offers different versions tailored to various applications. In our method, the backbone component aggregates different image granularities and forms image features. We employ a CNN with a focus layer to enrich the training dataset and enhance the model's robustness, addressing the problem of repeated gradients in large convolutional network structures.

The neck component generates a feature pyramid and transfers image features to the prediction layer. We build upon the Mask R-CNN and feature pyramid network frameworks, optimizing information dissemination and enhancing the propagation of low-level features. Adaptive feature pooling is employed to restore damaged information paths and prevent arbitrary allocation.

Lastly, the prediction component conducts the final detection by applying an anchor box to the output feature map, generating an output vector with category probability, confidence score, and a bounding box. The loss function uses Generalized Intersection over Union (GIOU) loss, while the confidence loss and category loss employ the binary cross-entropy loss function. This configuration enables the YOLO V5 framework to detect and efficiently classify objects in the given input images.

3.4 Training

The prediction module performs the final detection in the head. An anchor box is applied to the output feature map to generate an output vector with category probability and a bounding box. On the anchor, ATYOLO uses cross-grid matching rules to distinguish the positive and negative samples of the anchor. The loss function uses GIOU loss, a temporal correlation loss $\mathcal{L}_{tc}$, and an attention loss $\mathcal{L}_{at}$, as shown in

$$\mathcal{L} = \alpha GIOU + \beta \mathcal{L}_{tc} + \gamma \mathcal{L}_{at}, \tag{8}$$

where α, β, and γ are the tradeoff parameters. Then, we train ATYOLO in three steps. In our study, we performed a grid search over the above hyperparameters where the full objective was employed for parameter fine-tuning over ten epochs. Guided by prior research,[23] we considered numerical ranges for $\alpha \in \{0.5, 1, 1.5\}$, $\beta \in \{0.5, 1, 2\}$, and $\gamma \in \{0.5, 1, 1.5\}$. For the final model selection, we opted for $\alpha = 1$, $\beta = 1$, and $\gamma = 0.5$, which yielded the best results throughout the experiments, as assessed by the mean average precision (mAP) metric.

GIOU loss: Compared with the intersection over union (IoU), the GIoU focuses on overlapping areas and considers other non-overlapping areas, which can better reflect the degree of coincidence. The calculation formulas of IoU and GIoU are as follows.

$$IoU = \frac{|A \cap B|}{|A \cup B|} \tag{9}$$

$$GIoU = IoU - \frac{|C \setminus (A \cap B)|}{|C|} \tag{10}$$

In Eqs. (9) and (10), A represents the area of the prediction box, B represents the area of the target box, and C represents the smallest area that includes A and B in a closed shape.

3.4.1 Attention loss

The generation of attention maps is supervised using cross-entropy. At first, we construct the ground truth attention map A_m, in which elements in the ground truth boxes are equal to 1 and others are equal to 0. There are four feature maps for multibox prediction that generate multiscale attention maps $A^0_{p_{sc}}$. Therefore, each $A^0_{p_{sc}}$. is first unified to the same resolution as the input image through bilinear upsampling operation, followed by the generation of $A^{up}_{p_{sc}}$. Each upsampled attention map can generate a scale-related attention loss with cross-entropy, and we add up four scale losses as the final attention loss. Then, $\mathcal{L}_{at}$ can be given as

$$\mathcal{L}_{at} = \sum_{c=1}^{4} \mu\left(-A^{up}_{p_{sc}} \log\left(A_m\right) - \left(1 - A^{up}_{p_{sc}}\right) \log\left(1 - A_m\right) \right), \tag{11}$$

where μ averages all elements in a matrix.

