AJENet: Adaptive Joints Enhancement Network for Abnormal Behavior Detection in Office Scenario

Chengxu Liu, Yaru Zhang, Yao Xue, and Xueming Qian

***Abstract*— With the increasing popularity of intelligent surveillance systems, abnormal behavior detection of human beings based on computer vision is attracting more attention. It aims to classify and locate the abnormal behaviors and coordinates of human beings, respectively, and is a fundamental technology for intelligent security. Existing approaches mainly focus on exploring abnormal behavior features through object detectors. However, in office scenarios, almost all abnormal behaviors are closely associated with the fine-grained feature around the nose, wrist, elbow, and other human joint points regions. Detectors for generic objects cannot adequately capture such differences between abnormal behaviors, resulting in sub-optimal performance. In this paper, we focus on human joints and take one step further to enable effective behavior characteristics learning in office scenarios. In particular, we propose a novel Adaptive Joints Enhancement Network (AJENet), which includes two closely-related components, Joints Predict block (JP) and Adaptive Key Joints Enhancement block (AKJE). JP block is used to predict the human joints and facilitates the feature learning around them implicitly. By inputting the features around joints, the AKJE block enhances the feature representations of key joints according to the abnormal behavior characteristics adaptively. Experimental results demonstrate that our method outperforms other state-of-the-art methods on the collected real office scenario Office Behavior Dataset. Besides, to verify the generalization capabilities and potential of AJENet, we construct comparisons on another generic dataset PASCAL VOC 2012 Action.**

***Index Terms*— Abnormal behavior detection, object detection, joint points, feature enhancement.**

I. Introduction

ABNORMAL behavior can be defined as actions that are unexpected and often evaluated negatively because they differ from typical or usual behavior [1]. With the increasing safety awareness of the public, the demand for video surveillance and abnormal condition detection has grown. Abnormal behavior detection, as a subtask of object detection, is widely used in fields such as intelligent security, human-computer interaction, and intelligent surveillance systems [2].

Abnormal behavior is defined as behavior that is inconsistent with usual or expected behavior, which may be rare, dangerous, or otherwise should not occur. Traditional abnormal behavior surveillance is used to detect abnormalities through the observation of the staff. Such an approach, which only relies on the subjective judgment of the staff, is not only not precise enough, but also time-consuming and inefficient. Therefore, the study of algorithms for abnormal behavior detection has significant commercial value in the field of intelligent surveillance [2]. Besides, compared with abnormal behavior detection of ordinary open scenes, the behaviors in office scenarios are less visible and the characteristics between different behaviors are less differentiated. Existing research lacks adequate attention to abnormal behaviors in increasingly common office scenarios and does not yield good performance. For example, targeting the large flow of people in real grid business offices, Qiao et al. [3] improve YOLOv3 [4] and proposes a system for detecting the number of people in grid business offices and judging whether there is an abnormal situation currently. Different from them, **in this paper, we propose an abnormal behavior detection network in real office scenarios, which can automatically recognize abnormal behavior of employees during work**, which effectively improves work efficiency and reduces safety risks. In our office scenario, we identify eating, lying on desk, and fighting as abnormal behaviors.

Benefiting from significant progress of deep learning, recent years have witnessed an increasing number of advanced algorithms [5], [6], [7], [8], [9] to improve the performance of abnormal behavior detection. Most efforts are based on generic object detection algorithms [4], [10], [11], [12], [13], [14] to obtain the category and location of abnormal behavior. They mainly improve the detection performance by enhancing the multi-scale feature and optimizing the positive/negative sample selection mechanisms during training. However, the office scenario of humans with abnormal behaviors is completely different from the general object. As shown in Fig. 1, almost all the important information about abnormal behaviors is related to the fine-grained feature around joint points regions of the human being. Specifically, here we are referring to joint points as specific points that are significant in human body images. These points are key parts of the human body,

Manuscript received 15 September 2022; revised 21 February 2023 and 26 May 2023; accepted 2 July 2023. Date of publication 14 July 2023; date of current version 7 March 2024. This work was supported in part by the NSFC under Grant 62272380 and Grant 62103317; in part by the Science and Technology Program of Xi'an, China, under Grant 21RGZN0017; and in part by the Shaanxi Key Research and Development under Grant 2022QFY01-17 and Grant 2022FP-40. This article was recommended by Associate Editor D. Zeng. *(Corresponding author: Xueming Qian.)*

Chengxu Liu and Yao Xue are with the School of Information and Communication Engineering, Xi'an Jiaotong University, Xi'an 710049, China (e-mail: liuchx97@gmail.com; xueyao@xjtu.edu.cn).

Yaru Zhang is with the School of Software Engineering, Xi'an Jiaotong University, Xi'an 710049, China (e-mail: zyracademic@163.com).

Xueming Qian is with the Ministry of Education Key Laboratory for Intelligent Networks and Network Security and the SMILES Laboratory, School of Information and Communication Engineering, Xi'an Jiaotong University, Xi'an 710049, China, and also with Shaanxi Yulan Jiuzhou Intelligent Optoelectronic Technology Company Ltd., Xi'an 710000, China (e-mail: qianxm@mail.xjtu.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCSVT.2023.3295432.

Digital Object Identifier 10.1109/TCSVT.2023.3295432

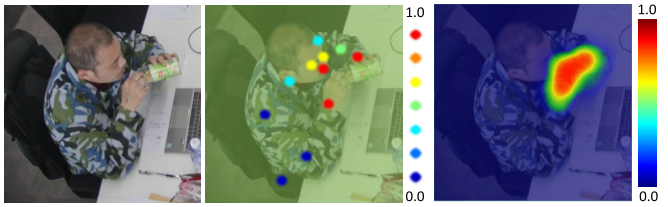

Fig. 1. An example of abnormal behavior 'eating' in office scenarios The original input image is shown on the left. The response strength of the joints is shown in the middle, which is indicated by points of different colors. The corresponding heatmap is shown on the right. The abnormal behavior of 'eating' focus on the joints of the mouth and hands, leading to a higher response in the corresponding heatmap. It demonstrates the fundamental role of joints in abnormal behavior detection.

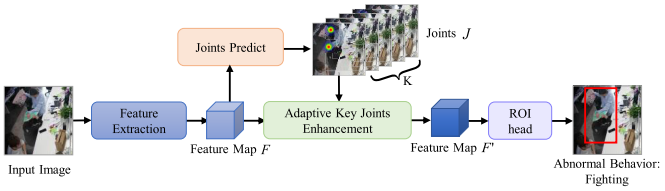

Fig. 2. Overview of the AJENet. Feature extraction is a network to extract features F from the input image. The proposed joints predict (JP) block and adaptive key joints enhancement (AKJE) block are used to predict the human joints J and output the enhance the features F' around joints. The RoI head is used to output the abnormal behavior result.

such as left/right hands, left/right elbows, left/right shoulders, etc. Especially for some challenging scenes (*e.g.,* 'eating'), the characteristic representations are mainly concentrated near the left hand, right hand, and mouth. This reflects that the amount of visual representations of the abnormal behavior is mainly concentrated around these three joints. Therefore, to learn the abnormal behavior representation better, the characteristics representations should be learned in the area where the three joints are located. Different from 'eating', the abnormal behavior of 'lying on desk' is mainly judged by the spatial relationship between the head and the table. With the head close to the table, the joints of the head and back can be clearly recognized, which is beneficial for abnormal behavior detection. Even more, some normal behaviors are very ambiguous (*e.g.,* 'normal'). The network mainly focuses its attention on the hands and detects 'normal' work behavior by determining whether the hands are touching the keyboard, mouse, or laptop. In general, these existing state-of-the-art detectors for general objects still lack optimization for important joints and cannot capture fine-grained feature differences between abnormal behavior. Learning the fine-grained feature representations of various behaviors in office scenarios through the spatial locations and coordinate relationships of key joints is essential.

While some works are based on images-based behavior recognition algorithms [15], [16], [17], [18] to complete the abnormal behavior recognition. They attempt to train a powerful feature extraction network to extract behavior representations. However, those methods focus more on scenarios with a single behavior as the primary objective in an open field of view (e.g. jumping) and ignore spatial fine-grained features that are critical in the real office with subtle motion and multiple objectives.

In this paper, we target to intelligent monitoring in office scenarios, and propose a novel joints-guided abnormal behavior detection method, called AJENet, as shown in Fig. 2. Firstly, AJENet uses a network to extract features from the input image. Secondly, it predicts the human joints by the proposed joints predict (JP) block and enhances the features based on human joints by the proposed adaptive key joints enhancement (AKJE) block. The enhanced features are ultimately used to detect abnormal behavior in office scenarios. In general, AJENet focuses on exploring the representation of abnormal behaviors by using key joints of the human being to assist the learning of spatial information. In particular, to enhance the feature representations of key joints, we design two components JP block and AKJE block to learn the abnormal behavior features separately with two strategies, explicit and implicit.

For the strategy of explicit learning, we propose a novel **A**daptive **K**ey **J**oints **E**nhancement block (AKJE), which is used to enhance the feature around the key joints by the obtained joints position. This design can adaptively output the importance of different joint features in each kind of abnormal behavior firstly. Then it enhances features through an adaptive attention mechanism. The two main advantages of explicit learning are as follows. 1) It can enhance the feature of different joint points by an adaptive mechanism and achieve a better feature representation of each kind of abnormal behavior. 2) The adaptive importance of joint points makes the enhanced feature robust, despite suffering from the inaccurate joint position. However, the inaccurate joint position leads to a deviation in the response of joint point features. It is equivalent to weakening the feature representation of behaviors, resulting in lower detection ability.

To solve this issue, we further propose the strategy of implicit learning, which is used as a complement to improve the accuracy of joint points. We propose a **J**oints **P**redict block (JP) to potentially improve the accuracy of joint points by sharing the feature extraction part. Specifically, we add a branch for predicting joints in front of the AKJE block and optimize them using the well-designed JointsLoss. This design enables the high-level semantics learning of joint features and improves the accuracy of joint points and feature representations at the same time. In our proposed AJENet, the two closely-related components promote each other and achieve more significant improvement than other state-of-the-art (SOTA) methods in the office scenario.