3.4.2 Temporal correlation loss

Pixel-level variations can greatly affect the detection results, so there are always large fluctuations when using static detection methods to detect targets in videos. Therefore, toward the temporal consistency of the video, a correlation loss should be developed for sequence training. For this purpose, we encourage ATYOLO to produce similar global classification results for consecutive frames. We first compute the first k high predicted scores for each class after NMS and then sum them to generate a class-distinguished score list denoted as (*sl*). The score list should maintain small fluctuations in consecutive frames. Thus, $\mathcal{L}_{tc}$ can be obtained as

$$\mathcal{L}_{tc} = \left(\sum_{t=1}^{len} sl_t - sl_{ave} \right) / len \ , \tag{12}$$

where sl_t is the score list at time step t, sl_{ave} denotes the mean score list among $sl_{1:t-1}$, and *len* represents the sequence length. It should be noted that the temporal correlation loss works in a self-supervised manner. That is, there is no incoming ground truth label when computing $\mathcal{L}_{tc}$.

4. Experimental Results and Discussion

4.1 Datasets

ImageNet VID is currently the most extensive dataset for temporal object detection.[24] The objective of our task was to detect 30 different classes of targets across consecutive frames, including persons, animals, and interactors. The training set contained 4000 videos totaling 1181113 frames, while the validation set consisted of 555 videos with 176126 frames. To facilitate training, we utilized the ImageNet Detection (DET) dataset, consisting of 200 categories, with the 30 VID classes being a subset. We trained the YOLO V5 model using both VID and DET

datasets but only used data from the 30 VID classes. The large number of frames in the VID training set made it difficult to train a network using all frames directly. Furthermore, the data for each category were imbalanced, with some videos having over 1000 frames while others had only a few. We followed the methods proposed in Ref. 25 to address these challenges. Specifically, we sampled up to 2000 images per class from DET and selected 10 frames from each VID video to train the YOLO V5 in the first step. We used all of the VID training videos in the subsequent two training steps.

SDD is a comprehensive dataset that includes images and videos of objects from various classes in motion and interacting within a real-world university campus. The dataset consists of six classes and eight different scenes. However, because of limitations in the information provided by the ortho-image, we concentrate on a subset of the dataset with four scenes, namely, the bookstore, *hyang*, circle, and little scenes. These scenes capture more objects and are more suitable for object detection tasks. However, the original dataset suffers from a severe imbalance in the distribution of object classes, with some classes being much more abundant than others. To mitigate this issue, we divide the six classes into three groups based on the object's appearance and speed of movement: pedestrians, bikers, and cars. This approach enables a more balanced representation of the object classes, improving the dataset's overall utility. The dataset comprises 69673 images for training and validation and 53224 for testing. However, SDD presents significant challenges for object detection tasks, mainly because of the small size of the three types of objects. Specifically, these objects each occupy less than 0.2% of the image size, with a significant proportion falling within the 0.1–0.15% size range. These statistics highlight the difficulty of accurately detecting and classifying these small objects, making SDD an ideal benchmark for evaluating object detection algorithms.

Following the previous work, we evaluated the model's performance using average precision (AP).[26] In scientific literature, mAP is defined as the average of AP values calculated for each individual category. Since computing the integral can be relatively challenging, an interpolation method is introduced to calculate AP. The interpolation formula of mAP is as follows.

$$mAP = \sum_c \sum_n \left(R_n - R_{n-1}\right) \cdot P_n \,/\, Num_c \tag{13}$$

Here, R_n represents the recall values, P_n denotes the interpolated precision values, Num_c is the number of classes, and c and n represent different interpolation points and classes, respectively.

4.2 Experimental setting

In this study, we conducted experiments on object detection in videos using an improved YOLO-V5-based model trained end-to-end with a stochastic gradient descent. The experiments were performed on a system with a GeForce RTX 3080ti GPU and 12 GB of video memory running on the Windows 10 operating system. The PyTorch framework was used, and Python was chosen as the development language, with the necessary libraries such as CUDA10.0 and open source computer vision library installed.

In the context of model initialization, we employed He initialization for the CNN layers preceding the multilevel temporal unit, as well as the attention module. This initialization takes into account the properties of the rectified linear unit activation function when setting the standard deviation. For the other layers within the AT-LSTM, we utilized Xavier initialization. This approach aids in the propagation of information across the temporal sequences, mitigating issues related to gradient vanishing or exploding. Furthermore, both the memory state and hidden state of the LSTM were initialized as zero vectors.