Our contributions are summarized as follows:

- We propose a novel adaptive joints enhancement network (AJENet) for abnormal behavior detection. It can enhance the behavior-related spatial features and enable better abnormal behavior representation learning. The extensive experiments demonstrate that the proposed AJENet can significantly outperform existing SOTA methods on the collected real office scenario dataset.
- We propose an adaptive key joints enhancement block, which can explicitly enhance the feature representations of joint points according to the characteristics of each kind of behavior adaptively.
- We propose a joints predict block, which introduces the joint points prediction abnormal behavior detection and

optimizes the feature extraction jointly by well-designed JointsLoss.

The rest of the paper is organized as follows. Related work is reviewed in Sec. II. The proposed method is elaborated in Sec. III. The collection and construction of the dataset are described in Sec. IV. Experiments and analysis of the related parameters are presented in Sec. V. Discussions are described in Sec. VI. Finally, we conclude this work in Sec. VII.

II. Related Works

In this section, we discuss four main areas related to abnormal behavior recognition in the literature, *i.e.,* abnormal behavior detection, object detection, image-based behavior recognition, and feature enhancement.

A. Abnormal Behavior Detection

The core technology of abnormal behavior detection is to recognize the category and location of abnormal behaviors in the image captured by the camera monitoring. In the field of abnormal behavior detection of monitoring, there are an increasing number of studies emerging. Typically, Ko et al. [5] propose a unified framework based on a DNN to detect abnormal behavior from a standard RGB image and improve detection speed while maintaining accuracy. Tay et al. [6] propose a CNN-based abnormal behavior detector to automatically learn the most discriminative characteristics. Fang et al. [19] and Ji et al. [7] propose a real-time abnormal behavior detection method using improved YOLOv3 [4]. These methods attempt to handle abnormal behavior detection through improved deep learning-based generic object detection methods and temporal information in surveillance.

Besides, depending on the specific scenario, many methods try to pre-define different kinds of abnormal behavior in advance. Mehmood et al. [20] define human falls, some kinds of suspicious behavior, and violent acts as abnormal activities and provides a lightweight framework (LightAnomalyNet) to effectively represent and differentiate between normal and abnormal events. Aiming at the crowd abnormal behavior detection [21], Lentzas et al. [8] connect abnormal behavior detection into ambient assisted living systems for elderly people and provide a review of recent studies focusing on abnormal behavior detection specifically for seniors. Hao et al. [9] propose an end-to-end abnormal behavior detection framework for abnormal or violent behavior by people with mental disorders. Alairaji et al. [22] propose a system to help monitor the activities of students and recognize abnormal/suspicious behavior instantly. Modified from YOLOv3 [4], Qiao et al. [3] propose an abnormal behavior detection system for a large flow of people abnormal detection of grid business offices.

In this paper, based on the behaviors of real office scenarios, we define eating, lying on desk, and fighting as abnormal behaviors. We focus on human joints that promote the detection of abnormal behavior in office scenarios and propose an adaptive joints enhancement network to better abnormal behavior detection in office scenarios.

B. Object Detection

With the development of deep learning recently, a series of object detectors emerge in an endless stream and are widely used in the industrial field, especially in intelligent surveillance systems. There are an increasing number of studies on abnormal behavior detection through detection algorithms [23].

Among these algorithms, one-stage detectors have emerged as a popular paradigm, such as SSD [12], YOLOv3 [4], RetinaNet [13], *etc*. These methods predict the classification and localization of the bounding box directly based on the extracted features. Typically, SSD [12] and YOLOv3 [4] introduce anchor mechanisms on multi-scale features and directly predict the category and confidence scores of bounding boxes. RetinaNet [13] solves the imbalance of positive and negative samples by a specific loss function. Many top-performing frameworks still adopt the proven two-stage pipeline, such as Faster R-CNN [10], Cascade R-CNN [11], *etc*. Faster R-CNN [10] proposes to generate RoI by using fully convolutional networks as a region proposal network (RPN) which greatly improves the detection speed. Cascade R-CNN [11] further proposes a multi-scale detection framework based on it, which greatly improves the performance of small objects. ATSS [14] finds the essential differences between the anchor-free and anchor-based algorithms and creatively proposes to automatically select positive and negative samples according to the statistical characteristics of the object. Recently, TOOD [24] designs a novel Task-aligned Head (T-Head) that offers a better balance between learning task-interactive and task-specific features. In addition to the general object domain, detection algorithms are also very important in the industrial field. Including medical cancer cell detection [25], face detection [26], product detection [27], and so on.

However, these current state-of-the-art detectors are ineffective for abnormal behavior detection in office scenarios. It is because these methods cannot capture the fine-grained feature differences around key joint differences, which are critical for the classification and localization of abnormal behavior in office scenarios. So these algorithms are necessary to be improved to get better performance, and in this paper, we propose an adaptive joints enhancement network to enable the behavior feature learning of joint points and get better abnormal behavior detection results in office scenarios.

C. Image-Based Behavior Recognition

Image-based behavior recognition aims to recognize the behavior in the still image. Compared to the more common video-based behavior recognition [28], image-based behavior recognition is a more challenging task, due to large appearance variations and lack of motion descriptions. Some traditional image-based behavior recognition methods [29] capture the features of different behaviors through hand-crafted feature descriptors. Most of the existing work focuses on scene-object contexts [30], [31] or human parts-poses-attributes [32], [33], [34], [35]. Among them, the methods based on scene-object contexts consider the image or target as an entirety and utilizes spatial information to

recognize behaviors. Oquab et al. [30] utilize an eight layers CNN network to obtain the texture representation of the target. Simonyan et al. [31] obtain scene features by combining 16-layer and 19-layer CNNs, and then connecting SVMs to acquire behavior recognition results. The other approach, focusing on human parts-poses-attributes, is more concerned with behavior-related local region information. Hoai [32] divides the scene into various regions at different scales and then inputs the features and positions of each region into the eight layers network to obtain the behavior recognition results. Action Part [33] designs individual classifiers for the head, torso, legs, and whole person on the basis of Poselets. R*CNN [34] utilizes the framework of RCNN to make better use of spatial texture information by extracting features from multiple behavior-related regions. Multi-branch [35] combines scene features and the local region features for behavior recognition.

Different from the above methods, by introducing human joints, our approach enables better classification and localization with behavior-related features and texture information. We adaptively enhance the relevant region features for various behaviors with better robustness.

D. Feature Enhancement

Feature enhancement is an efficient method to enhance feature representation [36], [37], it is generally implemented by an attention mechanism. It is an effective module that can focus on the regions of interest with limited weights and calculations and is widely used in various fields [26], [38], [39], [40], [41]. In essence, the attention mechanism is to filter out the important information from a large amount of redundant information and focus on the important ones, ignoring most of the unimportant ones. Non-local [42] is an effective attention mechanism that performs feature refinement to expand the receptive field from a local area to the whole image. Self-attention mechanism [43] reduces the dependence on external information and is better at capturing the internal correlation of data or features. To better recalibrate the channel-level characteristic response, SENet [36] proposes to use the global average pool to extract the global descriptors, which explicitly model the interdependence between the channels.

In the feature enhancement associated with the human joints, AdaSGN [44] reduces the computational cost of the inference process by adaptively controlling the input number of the joints, achieving better video-based action recognition performance. CD-JBF-GCN [45] proposes a correlation-driven joint-bone fusion graph convolutional network as an encoder to learn more discriminative feature representations. PoseConv3D [46] relies on a 3D heatmap volume as the base representation of human skeletons and is more effective in learning spatiotemporal features. Different from these methods that use the correlation within the features themselves, we add joint points information and reconstruct the feature in the spatial dimension. Our method enhances the characteristic response of joint point areas by adaptive modeling the feature interdependence between the joints.

III. Method

A. Overview

The overview of the proposed adaptive joints enhancement network (AJENet) is shown in Fig. 2. The whole network can be divided into four steps. 1) Feature extraction composed by FPN [13] is used to extract the feature of the input image. 2) Obtaining the joints information from the feature by using the joints predict (JP) block. 3) Inputting the joints information into adaptive key joints enhancement (AKJE) block to adaptive enhance the feature around the joints. 4) Inputting the enhanced feature into the RoI head to output the regression and classification results. The highlight of our approach is the proposal of two closely-related components, JP and AKJE, to enable the joints to feature representation learning.

B. Feature Extraction

We follow existing works [10], [11], using the ResNet50 to extract multi-scale features which combine with FPN [13], as the feature extraction part in our AJENet. Finally, this part outputs features with five scales.

Different from them, in our network, the feature extraction part can be constrained indirectly by the joint information at the same time during training. This indirect supervision of joints enables the network to focus more on the features learning around the joints to some extent.

C. Joints Predict Block

To better capture fine-grained feature differences between abnormal behavior, we propose a joints predict (JP) block following the feature extraction to predict the human joints, as shown in Fig. 2. It is worth noting that our proposed JP block and behavior detection share the same feature extraction network, since the supervision of the joints during training can simultaneously facilitate the feature learning around the joints. Besides, to stabilize the optimization of joints, we propose the JointsLoss in training to constrain the learning of the network. In this section, We describe the network structure and the proposed JointsLoss during training.

1) Structure: As shown in Fig. 3, JP block can be composed of stacked residual blocks, denoted as $RBs(\cdot)$. In detail, the $RBs(\cdot)$ includes four residual blocks with 3×3 kernel size and one residual block with 1×1 kernel size. Among them, the 3×3 residual blocks are applied to extract the joint features from the input feature $\mathbf{F}$. The 1×1 residual block pays attention to local joints and is used to output the joints result $\mathbf{J} \in \mathbb{R}^{H \times W \times K}$. The joints result $\mathbf{J}$ indicates the position distribution of joints, which includes K categories belonging to K output channels. It is worth noting that there are 17 types of joint points used in our method, referring to specific points of importance in the human body, such as eyes, shoulders, elbows, wrists, etc. The experimental analysis of the selection of joint point types and the number of choices can be found in Sec. VI-B.4. In detail, the process can be expressed by:

$$\mathbf{J} = RBs(\mathbf{F}), \tag{1}$$

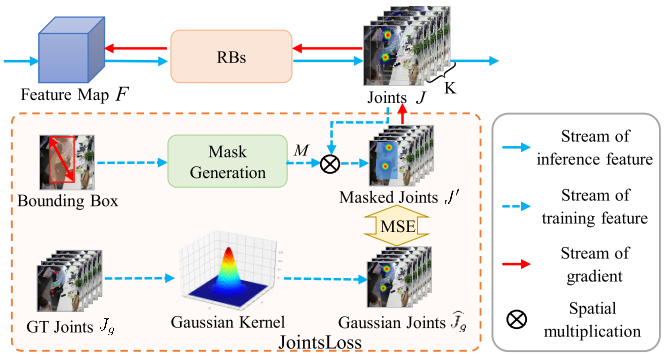

Fig. 3. The structure of the Joints Predict (JP) block. During inference, the JP block is used to predict human joints J. During training, the JointsLoss, which generates the Gaussian Joints $\hat{J}_g$ and Masked Joints J', is used to supervise the training of the entire network.

where $RBs(\cdot)$ denotes the stacked residual blocks. $\mathbf{F}$ is the input feature, which outputs from the features extraction network. $\mathbf{J}$ is the joints result. In $\mathbf{J}$, if one position is in proximity to a joint, then its output will have a larger value.

2) JointsLoss: The joints result $\mathbf{J}$ has the same size as the input image and the largest pixel value at each joint coordinate. If we directly use sparse ground truth of joints to supervise JP block during training, it will result in fewer positive samples. It may cause unstable training and unexpected output in the background area irrelevant to the behavior. To further suppress irrelevant samples and boost the stability of the training of the joints predict block, we specially design the JointsLoss. The JointsLoss is capable of removing false outputs from the background and expanding the original sparse joints to increase the number of positive samples.

As shown in Fig. 3, the joints only appear in the human body and are also in the detected bounding box (*i.e.,* the foreground region). Therefore, to mitigate the effect of irrelevant or incorrect joint samples (*i.e.,* negative samples) in the background, we add the mask generation mechanism to suppress the joints that output in irrelevant regions. Specifically, we first generate the foreground mask $\mathbf{M}$ by utilizing the bounding box of abnormal behavior, which come from the ground truth. This process can be expressed by:

$$\mathbf{M}(x, y) = \begin{cases} 1, & if\ (x, y) \in bbox; \\ 0, & else, \end{cases} \tag{2}$$

where $bbox$ denotes the ground truth of bounding boxes. We multiply the joints result $\mathbf{J}$ and foreground mask $\mathbf{M}$ to get masked joints $\mathbf{J}'$. It can be expressed as:

$$\mathbf{J}' = \mathbf{J} \otimes \mathbf{M}, \tag{3}$$

where $\otimes$ denotes the multiplication of corresponding locations. Compared with not performing the operations on the joints result $\mathbf{J}$, this design can obtain masked joints $\mathbf{J}'$ that effectively reduce the impact of negative or incorrect samples in extraneous regions outside the behavior bounding box during training, thus yielding more accurate and robust joint locations.

Then, to increase the number of positive samples during training, we filter the ground truth of joints $\mathbf{J_g}$ (*i.e.,* the GT joints in Fig. 3) with a Gaussian kernel to expand the pixel of joints, as shown in Fig. 3. This operation can boost the number of positive samples during training, and effectively stabilize the training of the JP block. In detail, we use the Gaussian kernel to filter the ground truth of joints $\mathbf{J_g}$. The Gaussian kernel can be represented as:

$$G(x, y) = A \cdot exp(-(\frac{(x - x_0)^2}{2\sigma^2} + \frac{(y - y_0)^2}{2\sigma^2})), \tag{4}$$

in which x_0, y_0 denote the coordinates of the central joints in the x-axis and y-axis, respectively, σ is the standard deviation, and A is the intensity of the Gaussian filter. The filtered joints (*i.e.,* the Gaussian joints in Fig. 3) can be expressed as:

$$\widehat{\mathbf{J}}_\mathbf{g} = \Phi_G(\mathbf{J_g}), \tag{5}$$

where $\Phi_G(\cdot)$ is a filter operation with kernel G. $\mathbf{J_g}$ is the ground truth of joints. $\widehat{\mathbf{J}}_\mathbf{g} \in \mathbb{R}^{H \times W \times K}$ denotes the outputted Gaussian joints and has the same size as the $\mathbf{J_g} \in \mathbb{R}^{H \times W \times K}$, where each category of joints belongs to a channel and takes the value of 1 at the joints location and 0 at other locations. Such a design of increasing positive samples through a Gaussian distribution can significantly alleviate the imbalance of positive and negative samples during training. In our method, we set σ as 4 times the standard deviation and A as 1.

Finally, after the above operations to obtain the masked joints $\mathbf{J}'$ and Gaussian joints $\widehat{\mathbf{J}}_\mathbf{g}$, we further use the MSE loss function to constrain the JP block. It can be expressed as:

$$\begin{aligned} &l_{joint}(\widehat{\mathbf{J}}_\mathbf{g}, \mathbf{J}) \\ &= l_{mse}(\widehat{\mathbf{J}}_\mathbf{g}, \mathbf{J}') \\ &= \frac{1}{H \cdot W \cdot K} \sum_{k=1}^{K} \sum_{x=1}^{H} \sum_{y=1}^{W} (\widehat{\mathbf{J}}_\mathbf{g}(x, y, k) - \mathbf{J}(x, y, k))^2 \cdot \mathbf{M}(x, y), \end{aligned} \tag{6}$$

where l_{joint} denotes the proposed JointLoss. $\widehat{\mathbf{J}}_\mathbf{g}$ and $\mathbf{J}$ represent the Gaussian joints obtained from ground truth and joints result predicted from JP block, respectively. $\mathbf{M}$ is the foreground mask of abnormal behavior in the bounding box. x, y, and k denote the coordinates of the elements in joints result $\mathbf{J}$. The whole design of JointsLoss is enabled to eliminate background interference during training, while increasing the positive sample of joints for training. It not only can stabilize the training effectively, but also can enhance the robustness of the network.

Besides, this JointsLoss can also be stacked on multiple scales. Specifically, we use the interpolation to transform these Gaussian joints $\widehat{\mathbf{J}}_\mathbf{g}$ to different scales s and obtain multi-scale Gaussian joints $\widehat{\mathbf{J}}^\mathbf{s}_\mathbf{g}$. During training, we use the multi-scale Gaussian joints $\widehat{\mathbf{J}}^\mathbf{s}_\mathbf{g}$ to constrain the predicted joints $\mathbf{J}$. The adoption of a multi-scale structure is mainly in two considerations. 1) The multi-scale joints can further stabilize the training of joints predict block. 2) The multi-scale structure enhances the robustness of the model to joint location shifts. In the following part, we describe this structure only at one scale for brevity.

D. Adaptive Key Joints Enhancement Block

To achieve a better feature representation around joints of each kind of abnormal behavior, we propose the adaptive key

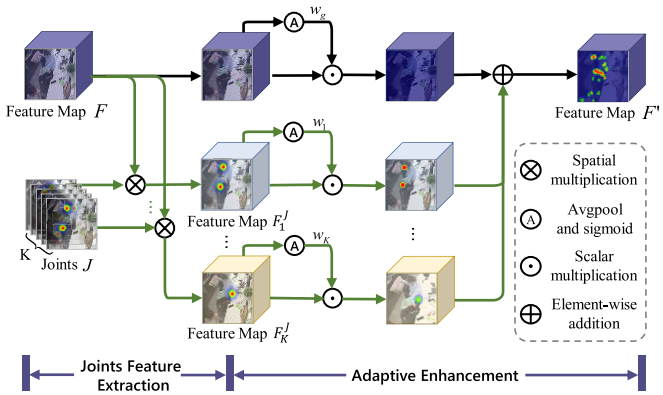

Fig. 4. The structure of the Adaptive Key Joints Enhancement (AKJE) block. During the joint feature extraction process, this block is used to extract features around each joint. During the adaptive enhancement process, this block uses an adaptive attention mechanism to enhance and fuse these features.

joints enhancement (AKJE) block to enhance the features. In particular, as shown in Fig. 4, we first extract features around joints based on the joints result $\mathbf{J}$ obtained from the JP block. Then, the features $\mathbf{F}$ from the feature extraction network are further fused to output the enhanced features $\mathbf{F}'$ by an adaptive attention mechanism.

This design enables the model to learn the confidence scores of joints based on their features, and thus adaptively enhances the important joints among them. In the following, we divide it into joint feature extraction and adaptive enhancement to describe them separately.

1) Joints Feature Extraction: To extract the features around joints, we take the joints result $\mathbf{J} \in \mathbb{R}^{H\times W\times K}$ as input to provide the joints' position. In particular, we use $\mathbf{F}$ to denote the feature that is output from the feature extraction network. For the joints k, the extracted joints feature $\mathbf{F_k^j}$ can be obtained by:

$$\mathbf{F_k^j} = \mathbf{J_k} \otimes \mathbf{F}, \tag{7}$$

where $\mathbf{F_k^j}$ represents the extracted joints feature of the k^{th} joints. $\mathbf{J_k}$ is the k^{th} joints in $\mathbf{J} \in \mathbb{R}^{H\times W\times K}$. $\mathbf{F}$ is the feature output from the feature extraction network. $\otimes$ denotes the multiplication of corresponding locations. We can obtain all the extracted joints feature $\mathbf{F_k^j} \in \mathbb{R}^{H\times W\times C}$ in the same way, where $k \in \{1, 2, \ldots, K\}$ represents the k^{th} joints in all K categories.