In the process of model training, we conducted fine-tuning of hyperparameters in consideration of the training dynamics. To maximize training efficiency, we configured a batch size of 8 and initialized the learning rate at 0.0003. Additionally, to mitigate potential underfitting issues, we set the weight decay regularization coefficient to 0.0005 and introduced two data augmentation techniques: random Hue shift and random saturation adjustment. Given the relatively stable performance of the model across different training epochs, we set the dynamic factor to 0.8. Finally, on the basis of the convergence behavior of the model, we established the training duration at 150 epochs. After completing the training, we saved the optimal detection model file and evaluated the performance using the verification dataset.

4.3 Comparison on ImageNet VID

Table 1 shows the performances of the proposed method and other state-of-the-art models on the ImageNet VID dataset. The results show that ATYOLO outperforms the existing methods when utilizing the YOLO v5 backbone, achieving an outstanding mAP of 81.77%. Moreover, when using the attentional component, the proposed model surpasses the baseline models, YOLO v5 and temporal single-shot detector (TSSD),[23] by 5.95 and 16.37 points, respectively, in terms of mAP. TSSD[23] is the first batch of attempts to integrate attention modules into traditional detectors to improve detection accuracy. It is essential to highlight that the attentional

Table 1
Performances of various methods on ImageNet VID.

Method	Backbone	mAP
TSSD[23]	VGG-16	65.4
DFF[27]	ResNet-50	70.3
YOLO v3[28]	DarkNet	72.27
FGFA-1[29]	ResNet-50	74
FGFA-2[29]	ResNet-101	76.3
D&T-1[30]	ResNet-50	76.5
RDN[3]	ResNet 101	76.7
MEGA[8]	ResNet-50	77.3
SELSA[6]	ResNet-50	78.4
D&T-2[30]	ResNet-101	79.8
PLSA[9]	ResNet-101	80
STMN[31]	ResNet-101	80.5
YOLO v5	DarkNet	75.82
ATYOLO	DarkNet	81.77

module is designed as an easy-to-integrate component, making it highly adaptable and compatible with any video object detection method to improve the performance. Although the experimental results demonstrate that incorporating our module into YOLO v5 achieves impressive performance gains, integrating our module into more advanced methods will result in even more significant improvements and potentially lead to state-of-the-art results. This provides a practical reference for future research on the YOLO family.

4.4 Comparison on SDD

Table 2 illustrates the effectiveness of the proposed attentional approach in enhancing the performance of the small-scale object detection algorithm for video object detection tasks. The approach incurs an increase in mAP, showing an additional 1.83 and 1.24% compared with feature fusion and scaling-based single shot detector (FS-SSD) 512 and FS-SSD+ spatial context analysis (SCA),[37] respectively, and because it processes the information of the global-local frames, it introduces additional motion information, resulting in a significant 2.4% improvement in "Biker" (AP) compared with the leading detector (FS-SSD+SCA[37]). This improvement is notably better than that achieved by FS-SSD-512-SCA,[37] which showed an increase of 0.49%.

Furthermore, the results in Table 2 demonstrate that the proposed algorithm outperforms the base algorithm by 1.51, 0.82, and 2.4% in the three challenging objects. This highlights the advantage of ATYOLO, which exploits global–local information in the whole processing stage and exploits the similarity of temporal information between adjacent frames, particularly for detecting small-size objects. Therefore, the proposed ATYOLO holds great potential in improving the performance of small-scale video object detection tasks.

4.5 Qualitative analysis

The proposed model demonstrates excellent detection results in real-world scenarios, including moving cars and multi-angle aircraft. As network dimensions increase, conventional network models naturally reduce their receptive fields, leading to the loss of crucial fine-grained

Table 2
Performances of various methods on SDD.