Besides, we can extract joint features on multiple scales to obtain more accurate and robust features around joints. Specifically, we first use $\mathbf{F_s}$ to denote the feature $\mathbf{F}$ at scale s that is output from the feature extraction network. Then, we downsample the joints $\mathbf{J}$ to the scale s. Compared to other down-sampling methods (*e.g.,* convolution), this method has a smaller computational effort while localizing accurately. Finally, we obtain the extracted joints feature $\mathbf{F_s^j}$ at scale s in the same way as in Equ. 7. In the following part, we describe the extracted joint features only at one scale for brevity.

2) Adaptive Enhancement: Based on the extracted joints feature $\mathbf{F_k^j}$ of the k^{th} joints, we propose to squeeze it to generate a confidence score of joints, and then enhance the joints features by weighted fusion. The j in $\mathbf{F_k^j}$ represents a symbol that indicates that $\mathbf{F_k^j}$ is the feature of joints.

In detail, as shown in Fig. 4, we use the global average pool to aggregate the local descriptors around joints. The obtained descriptors w_k corresponding to the joints k can be formulated as:

$$w_k = \frac{1}{H \times W \times C} \sum_{x=1}^{H} \sum_{y=1}^{W} \sum_{c=1}^{C} \mathbf{F_k^j}(x, y, c), \tag{8}$$

where $\mathbf{F_k^j}$ represents the extracted joints feature of the k^{th} joints. $w_k \in \mathbb{R}^{1\times 1\times 1}$ aggregates all the information in $\mathbf{F_k^j}$. H, W, and C denote the dimension of the feature. We can obtain the $\{w_1, w_2, \ldots, w_K\}$ in the same way. $k \in \{1, 2, \ldots, K\}$ represents the k^{th} joints in all K categories. We use the widely used $Sigmoid(\cdot)$ function to normalize the obtained feature descriptors $w_k, k \in \{1, 2. \ldots, K\}$. This process can be formulated by:

$$\begin{aligned} w_k{}' &= Sigmoid(w_k) \\ &= \frac{1}{1 + e^{-w_k}}, \end{aligned} \tag{9}$$

where $w_k{}'$ denotes the normalized feature descriptor, which can represent the confidence score of k^{th} joints between 0 and 1. We also can obtain the confidence score $w_g{}'$ of input feature $\mathbf{F}$ in the same way. The use of the $Sigmoid(\cdot)$ function enables the importance of each joint to be calculated separately and adaptively, while alleviating the coupling that occurs when multiple joints are related.

Then, we enhance the joints feature by weighted fusion, this process can be formulated by:

$$\mathbf{F}' = (\mathbf{F} \odot w_g{}') \oplus \sum_{k=1}^{K} (\mathbf{F_k^j} \odot w_k{}'), \tag{10}$$

where $\mathbf{F}'$ represents the enhanced feature, which has the same dimensions as $\mathbf{F}$. $\mathbf{F}$ and $\mathbf{F_k^j}$ indicate the input feature extracted from the input image and the extracted joints feature of the k^{th} joints above-mentioned, respectively. $w_g{}'$ and $w_k{}'$ are the confidence score obtained from the above calculation. $\odot$ and $\oplus$ represent the scalar multiplication and element-wise addition, respectively. The output enhanced feature $\mathbf{F}'$ has the same dimensions as the input feature $\mathbf{F}$.

To demonstrate the function of the block intuitively, we visualize the joints and the corresponding heat map in Fig. 5. It can be seen that, for different abnormal behaviors, AKJE can focus on the joints that are more important for behavior detection and then effectively enhance the features in key joints. Finally, as shown in Fig. 2, the enhanced feature $\mathbf{F}'$ can be fed into the classifier and regressor for abnormal behavior detection.

E. RoI Head

Based on the enhanced features following the AKJE block, we add two parallel modules for abnormal behavior classification and regression, respectively.

For the behavior classification branch, we first extract the classification feature from enhanced feature $\mathbf{F}'$ using four convolutional layers with 3×3 kernel size. Then we use a convolutional layer with 1×1 kernel size to transform the

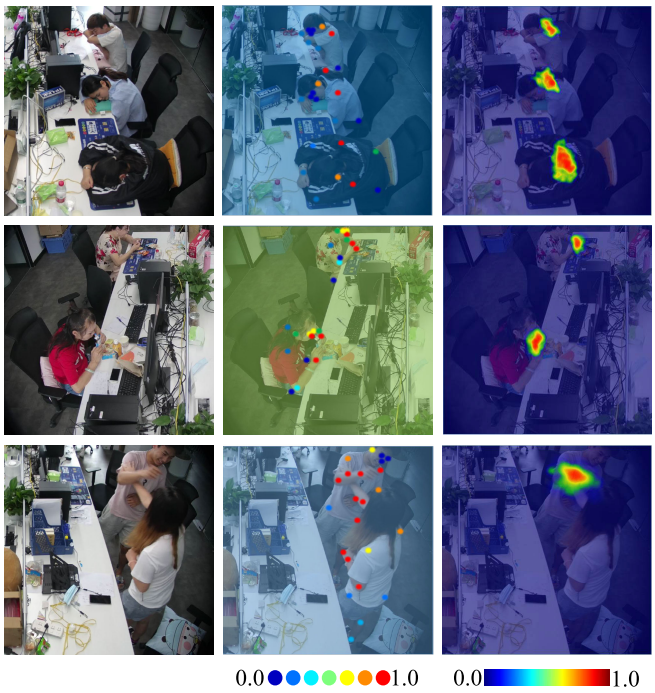

Fig. 5. Visualization of key joints attention on the Office Behavior Dataset. The first column is the original input image. The coordinate points in the second column represent the joints with different weights output by the AKJE block. The third column is heatmaps of the feature visualization, which denote the spatial attention regions that the network pays more attention to when detecting anomalous behavior. Where warm colors represent a large weight.

channel of the feature into a number of categories. Finally, the branch outputs the classification results of the abnormal behavior.

Similarly, for the regression branch, the regression feature is first extracted by four convolutional layers with 3×3 kernel size. Then we add a convolutional layer with 1×1 kernel size and transform the channel of the feature to four. Finally, we output the coordinates of the bounding box $(x_t l, y_t l, x_b r, y_b r)$ where the abnormal behavior is located, where $(x_t l, y_t l)$ and $(x_b r, y_b r)$ represent the coordinates of the top-left corner and the bottom-right corners of the bounding box, respectively.

Finally, we use the Non-Maximum Suppression (NMS) algorithm to remove the redundant detection results and retain the bounding boxes with the highest quality. Finally, we summarize and output their locations and classification results.

F. Training

As described above, during training, our approach includes a total of three loss components, the classification and regression loss for abnormal behaviors detection, and the JointsLoss for enhancing the representation of learning around joints.

In detail, we follow RetinaNet [13], using FocalLoss as our classification branch, as follows:

$$l_{cls}(p_t) = \sum_{t}^{B} -\alpha_t (1 - p_t)^{\gamma} log(p_t), \tag{11}$$

where the definition of p_t is described as:

$$p_t = \begin{cases} p, & if \ \ y = 1, \\ 1 - p, & otherwise, \end{cases} \tag{12}$$

where B is the total number of samples, α, and γ are the modulating factor, $y \in \{\pm 1\}$ specifies the ground-truth class. We set α and γ as 0.25 and 2.0, respectively. p denotes the score of the output of the classification branch. p_t represents the notation of a sample's score for convenience.

The bounding box regression adopted smooth L_1 loss function can be represented by l_{reg} as follows:

$$l_{reg} = \sum_{i \in pos}^{N} \sum_{m \in \{cx,cy,w,h\}} smooth_{L_1}(b_i^m - g_i^m), \tag{13}$$

where N is the number of matched positive boxes, b and g are the predicted box and the ground truth box respectively as the same as described in [10]. The box center (cx, cy), width w, and height h are the offsets used for regression.

It is essential for the network to balance these three tasks. Therefore, we multiply l_{joint}, as described in Equ. 6, with a weight λ to enable the validity of the joints' prediction branch while not harming the performance of classification and regression. The total loss function is formulated as follows:

$$l_{total} = l_{cls} + l_{reg} + \lambda \cdot l_{joints}, \tag{14}$$

where l_{cls} is the classification loss and l_{reg} is the location regression loss.

IV. Datasets

As described in Sec. II mentioned above, the study of abnormal behavior detection in office scenarios still remains to be further explored. Therefore, in this paper, we collect an abnormal behaviors detection dataset in the office scenario, called Office Behavior Dataset. Besides, to supervise the feature learning around human joints, we propose a joints generation strategy to generate the joint labels in datasets.

A. Office Behavior Dataset

Based on the surveillance of real office scenarios, we collect the dataset using high-definition cameras. The collected Office Behavior Dataset contains 43,530 images with a resolution of 1920×1080, and each image contains at least three office people. There are three categories of abnormal behavior in this dataset, labeled as eating, lying on desk, and fighting, and one normal behavior, labeled as normal.

Among them, eating and lying on desk are the abnormal behaviors of single person, including a person labeled with its bounding box and category. Fighting is the abnormal behavior of multiple people, including two-person labeled with their bounding box and category. Other behaviors are recognized as normal. Among them, 80% is used as training data, and the remaining 20% is used as test data.