Methods	Backbone	Car	Pedestrian	Biker	mAP
Faster R-CNN(32)	VGG-16	58.57	67.08	53.24	59.63
R-FCN(33)	ResNet	61.57	67.42	54.92	61.3
DSSD(34)	ResNet	62.75	71.69	54.8	63.08
FSSD(35)	VGG-16	65.44	71.98	55.14	64.19
SSD 512(36)	VGG-16	60.55	67.54	52.89	60.33
SSD 300(36)	VGG-16	57.26	66.58	50.89	58.24
YOLO v3(28)	DarkNet	61.83	73.52	51.33	62.23
FS-SSD 512(37)	VGG-16	66.49	73.08	57.95	65.84
FS-SSD+SCA(37)	VGG-16	66.72	74.1	58.44	66.43
YOLO v5	DarkNet	63.78	73.84	55.39	64.34
ATYOLO	DarkNet	68.23	74.92	60.84	67.67

details. Moreover, the temporal correlations between frames gradually diminish as frames progress. Hence, traditional network models exhibit noticeable limitations in handling small target objects. The introduced model exploits attention mechanisms and long sequence modeling, to effectively harness both global and local information. This modeling approach enhances the expressive power of objects, particularly in the detection tasks of moving targets and small-sized objects. Simultaneously, the attention mechanism assigns greater weight to objects, thus overcoming the drawback of information loss.

Despite variations in the scale, perspective, and temporal order of video frames, all objects remain within the model's purview. Quantitative and qualitative results substantiate the crucial role played by attention mechanisms and temporal modeling in enhancing the precision of real-world object detection. These findings underscore the significance of attention mechanisms and temporal modeling in improving the accuracy of object detection in practical scenarios.

The comparative results of object detection are illustrated in Fig. 4 and indicate that our model achieves optimal performance, particularly under challenging conditions. The hidden variables incorporating information from previous frames, along with the attention maps they contribute to, enable the model to capture heavily occluded objects (tigers) or objects blurred due to motion (chickens). The attention maps of the model aid in enhancing the representation of