B. Joints Generation Strategy

The joints generation strategy can be divided into four steps. 1) We chose the SOTA joints prediction models as the joints prediction network to generate the coarse joints. Specifically, for the abnormal behaviors of single person, we use the top-down heatmap method HRNet [47]. For the abnormal behavior between multiple people, we use the associative embedding method HigherHRNet [48]. 2) We chose a high-quality joints dataset (*i.e.,* Microsoft COCO dataset) to fine-tune the selected joints generation network for generating the required joints. This dataset includes 250,000 people with joints. The total number of joints categories is 17.[1] 3) We use the chosen high-quality joints dataset to train the joints prediction network and generate the coarse joints. Specifically, HRNet achieves an AP of 0.716 on COCO val2017. HigherHRNet attains an AP of 0.772 on COCO val2017. This performance demonstrates the accuracy of the generated coarse joints. 4) To guarantee the accuracy of the output joints, we refine them manually based on the joints output obtained above. All these steps ensure the accuracy of the joints.

V. Experiments

We compare our method with other state-of-the-art behavior recognition and object detection methods. We design sufficient ablation experiments to demonstrate the effectiveness of each block in our method and analyze the results in detail.

In this section, we first describe two behavior detection datasets. We then present the detailed experimental design, which includes compared methods, performance evaluation, and implementation details. Finally, we show and analyze the actual experimental results.

A. Dataset

1) Office Behavior Dataset: As described in Sec. IV-A, the Office Behavior Dataset contains 43,530 images, all collected from office scenarios. There are four categories labeled in this dataset, namely eating, lying on desk, fighting, and normal. We only detect the first three abnormal behaviors and use the strategy in Sec. IV-B to generate the joints label of them. Among them, 80% is used as training data, and the remaining 20% is used as test data.

2) PASCAL VOC 2012 Action Dataset: The PASCAL VOC Action Dataset [49] serves as one of the PASCAL VOC 2012 challenges, containing a total of 4,588 images. There are 11 types of behaviors, namely jumping, phoning, playing instrument, reading, riding bike, riding horse, running, taking photo, using computer, walking, and others. Each person is marked with a bounding box for their position and category. We use the strategy in Sec. IV-B to generate the joints label. The ratio of the training set and validation set is 1:1.

B. Experimental Settings

1) Compared Methods: To verify the advantages of our proposed AJENet, we compare our method with other behavior recognition methods and object detection methods. The details and settings of these methods are as follows:

Oquab et al. [30]: This method trains an eight layers CNN network to behavior recognition based on the predicted bounding box.

Hoai [32]: This method inputs the multiple regions with different scales and locations into an eight layers network to obtain their respective recognition results, and then integrates them as the final output.

Action Part [33]: This method increases the number of convolutional layers on the basis of Poselets [50], and simultaneously trains classifiers for the head, torso, legs, and whole person.

Simonyan et al. [31]: This method combines a 16-layer and a 19-layer network and trains SVMs on fc7 features to output the behavior recognition results.

Faster R-CNN [10]: This method first detects the candidate behaviors through the RPN subnetwork, and then refines the candidate behaviors through the ROI head.

*R*CNN [34]:* Based on the RCNN [51], more regions are used for prediction to use the context information better and output the behavior detection results.

SSD [12]: This method takes VGG16 [31] as the backbone, and extracts feature maps at different scales to do behavior detection.

YOLOv3 [4]: The method adopts DarkNet53 [4] as the backbone to detect targets on three different scales.

RetinaNet [13]: This method proposes the FPN to detect targets at different scales and introduces FocalLoss to focus on difficult samples.

Multi-branch [35]: This method adds two attention branches, namely scene-level attention and region-level attention, to output the recognized behavior.

Cascade R-CNN [11]: It continuously raises the IoU threshold of detection results by cascading the ROI head.

FCOS [52]: The method detects the target by predicting these center points and four distances from the center point to the rectangular boundary.

ATSS [14]: The method used an adaptive training sample selection technique based on RetinaNet.

TOOD [24]: The method to solve the inconsistency of classification and regression of detection and its performance surpasses the recent one-stage detectors by a large margin.

*(*Ours): we proposed an abnormal behavior detection method, which includes the Joints Predict (JP) block and Adaptive Key Joints Enhancement (AKJE) block.

2) Performance Evaluation: For fair comparisons, we follow existing works [31], [34], [35] to use VOC07 object detection evaluation indicators to compare performance, which including $Recall$, $Precision$, AP, and mAP. In this experiment, the threshold of IoU is set to 0.5.

3) Implementation Details: Our experiment is conducted on an NVIDIA 1080Ti GPU through PyTorch and mmdetection. We use SGD as the optimizer and momentum is 0.9, and weight decay is 0.0001. The initial learning rate is set to 0.01. At the 8^{th} and 11^{th} epochs, the learning rate decays by a factor of 10. The total number of epochs is 12. At the same time, the warmup mechanism is adopted. In the first 500 iter, the

[1]Including nose, left eye, right eye, left ear, right ear, left shoulder, right shoulder, left elbow, right elbow, left wrist, right wrist, left hip, right hip, left knee, right knee, left ankle, right ankle.

TABLE I
RESULTS OF QUALITATIVE COMPARISON ON OFFICE BEHAVIOR DATASET. 'LYING' INDICATES THE 'LYING ON DESK' FOR BREVITY. RED INDICATES THE BEST AND BLUE INDICATES THE SECOND BEST PERFORMANCE (BEST VIEW IN COLOR)

Method	Backbone	Categories			mAP
		eating	lying	fighting	
Oquab *et al.* [30]	AlexNet	0.422	0.698	0.602	0.574
Hoai [32]	AlexNet	0.501	0.745	0.653	0.633
Action Part [33]	AlexNet	0.582	0.795	0.728	0.702
Simonyan *et al.* [31]	VGG16	0.611	0.800	0.742	0.718
Faster R-CNN [10]	R50-FPN	0.720	0.853	0.789	0.787
R*CNN [34]	VGG16	0.705	0.840	0.787	0.777
SSD [12]	VGG16	0.509	0.720	0.677	0.635
YOLOv3 [4]	DarkNet53	0.614	0.808	0.721	0.714
RetinaNet [13]	R50-FPN	0.683	0.847	0.777	0.769
Multi-branch [35]	VGG16	0.725	0.851	0.793	0.790
Cascade R-CNN [11]	R50-FPN	0.727	0.860	0.801	0.796
FCOS [52]	R50-FPN	0.679	0.845	0.774	0.766
ATSS [14]	R50-FPN	0.723	0.858	0.791	0.791
TOOD [24]	R50-FPN	0.726	0.859	0.793	0.793
AJENet(Ours)	R50-FPN	0.737	0.869	0.816	0.807

learning rate increases from 0.001 to 0.01. Resize the input images to 512 and keep the aspect ratio. Except for adding random horizontal flips, no other data augmentation methods are used.

C. Comparisons With State-of-the-Art Methods

To verify the effectiveness of our method, we compare other state-of-the-art related behavior recognition methods and object detection methods. Including Oquab et al. [30], Hoai [32], Action Part [33], Simonyan et al. [31], Faster R-CNN [10], R*CNN [34], SSD [12], YOLOv3 [53], RetinaNet [13], Multi-branch [35], Cascade R-CNN [11], FCOS [52], ATSS [14], and TOOD [24].

1) Quantitative Comparison: As shown in Tab. I, the results for each algorithm on the Office Behavior Dataset. Our proposed AJENet uses Cascade R-CNN [11] as the base framework. For fair comparisons, we follow existing works [13], [14] to use ResNet50 and FPN [13] as the backbone. AJENet achieves a result of 80.7% mAP and significantly outperforms the other algorithms by a large margin. Compared with typical behavior recognition methods (*e.g.,* Multi-branch [35], R*CNN [34]), our method can effectively focus on multiple objects in the image, which has high superiority in office scenarios. Compared with generic target detection methods (*e.g.,* Cascade R-CNN [11], ATSS [14]), AJENet can learn the fine-grained features of abnormal behaviors in office scenarios and thus has more accurate detection results. Among them, the performance of fighting improved significantly, by 1.5% higher, and eating increased by 1.0%. It is because these methods cannot capture the fine-grained feature differences around key joint differences, which are critical for the classification and localization of abnormal behavior in office scenarios. This large margin demonstrates the power of AJENet in abnormal behavior representation learning.

To further verify the generalization capabilities of AJENet, we also demonstrate the effectiveness of our method on the VOC Action 2012 val set [49], as shown in Tab. II. AJENet can significantly outperform other methods by more than 1.0%. This large margin also demonstrates the generalization capabilities of AJENet.

2) Qualitative Comparison: To further compare the visual qualities of different approaches, we visual results generated by AJENet and other SOTA methods on Office Behavior Dataset and PASCAL VOC 2012 Action Dataset in Fig. 6 and Fig. 7. For fair comparisons, we take the original result or author-released code to get those results with the same training strategies. It can be observed that AJENet has great accuracy in visual results. For example, in the first column of Fig. 6, behaviors 'eating' and 'normal', have very similar visual features. AJENet guides the network to learn mouth region features by introducing joints, leading to better detection performance. It indicates the necessity of introducing joint information to learn the feature representation.

3) Model Complexity: To further demonstrate the superiority of our approach, we analyzed the Params and FLOPs of each component. In particular, as described in Sec. III-C.1, our proposed JP block consists of four residual blocks with 3×3 kernel size and one residual block with 1×1 kernel size. The total Params and FLOPs of the JP block are 0.591M and 19.13G, respectively. As described in Sec. III-D, the AKJE block enhances the features by a two-part operation of joints feature extraction and adaptive enhancement, and it does not contain any Params. The Params and FLOPs of the AKJE block are 0.0M and 0.58G, respectively.

It is worth noting that our approach does not require additional joint points as input during the inference, and the two blocks achieve a significant performance improvement by adding smaller Params and FLOPs compared to the whole method. It demonstrates the superiority of each component of AJENet, which can get better performance for behavior detection in office scenarios.