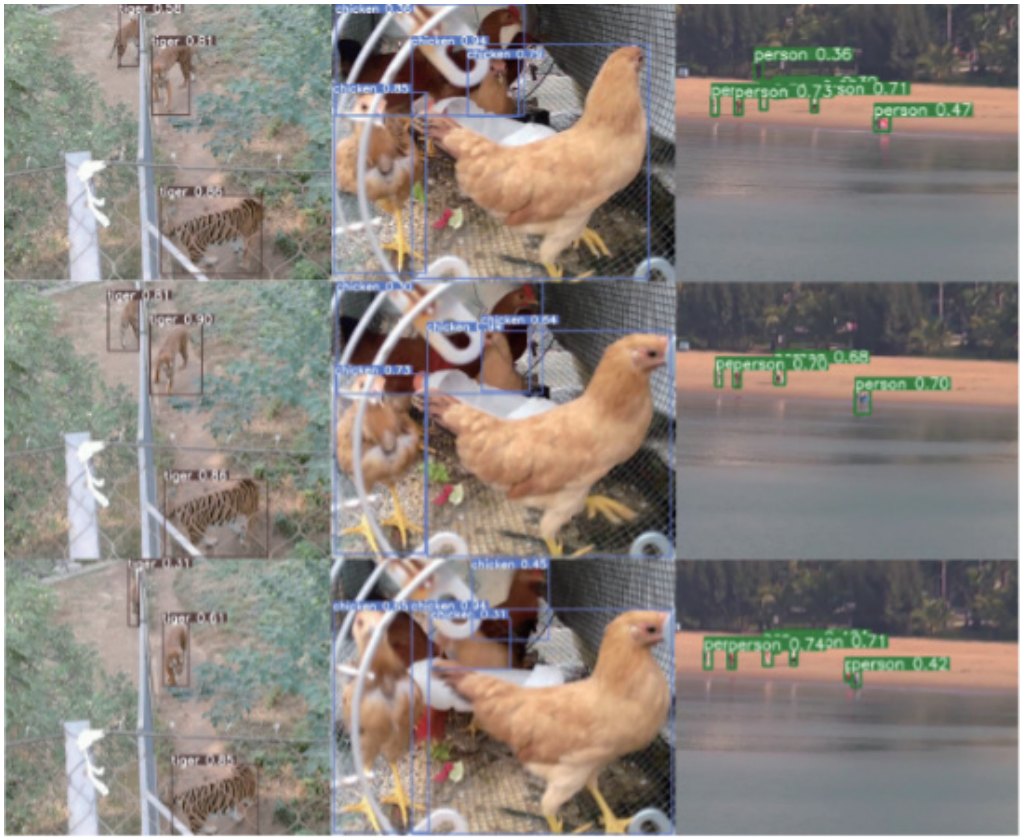


Fig. 4. (Color online) Qualitative results of multiscale object detection in videos.

detected objects, coupled with the feature pyramid, allowing the model to effectively detect small-scale targets (humans). These findings underscore the critical importance of incorporating attention mechanisms to enhance object detection performance.

4.6 Ablation study

First, we aim to investigate how the number of attention blocks affects the performance of the scale, temporal, and task-aware attention mechanisms. To accomplish this, we stack various numbers of attention blocks and evaluate their impact.

We integrate two attention modules to form low–high feature extraction (ATYOLO) and integrate three modules to form low–medium–high feature extraction (ATYOLO-2).

The experimental results, as shown in Table 3, indicate that the performance of the detector decreases as the number of attention blocks increase. Surprisingly, we achieve the highest performance using only two attention blocks. One plausible explanation for the observed decrease in performance with an increasing number of attention blocks is overfitting of the training data, particularly in the case of ImageNet VID. This dataset has a limited number of objects per frame, which may cause the model to memorize the training data instead of generalizing it to unseen examples. From the findings, we suggest using two attention blocks to achieve the best performance when employing the scale, temporal, and task-aware attention mechanisms. These results can be valuable for developing efficient attention-based models for various computer vision applications.

We carry out experiments to evaluate the performance of ATYOLO on ImageNet VID. The results presented in the above subsection demonstrate that ATYOLO, which uses the YOLO v5 backbone, achieves an mAP of 81.77%. The use of a shallower architecture in ATYOLO makes it less computationally complex than deeper architectures while still achieving comparable accuracy. As shown in Table 4, the results suggest that the proposed ATYOLO is a promising approach for object detection tasks and outperforms existing state-of-the-art methods in terms of

Table 3
Performance with various numbers of attention blocks.

Method	Parameters	mAP (%)
YOLO v5	6.9M	75.82
ATYOLO	8.37M	81.77
ATYOLO-2	8.52M	78.62

Table 4
Accuracy and computation costs.

Method	Parameters	mAP (%)	FLOPS
TSSD	4.9M	65.4	8.5B
YOLO v3	58.65M	72.27	115.6B
YOLO v4	60.94M	73.24	117.4B
YOLO v5	6.9M	75.82	12.7B
ATYOLO	8.37M	81.77	14.2B

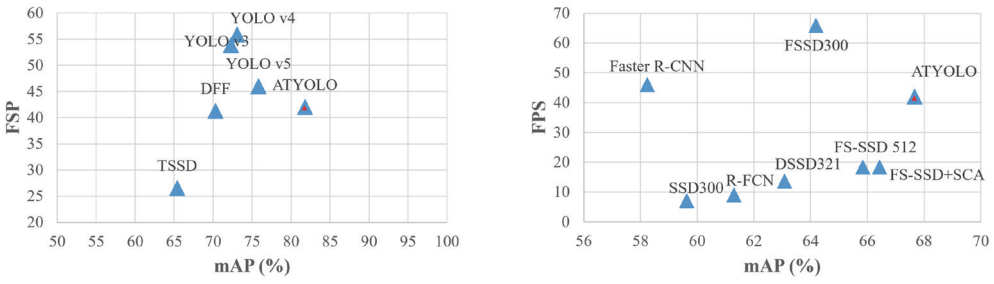
accuracy and computational efficiency. ATYOLO has 2.47 M more parameters than YOLO v5 but mAP is significantly improved by 5.95%.

Furthermore, we compared the performance of our proposed detector with those of other state-of-the-art detectors across two diverse benchmark datasets, as shown in Fig. 5. Impressively, ATYOLO distinguishes itself by achieving a substantial mAP of 81.77% on ImageNet VID, outperforming competing models such as YOLO V3, YOLO V4, and YOLO V5 by an appreciable margin of 5.95 mAP points. This noteworthy boost in detection accuracy is primarily attributed to the incorporation of LSTM for temporal modeling and attention mechanisms for background suppression and scale normalization. Furthermore, ATYOLO maintains its practicality with a real-time processing speed of 42 frames per second, making it an excellent choice for real-time applications, including autonomous driving and surveillance. Its competitive mAP of 67.67% on SDD underscores its adaptability to varying scenarios. These findings collectively emphasize the pivotal role played by LSTM and attention mechanisms in enhancing detection precision and solidify ATYOLO's potential in addressing real-world object detection challenges where a balance between accuracy and real-time responsiveness is paramount.

4.7 Discussion

In this study, we aimed to address the challenges associated with still-image object detectors and video object detection. Specifically, we focused on the degradation in the appearance of objects in videos owing to various factors such as motion blur, defocus, occlusion, illumination, scale, and spatial variance. In addition, we investigated the problem of object detection in UAV images, which is a crucial issue in computer vision with numerous real-world applications.

To overcome these challenges, we proposed a novel approach to effectively leverage temporal information across video frames. We disentangled feature representation by learning scale-aware, temporal-aware, and task-aware features and incorporated them with attention mechanisms. This approach allowed us to handle high intraclass similarity and temporal, scale, and task variance among video frames more effectively, leading to improved performance in



(a) (b)

Fig. 5. (Color online) Accuracy and speed on (a) ImageNet VID and (b) SDD.

object detection. The experimental results of using the unified attention head approach demonstrated consistent and significant improvements in performance. Overall, this study has contributed to developing more efficient and effective methods for object detection in challenging environments, including UAV images.

5. Conclusions

Aiming at detecting humans, animals, and interactors, in this study, we propose a new attentional detection method called ATYOLO, which exploits global-local information to measure spatiotemporal relations between adjacent frames, boosting behavior understanding. The attentional groups ensure that the object shares the same temporal information and combines previous frame temporal information and classification confidence to link detection boxes. We also utilize the dynamic mean rescore to calculate the classification confidence of the current frame. The results of the experiments demonstrate that the attentional framework, which uses YOLO V5 as the base detection network, performs better than other detectors and substantially improves the accuracy of video object detection without any significant increase in the amount of computation. Moreover, ATYOLO balances speed and accuracy, making it ideal for real-life applications such as video surveillance. Looking ahead, improving detection accuracy while maintaining real-time performance is the future direction of video object detection algorithms. In this research, we explored this direction, and we hope our findings will inspire further detection studies in this area.

Acknowledgments

This work was supported by the General Science Research Project from the Liaoning Provincial Department of Education under Grant LJKMZ20220610.