D. Ablation Experiments

To verify the effectiveness of each component in our method, we conduct ablation experiments on the Office Behavior Dataset and PASCAL VOC 2012 Action Dataset. The experimental results are shown in Tab. III. The 'Base' denotes the method of RetinaNet [13], which uses the backbone as ResNet50. On the Office Behavior Dataset, the results show that the mAP value has increased by 1.8% by joining the JP. It demonstrates that the addition of a supervised JP block enables the backbone network to learn features that are more useful for behavior detection and improve the performance of behavior classification tasks. When the AKJE block is added, the mAP is increased by 1.9%. It demonstrates that the block utilizes the location information of human joints and enhances the feature representation of joint regions. By adaptively adjusting the weights of different joints and assigning greater weights to important key joints, the network can better learn the essential representation of the behavior. Adding JP and AKJE at the same time, mAP increased by 2.7%. To show the detection performance of our method, we also visualize the detection results of AJENet in Fig. 8. It demonstrates the superiority of each component of AJENet, which can get better performance for behavior detection in office scenarios.

TABLE II

RESULTS OF QUALITATIVE COMPARISON ON VOC ACTION 2012 VAL SET [49]. 'INSTRUMENT' INDICATES THE 'PLAYING INSTRUMENT', 'PHOTO' INDICATES THE 'TAKING PHOTO', AND 'COMPUTER' INDICATES THE 'USING COMPUTER' FOR BREVITY. RED INDICATES THE BEST AND BLUE INDICATES THE SECOND BEST PERFORMANCE (BEST VIEW IN COLOR)

Method	Backbone	Categories										mAP
		jumping	phoning	instrument	reading	riding bike	riding horse	running	photo	computer	walking	
R*CNN [34]	VGG16	0.889	0.799	0.951	0.822	0.961	0.978	0.879	0.853	0.940	0.715	0.879
YOLOv3 [4]	DarkNet53	0.879	0.775	0.942	0.810	0.951	0.970	0.875	0.846	0.925	0.727	0.870
Multi-branch [35]	VGG16	0.878	0.784	0.937	0.811	0.950	0.971	0.860	0.855	0.931	0.734	0.871
Cascade R-CNN [11]	R50-FPN	0.902	0.800	0.965	0.823	0.961	0.984	0.885	0.875	0.950	0.745	0.888
ATSS [14]	R50-FPN	0.900	0.791	0.960	0.830	0.962	0.984	0.877	0.870	0.945	0.741	0.885
TOOD [24]	R50-FPN	0.898	0.795	0.961	0.830	0.964	0.985	0.888	0.876	0.951	0.742	0.888
AJENet(Ours)	R50-FPN	0.912	0.803	0.969	0.837	0.969	0.988	0.891	0.897	0.952	0.758	0.898

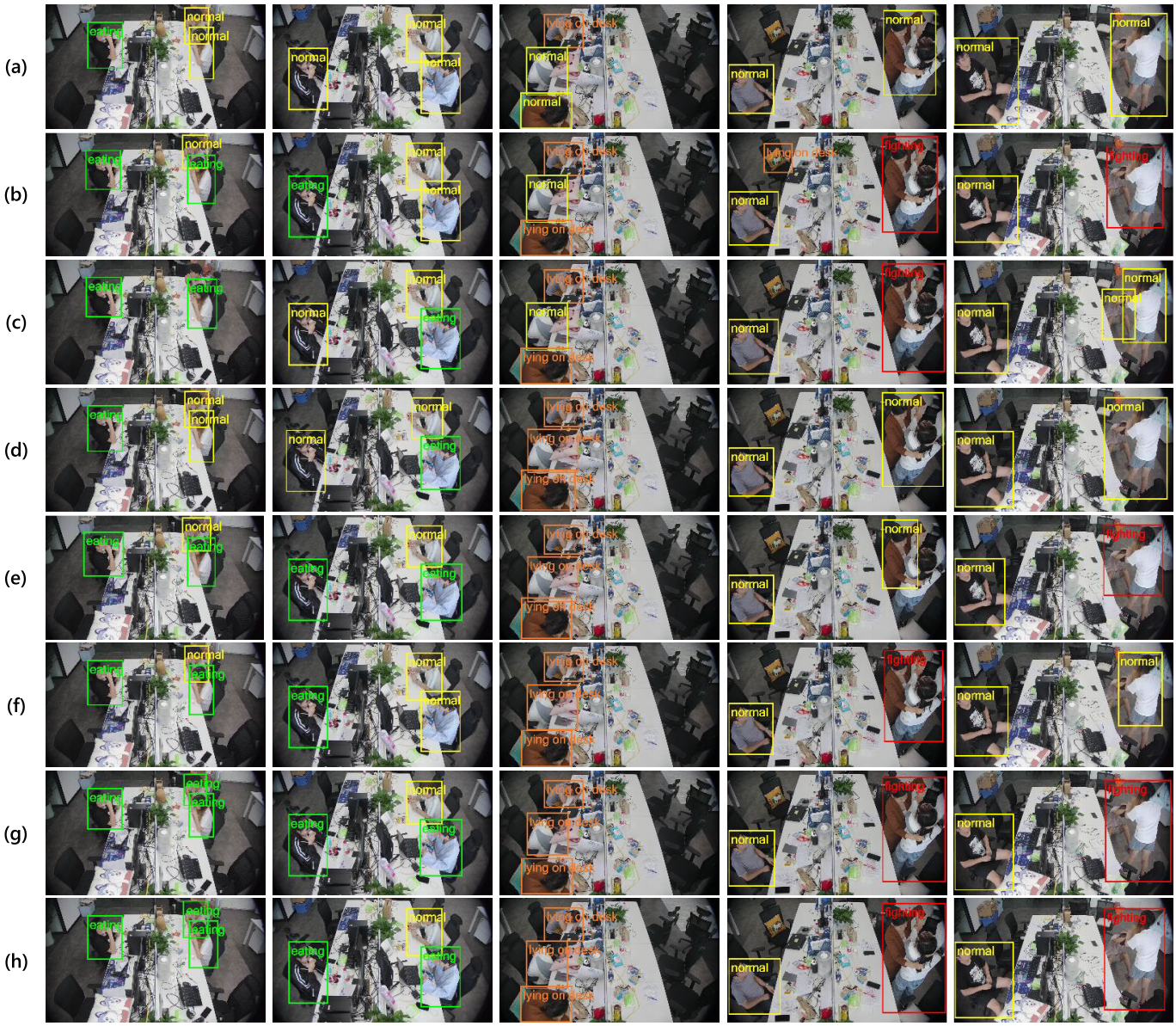

Fig. 6. Comparison of visualization results on the Office Behavior Dataset. (a) results by R*CNN [34]. (b) results by YOLOv3 [4]. (c) results by Multi-branch [35]. (d) results by Cascade R-CNN [11]. (e) results by ATSS [14]. (f) results by TOOD [24]. (g) results by AJENet(Ours). (h) ground-truth. Each abnormal behavior is labeled in the detected bounding box, where the detected category of abnormal behavior is labeled in the upper left corner of the box. The abnormal behavior 'eating' indicated by green, 'lying on desk' indicated by orange, 'fighting' indicated by red, and the 'normal' behavior indicated by yellow.

VI. DISCUSSIONS

To further demonstrate the reasonableness of AJENet, we first discuss the generalizability of AJENet under different generic object detection frameworks, then we discuss the structures of JP and AKJE on the Office Behavior Dataset.

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

Fig. 7. Comparison of visualization results on the PASCAL VOC 2012 Action Dataset. (a) results by Multi-branch [35]. (b) results by Cascade R-CNN [11]. (c) results by TOOD [24]. (d) results by AJENet(Ours). (e) ground-truth. Each behavior is labeled in the detected bounding box, where the detected category of behavior is labeled in the upper left corner of the box. The behavior 'phoning' indicated by orange, 'running' indicated by red, and the 'other' behavior indicated by purple.

TABLE III

RESULTS OF ABLATION EXPERIMENTS. AJENET CAN BE INTERPRETED AS 'BASE+JP+AKJE'. 'LYING' INDICATES THE 'LYING ON DESK' FOR BREVITY

Method	Office Behavior Dataset				VOC 2012 Action
	eating	lying	fighting	mAP	mAP
Base	0.683	0.847	0.777	0.769	0.880
Base+JP	0.702	0.868	0.792	0.787	0.887
Base+AKJE	0.715	0.860	0.790	0.788	0.891
AJENet	**0.723**	**0.868**	**0.796**	**0.796**	**0.898**

TABLE IV

RESULTS COMPARISON OF DIFFERENT DETECTION APPROACHES. 'LYING' INDICATES THE 'LYING ON DESK' FOR BREVITY

Method	Categories			mAP
	eating	lying	fighting	
RetinaNet [13]	0.683	0.847	0.777	0.769
RetinaNet+AJENet(Ours)	**0.723**	**0.868**	**0.796**	**0.796**
SSD [12]	0.509	0.720	0.677	0.635
SSD+AJENet(Ours)	**0.597**	**0.792**	**0.721**	**0.703**
YOLOv3 [4]	0.614	0.808	0.721	0.714
YOLOv3+AJENet(Ours)	**0.641**	**0.812**	**0.750**	**0.734**
Faster R-CNN [10]	0.720	0.853	0.789	0.787
Faster R-CNN+AJENet(Ours)	**0.733**	**0.864**	**0.807**	**0.801**
Cascade R-CNN [11]	0.727	0.860	0.801	0.796
Cascade R-CNN+AJENet(Ours)	**0.737**	**0.869**	**0.816**	**0.807**
FCOS [52]	0.679	0.845	0.774	0.766
FCOS+AJENet(Ours)	**0.720**	**0.866**	**0.795**	**0.794**
ATSS [14]	0.723	0.858	0.791	0.791
ATSS+AJENet(Ours)	**0.735**	**0.869**	**0.808**	**0.804**

A. Discussion About Different Detection Frameworks

To demonstrate the generalizability of AJENet under different frameworks, we add our proposed AJENet under different generic detection frameworks. As shown in the Tab. IV, the addition of the AJENet increases the model performance by 2.7%, 6.8%, 2.0%, 1.4%, 1.1%, 2.8%, and 1.3%. No matter what detection framework is used, our approach brings significant improvements. However, it is worth noting that with the use of a stronger detection framework, the effect of the AJENet is weakened. It is because the stronger framework has a stronger ability to extract features, the smaller the capacity of the AJENet can enhance. Among them, the addition of AJENet makes the performance improvement more obvious for the prominent joints information such as 'eating'. This also confirms the effectiveness of our methods.