References

1 Z. Zou, K. Chen, Z. Shi, Y. Guo, and J. Ye: Proc. IEEE **111** (2023) 257–276.
2 G. Cheng, X. Yuan, X. Yao, K. Yan, Q. Zeng, and J. Han: arXiv:2207.14096 (2022).
3 J. Deng, Y. Pan, T. Yao, W. Zhou, H. Li, and T. Mei: 2019 IEEE/CVF Int. Conf. Computer Vision (ICCV, Seoul, Korea (South), 2019) 7022–7031. https://doi.org/10.1109/ICCV.2019.00712
4 J. Lee, P. Wang, R. Xu, V. Dasari, N. Weston, Y. Li, and S. Chaterji: Proc. 5th Int. Workshop Embedded and Mobile Deep Learning (2021) 19–24.
5 L. Zheng, T. Zhou, and R. Jiang: 2021 4th Int. Conf. Algorithms, Computing and Artificial Intelligence (2021) 1–6.
6 Y. Chen, Y. Cao, and H. Hu: Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (2020) 10337–10346.
7 H. Deng, Y. Hua, T. Song, Z. Zhang, Z. Xue, R. Ma, and H. Guan: 2019 IEEE/CVF Int. Conf. Computer Vision, (2019) 6677–6686. https://doi.org/10.1109/ICCV.2019.00678
8 C. Guo, B. Fan, J. Gu, Q. Zhang, S. Xiang, V. Prinet, and C. Pan: 2019 IEEE/CVF Int. Conf. Computer Vision, (2019) 3908–3917. https://doi.org/10.1109/ICCV.2019.00401
9 Z. Li, Y. Wang, N. Zhang, Y. Zhang, Z. Zhao, D. Xu, G. Ben, and Y. Gao: Remote Sens. **14** (2022) 2385. https://doi.org/10.3390/rs14102385