B. Discussion About Joints Predict Block

In this section, we discuss the multi-scale structure, the loss functions used in the proposed JP block, the weights of α during training, and the categories of joints used.

1) Multi-Scale Structure: As described above in Sec. III-C.2, our proposed JP block can be adopted for multi-scale. To constrain the output joints on the appropriate feature layer, we choose different layers to join the JP for experiments, and the experimental results are shown in Tab. V. Experiments show that adding JP to all layers has the best performance. Besides, the experimental results show that compared with other branches, the performance of adding joints predict branches in the P7 layer is more obvious. It is because the joints' output from the P7 layer has a smaller scale. There is no need to predict very detailed joint positioning information and more guidance for joint classification. This further significantly improves the feature extraction capability of the model.

Besides, in order to constrain the output human joints at multi-scale structures using the proposed JointsLoss, we conduct related experiments on how to match the scale of the output joints and ground truth at different scales. The scale of the ground truth is larger than the scale of the output joints, the experimental method mainly includes scaling the output joints by deconvolution kernel or interpolation, and scaling the ground truth by connecting the convolution kernel or interpolation. The experimental results are shown in Tab. VI, the 'Base' denotes the method without the multi-scale structure. From the experimental results, the method of transforming the

Fig. 8. Image visualization of real/suspected abnormal behaviors in our collected Office Behavior Dataset and its detection results. (a) contains real 'fighting' and suspected 'fighting', (b) contains real 'lying on desk' and suspected 'lying on desk', and (c) contains real 'eating' and suspected 'eating'. All the images are sampled randomly. It can clearly show the diversity and difficulty of the office scenario dataset, while demonstrating effective classification and localization of behaviors in our AJENet.

TABLE V

RESULTS COMPARISON OF DIFFERENT JOINTS PREDICT SCALES. 'W/O' INDICATES WITHOUT

Layer	Categories			mAP
	eating	lying on desk	fighting	
w/o	0.683	0.847	0.777	0.769
P3	0.670	0.849	0.784	0.768
P4	0.667	0.848	0.787	0.767
P5	0.668	0.830	0.770	0.756
P6	0.644	0.839	0.787	0.757
P7	0.699	0.867	0.785	0.784
All layer	**0.702**	**0.868**	**0.792**	**0.787**

TABLE VI

RESULTS COMPARISON OF DIFFERENT MATCH MECHANISMS BETWEEN OUTPUT JOINTS AND GROUND TRUTH

Method	Categories			mAP
	eating	lying on desk	fighting	
Base	0.683	0.847	0.777	0.769
Base+output-deconv	0.669	0.857	0.788	0.771
Base+output-interpolate	0.651	0.850	0.762	0.754
Base+gt-conv	0.699	0.862	**0.792**	0.784
Base+gt-interpolate	**0.702**	**0.868**	**0.792**	**0.787**

ground truth scale has better performance than the method of transforming the output joints scale. Among them, the method of interpolating ground truth has the highest mAP. It is because the joints are the low-dimensional information, and the output joints are only different in scale, and there is no difference in feature space. Therefore, the interpolation method can achieve the goal. It is simpler, more effective, and can reduce the difficulty of model learning.

TABLE VII

RESULTS COMPARISON OF DIFFERENT LOSS FUNCTIONS IN JP BLOCK

Loss function	Categories			mAP
	eating	lying on desk	fighting	
MSELoss	**0.702**	0.857	0.780	0.780
L1Loss	0.683	0.847	0.771	0.767
SmoothL1Loss	0.680	0.857	0.779	0.772
FocalLoss [13]	0.668	0.858	0.785	0.770
WingLoss [54]	0.684	0.860	**0.793**	0.779
JointsLoss(Ours)	**0.702**	**0.868**	0.792	**0.787**

TABLE VIII

RESULTS COMPARISON OF DIFFERENT JOINTSLOSS WEIGHTS λ

Weight	Categories			mAP
	eating	lying on desk	fighting	
0	0.683	0.847	0.777	0.769
1	0.698	0.861	0.791	0.783
5	0.690	0.866	0.790	0.782
10	**0.702**	**0.868**	**0.792**	**0.787**
50	0.692	0.861	0.790	0.781
100	0.697	0.807	0.789	0.764

TABLE IX

RESULTS COMPARISON OF DIFFERENT JOINT TYPES

Joint type	Categories			mAP
	eating	lying on desk	fighting	
CrowdPose [55]	0.675	0.849	**0.793**	0.772
MHP [56]	0.635	0.836	0.780	0.750
COCO	**0.702**	**0.868**	0.792	**0.787**
COCO-WholeBody [57]	**0.702**	0.867	**0.793**	**0.787**

2) Loss Function: To verify the effectiveness of the proposed JointsLoss, we compare other loss functions commonly used to constrain joints. As shown in Tab. VII, the results indicate our proposed JointsLoss has the best performance. It demonstrates that JointsLoss can supervise JP more effectively and enable the model to attain better convergence. Besides, in JointsLoss, the use of mean squared deviation improved by 0.7% compared to MSELoss. It demonstrates that our designed mask generation mechanism and Gaussian filtering operation can effectively suppress the effect of the background region on the model and increase the number of positive samples, respectively. Thus, we balance the number of positive and negative samples during training and achieve better performance.

3) The Weights of λ: As described in Equ. 14 in Sec. III-F, to select the appropriate weight λ of JointsLoss, we perform the experiment by setting the weight values distributed in [0, 100]. As shown in Tab. VIII, the experiments show that when the loss of weight is 10, the mAP is the highest. It can be seen that when the λ is too small, the JP cannot achieve the purpose of optimizing feature extraction. On the contrary, if the λ is too large, the model deviates from the focus and puts more optimization on the joint prediction, which in turn has a negative impact on the classification and location of the object. Therefore, we set λ as 10 in the final model.

4) The Categories of Joints: To explore the effect of the categories of joint points, we experimented with four types of skeleton point types: CrowdPose [55][2] with 14 joint points,

[2]https://github.com/Jeff-sjtu/CrowdPose

TABLE X
RESULTS COMPARISON OF DIFFERENT FEATURE ENHANCEMENT SCALES. 'W/O' INDICATES WITHOUT

Layer	Categories			mAP
	eating	lying on desk	fighting	
w/o	0.683	0.847	0.777	0.769
P3	0.689	0.849	0.788	0.775
P4	0.687	0.834	0.783	0.768
P5	0.708	0.832	0.777	0.772
P6	**0.715**	0.858	0.789	0.787
P7	0.660	0.843	0.770	0.758
All layer	**0.715**	**0.860**	**0.790**	**0.788**

TABLE XI
RESULTS COMPARISON OF DIFFERENT ENHANCEMENT METHODS. 'W/O' INDICATES WITHOUT

Method	Categories			mAP
	eating	lying on desk	fighting	
w/o	0.683	0.847	0.777	0.769
SENet [36]	0.702	0.848	0.789	0.780
Non-local [42]	0.686	**0.860**	0.787	0.778
Self-attention [43]	0.685	0.845	**0.791**	0.774
CBAM [37]	0.685	0.852	0.788	0.775
AKJE(Ours)	**0.715**	**0.860**	0.790	**0.788**

MHP [55][3] with 16 joint points, COCO[4] with 17 joint points, and COCO-WholeBody [57][5] with 133 joint points. Each one contains a different number and category of joints. As shown in Tab. IX, the experiments show that using COCO and COCO-WholeBody [57] have the best performance. It is mainly because some behaviors in the Office Behavior Dataset focus on the joints of the face, and COCO and COCO-WholeBody pay more attention to the joints of the face than CrowdPose [55] and MHP [55], so the performance of COCO and COCO-WholeBody is better. In addition, based on the COCO, COCO-WholeBody has carried out a more detailed division of the human joints. Moreover, it is worth noting that we obtain the highest results for both experiments using two types of skeleton point, COCO and COCO-WholeBody [57], thanks to the adaptive importance of joint points in JP block that makes the enhanced features robust enough. However, with the increase of the categories of joint points, the parameters and calculation amount of the model increase correspondingly. Therefore, the trade-off between computational cost and performance, using COCO has better performance.

C. Discussion About Adaptive Key Joints Enhancement Block

In this section, we discuss the multi-scale structure of feature enhancement and the different enhancement mechanisms.

1) Multi-Scale Structure: As described above in Sec. III-D, our proposed AJJE block can be adopted for multi-scale. We choose different layers to join the AKJE block for experiments. As shown in Tab. X, the experiments show that adding AKJE to all layers has the best performance. Besides, the experimental results show that compared with other scales, the performance improvement of feature fusion in the P6 layer is more obvious. This is mainly related to the object scale, the objects in the Office Behavior Dataset are distributed on the P6 layer, so feature enhancement in this layer is more helpful for subsequent classification and localization tasks.

2) Feature Enhancement Mechanisms: To verify the effectiveness of the feature enhancement, we compare common attention methods [36], [37], [42], [43] for feature enhancement. As shown in Tab. XI, the experiments show that our proposed AKJE block performs better than other methods. It is mainly related to the fact that AKJE pays more attention to joints, and joint information is very important for behavior detection. In addition, the AKJE can adaptively learn more important joints. Therefore, compared with SENet [36] and CBAM [37], which directly uses joints for channel-by-channel enhancement, the performance is better.