10 Z. Teng, Y. Duan, Y. Liu, B. Zhang, and J. Fan: IEEE Trans. Geosci. Remote Sens. **60** (2022) 1. https://doi.org/10.1109/TGRS.2021.3064840
11 H. Zhu, H. Wei, B. Li, X. Yuan, and N. Kehtarnavaz: Appl. Sci. **10** (2020) 7834. https://doi.org/10.3390/app10217834
12 J. Yu, H. Gao, Y. Chen, D. Zhou, J. Liu, and Z. Ju: IEEE Trans. Hum.-Mach. Syst. **52** (2022) 784. https://doi.org/10.1109/THMS.2022.3144951
13 X. Wang, X. Xie, and J. Lai: Pattern Recognition and Computer Vision: First Chinese Conf. (PRCV 2018, Guangzhou, China, November 23–26, 2018) Proc. Part II. Cham (Springer International Publishing, 2018) 99–109.
14 J. Yu, H. Gao, Y. Chen, D. Zhou, J. Liu, and Z. Ju: IEEE Trans. Hum.-Mach. Syst. **52** (2022) 784. https://doi.org/10.1109/THMS.2022.3144951
15 C.-W. Chang, C.-Y. Chang, and Y.-Y Lin: Multimed. Tools Appl **81** (2022) 11825. https://doi.org/10.1007/s11042-021-11887-9
16 X. Chen, J. Yu, and Z. Wu: IEEE Trans. Cybernetics **50** (2020) 2674. https://doi.org/10.1109/TCYB.2019.2894261.
17 H. Zhu, H. Wei, B. Li, X. Yuan, and N. Kehtarnavaz: Sensors **20** (2020) 3591. https://doi.org/10.3390/s20123591
18 C. Yuan Qiang, D. Du, L. Zhang, L. Wen, W. Wang, Y. Wu, and S. Lyu: Proc. 28th ACM Int. Conf. Multimedia (2020) 709–717.
19 X. Wu, W. Li, D. Hong, R. Tao, and Q. Du: IEEE Geosci. Remote Sens. Mag. **10** (2022) 91. https://doi.org/10.1109/MGRS.2021.3115137.
20 J. Redmon, S. Divvala, R. Girshick, and A. Farhadi: Proc. IEEE Conf. Computer Vision and Pattern Recognition (2016) 779–788.
21 J. Redmon and A. Farhadi: arXiv preprint arXiv:1804.02767 (2018).
22 C. Chen, Z. Zheng, T. Xu, S. Guo, S. Feng, W. Yao, and Y. Lan: Drones **7** (2023) 190. https://doi.org/10.3390/drones7030190
23 X. Chen, J. Yu, and Z. Wu: IEEE Trans. Cybern. **20** (2020) 2674. https://doi.org/10.1109/TCYB.2019.2894261
24 O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, and L. Fei-Fei: Int J. Comput. Vis. **115** (2015) 211. https://doi.org/10.1007/s11263-015-0816-y
25 C. Feichtenhofer, A. Pinz, and A. Zisserman: 2017 IEEE Int. Conf. Computer Vision (ICCV, Venice, Italy, 2017) 3057–3065, https://doi.org/10.1109/ICCV.2017.330.
26 X. Chen, J. Yu, and Z. Wu: IEEE Trans. Cybernetics **50** (2020) 2674. https://doi.org/10.1109/TCYB.2019.2894261.
27 X. Zhu, Y. Xiong, J. Dai, L. Yuan, and Y. Wei: Proc. 2017 IEEE Conf. Computer Vision and Pattern Recognition (IEEE, 2017) 4141–4150. https://doi.org/10.1109/CVPR.2017.441
28 R. Joseph and A. Farhadi: ArXiv. abs/1804.02767 (2018). https://api.semanticscholar.org/CorpusID:4714433
29 X. Zhu, Y. Wang, J. Dai, L. Yuan, and Y. Wei: Proc. 2017 IEEE Int. Conf. Computer Vision (IEEE, 2017) 408–417. https://doi.org/10.1109/ICCV.2017.52
30 C. Feichtenhofer, A. Pinz, and A. Zisserman: Proc. 2017 IEEE Int. Conf. Computer Vision (IEEE, 2017) 3057–3065. https://doi.org/10.1109/ICCV.2017.330
31 C. Eom, G. Lee, J. Lee, and B. Ham: Proc. 2021 IEEE/CVF Int. Conf. Computer Vision (IEEE, 2021) 12016–12025. https://doi.org/10.1109/ICCV48922.2021.01182
32 S. Ren, K. He, R. Girshick, and J. Sun: Proc. IEEE Trans. Pattern Analysis and Machine Intelligence (IEEE, 2017) 1137–1149. https://doi.org/10.1109/TPAMI.2016.2577031
33 J. Dai, Y. Li, K. He, and J. Sun: Proc. The 30th Int. Conf. Neural Information Processing Systems (NIPS, 2016) 379–387. https://doi.org/10.5555/3157096.3157139
34 F. Cheng-Yang, W. Liu, A. Ranga, A. Tyagi, and A. C. Berg: ArXiv. abs/1701.06659 (2017). https://api.semanticscholar.org/CorpusID:7691159
35 Z. Li, F. Zhou: ArXiv. abs/1712.00960 (2018). https://doi.org/10.48550/arXiv.1712.00960
36 L. Wei, A. Dragomir, E. Dumitru, S. Christian, R. Scott, C. Y. Fu, and A. C. Berg: Computer Vision - ECCV 2016, L. Bastian, M. Jiri, S. Nicu, and W. Max, Eds. (Springer International Publishing, Cham, 2016) 1st ed., pp. 21–37. https://doi.org/110.1007/978-3-319-46448-0_2
37 X. Liang, J. Zhang, L. Zhuo, Y. Li and Q. Tian: IEEE Trans. Circuits Syst. Video Technol. **30** (2020) 1758. https://doi.org/10.1109/TCSVT.2019.2905881.

About the Authors

Yanjun Feng received her B.E. degree from Liaoning University of Technology in 1997 and her M.E. degree from Shenyang University of Technology in 2000. She is currently an associate professor with Shenyang Ligong University. Her main research interests include IoT technology and intelligent information processing. (braverfyj@126.com)

Jun Liu received his B.E. and M.E. degrees from Shenyang University of Technology in 1995 and 2000, respectively, and his Ph.D. degree from the Graduate University of Chinese Academy of Sciences and the Shenyang Institute of Automation, Chinese Academy of Sciences, in 2010. He is currently a professor with Shenyang Ligong University. His main research interests include intelligent sensors and detection technology, image and signal processing, and intelligent robots. (lj_mail_sut@163.com)