[3]https://github.com/ZhaoJ9014/Multi-Human-Parsing
[4]https://cocodataset.org/#keypoints-2017
[5]https://github.com/jin-s13/COCO-WholeBody

VII. CONCLUSION

In this paper, we propose an Adaptive Joints Enhancement Network (AJENet) for abnormal behavior detection in office scenarios. AJENet enables the model to pay more attention to more recognizable joint features, which consists of two closely-related components, the Joints Predict (JP) block and the Adaptive Key Joints Enhancement (AKJE) block. These two blocks guide the network learning of behavior-related joints from both implicit and explicit perspectives, and enable the network to focus on essential joint features of the behavior. In our collected Office Behavior Dataset, AJENet gets the state-of-the-art performance of abnormal behavior detection in office scenarios and improved the mAP by a large margin. Further, based on the existing work, our work will continue in the following two directions. Firstly, enrich feature representation with combined modalities of joints. Secondly, extending the human joints to other multi-modality tasks.

REFERENCES

[1] V. M. Durand and D. H. Barlow, *Essentials of Abnormal Psychology*. Boston, MA, USA: Cengage, 2015.
[2] C. Dhiman and D. K. Vishwakarma, "A review of state-of-the-art techniques for abnormal human activity recognition," *Eng. Appl. Artif. Intell.*, vol. 77, pp. 21–45, Jan. 2019.
[3] X. Qiao et al., "Design of abnormal behavior detection system in the state grid business office," in *Proc. Int. Conf. Artif. Intell. Secur.* Cham, Switzerland: Springer, 2021, pp. 510–520.
[4] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*.
[5] K.-E. Ko and K.-B. Sim, "Deep convolutional framework for abnormal behavior detection in a smart surveillance system," *Eng. Appl. Artif. Intell.*, vol. 67, pp. 226–234, Jan. 2018.
[6] N. C. Tay, T. Connie, T. S. Ong, K. O. M. Goh, and P. S. Teh, "A robust abnormal behavior detection method using convolutional neural network," in *Computational Science and Technology*. Cham, Switzerland: Springer, 2019, pp. 37–47.
[7] H. Ji et al., "Human abnormal behavior detection method based on T-TINY-YOLO," in *Proc. 5th Int. Conf. Multimedia Image Process.*, Jan. 2020, pp. 1–5.
[8] A. Lentzas and D. Vrakas, "Non-intrusive human activity recognition and abnormal behavior detection on elderly people: A review," *Artif. Intell. Rev.*, vol. 53, no. 3, pp. 1975–2021, Mar. 2020.
[9] Y. Hao et al., "An end-to-end human abnormal behavior detection framework for crowd with mental disorders," *IEEE J. Biomed. Health Informat.*, vol. 26, no. 8, pp. 3618–3625, Aug. 2022.

[10] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[11] Z. Cai and N. Vasconcelos, "Cascade R-CNN: Delving into high quality object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6154–6162.

[12] W. Liu et al., "SSD: Single shot MultiBox detector," in *Proc. ECCV*, 2016, pp. 21–37.

[13] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 318–327, Feb. 2020.

[14] 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. 9756–9765.

[15] C. Cao, C. Lan, Y. Zhang, W. Zeng, H. Lu, and Y. Zhang, "Skeleton-based action recognition with gated convolutional neural networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 11, pp. 3247–3257, Nov. 2019.

[16] Y. Wang, L. Zhou, and Y. Qiao, "Temporal hallucinating for action recognition with few still images," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5314–5322.

[17] R. Gao, B. Xiong, and K. Grauman, "Im2Flow: Motion hallucination from static images for action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5937–5947.

[18] A. Akula, A. K. Shah, and R. Ghosh, "Deep learning approach for human action recognition in infrared images," *Cognit. Syst. Res.*, vol. 50, pp. 146–154, Aug. 2018.

[19] M.-T. Fang, Z.-J. Chen, K. Przystupa, T. Li, M. Majka, and O. Kochan, "Examination of abnormal behavior detection based on improved YOLOv3," *Electronics*, vol. 10, no. 2, p. 197, Jan. 2021.

[20] A. Mehmood, "LightAnomalyNet: A lightweight framework for efficient abnormal behavior detection," *Sensors*, vol. 21, no. 24, p. 8501, Dec. 2021.

[21] K. Rohit, K. Mistree, and J. Lavji, "A review on abnormal crowd behavior detection," in *Proc. Int. Conf. Innov. Inf., Embedded Commun. Syst. (ICIIECS)*, Mar. 2017, pp. 1–3.

[22] R. M. Alairaji, I. A. Aljazaery, and H. T. H. Alrikabi, "Abnormal behavior detection of students in the examination hall from surveillance videos," in *Advanced Computational Paradigms and Hybrid Intelligent Computing*. Cham, Switzerland: Springer, 2022, pp. 113–125.

[23] D. Zhang, J. Han, G. Cheng, and M.-H. Yang, "Weakly supervised object localization and detection: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 9, pp. 5866–5885, Sep. 2022.

[24] C. Feng, Y. Zhong, Y. Gao, M. R. Scott, and W. Huang, "TOOD: Task-aligned one-stage object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 3490–3499.

[25] Y. Li, Y. Xue, L. Li, X. Zhang, and X. Qian, "Domain adaptive box-supervised instance segmentation network for mitosis detection," *IEEE Trans. Med. Imag.*, vol. 41, no. 9, pp. 2469–2485, Sep. 2022.

[26] X. Li, S. Lai, and X. Qian, "DBCFace: Towards pure convolutional neural network face detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 4, pp. 1792–1804, Apr. 2022.

[27] C. Liu, Z. Da, Y. Liang, Y. Xue, G. Zhao, and X. Qian, "Product recognition for unmanned vending machines," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Jun. 29, 2022, doi: 10.1109/TNNLS.2022.3184075.

[28] C. Dhiman and D. K. Vishwakarma, "View-invariant deep architecture for human action recognition using two-stream motion and shape temporal dynamics," *IEEE Trans. Image Process.*, vol. 29, pp. 3835–3844, 2020.

[29] D. K. Vishwakarma and T. Singh, "A visual cognizance based multi-resolution descriptor for human action recognition using key pose," *AEU-Int. J. Electron. Commun.*, vol. 107, pp. 157–169, Jul. 2019.

[30] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Learning and transferring mid-level image representations using convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1717–1724.

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

[32] M. Hoai, "Regularized max pooling for image categorization," in *Proc. Brit. Mach. Vis. Conf.*, 2014.

[33] G. Gkioxari, R. Girshick, and J. Malik, "Actions and attributes from wholes and parts," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 2470–2478.

[34] G. Gkioxari, R. Girshick, and J. Malik, "Contextual action recognition with R CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1080–1088.

[35] S. Yan, J. S. Smith, W. Lu, and B. Zhang, "Multibranch attention networks for action recognition in still images," *IEEE Trans. Cognit. Develop. Syst.*, vol. 10, no. 4, pp. 1116–1125, Dec. 2018.

[36] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-excitation networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 8, pp. 2011–2023, Aug. 2020.

[37] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. ECCV*, 2018, pp. 3–19.

[38] C. Liu, Y. Liang, Y. Xue, X. Qian, and J. Fu, "Food and ingredient joint learning for fine-grained recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 6, pp. 2480–2493, Jun. 2021.

[39] H. Jin, S. Lai, and X. Qian, "Occlusion-sensitive person re-identification via attribute-based shift attention," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 4, pp. 2170–2185, Apr. 2022.

[40] P. Huang, J. Han, N. Liu, J. Ren, and D. Zhang, "Scribble-supervised video object segmentation," *IEEE/CAA J. Autom. Sinica*, vol. 9, no. 2, pp. 339–353, Feb. 2022.

[41] C. Fang, H. Tian, D. Zhang, Q. Zhang, J. Han, and J. Han, "Densely nested top-down flows for salient object detection," *Sci. China Inf. Sci.*, vol. 65, no. 8, Aug. 2022, Art. no. 182103.

[42] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7794–7803.

[43] A. Vaswanin et al., "Attention is all you need," in *Proc. NeurIPS*, vol. 30, 2017.

[44] L. Shi, Y. Zhang, J. Cheng, and H. Lu, "AdaSGN: Adapting joint number and model size for efficient skeleton-based action recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 13393–13402.

[45] Z. Tu, J. Zhang, H. Li, Y. Chen, and J. Yuan, "Joint-bone fusion graph convolutional network for semi-supervised skeleton action recognition," *IEEE Trans. Multimedia*, vol. 25, pp. 1819–1831, 2022.

[46] H. Duan, Y. Zhao, K. Chen, D. Lin, and B. Dai, "Revisiting skeleton-based action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 2959–2968.

[47] J. Wang et al., "Deep high-resolution representation learning for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 10, pp. 3349–3364, Oct. 2021.

[48] B. Cheng, B. Xiao, J. Wang, H. Shi, T. S. Huang, and L. Zhang, "HigherHRNet: Scale-aware representation learning for bottom-up human pose estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 5385–5394.

[49] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. (2012). *The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results*. [Online]. Available: http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html

[50] S. Maji, L. Bourdev, and J. Malik, "Action recognition from a distributed representation of pose and appearance," in *Proc. CVPR*, Jun. 2011, pp. 3177–3184.

[51] 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.

[52] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: Fully convolutional one-stage object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9626–9635.

[53] 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.

[54] Z.-H. Feng, J. Kittler, M. Awais, P. Huber, and X.-J. Wu, "Wing loss for robust facial landmark localisation with convolutional neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2235–2245.

[55] J. Li, C. Wang, H. Zhu, Y. Mao, H.-S. Fang, and C. Lu, "CrowdPose: Efficient crowded scenes pose estimation and a new benchmark," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 10855–10864.

[56] J. Zhao, J. Li, Y. Cheng, T. Sim, S. Yan, and J. Feng, "Understanding humans in crowded scenes: Deep nested adversarial learning and a new benchmark for multi-human parsing," in *Proc. 26th ACM Int. Conf. Multimedia*, Oct. 2018, pp. 792–800.

[57] S. Jin et al., "Whole-body human pose estimation in the wild," in *Proc. ECCV*, 2020, pp. 196–214.