IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 29 October 2024, accepted 7 December 2024, date of publication 12 December 2024, date of current version 23 December 2024.

*Digital Object Identifier 10.1109/ACCESS.2024.3516586*

RESEARCH ARTICLE

# U-ActionNet: Dual-Pathway Fourier Networks With Region-of-Interest Module for Efficient Action Recognition in UAV Surveillance

**ABDUL MONAF CHOWDHURY[1], AHSAN IMRAN[1], MD MEHEDI HASAN[1], RIAD AHMED[2], AKM AZAD[3], AND SALEM A. ALYAMI[3], (Member, IEEE)**

[1]Department of Robotics and Mechatronics Engineering, University of Dhaka, Dhaka 1000, Bangladesh
[2]Department of Computer Science and Engineering, BRAC University, Dhaka 1212, Bangladesh
[3]Department of Mathematics and Statistics, Faculty of Science, Imam Mohammad Ibn Saud Islamic University (IMSIU), Riyadh 13318, Saudi Arabia

Corresponding author: Md Mehedi Hasan (mmhasan@du.ac.bd)

**ABSTRACT** Unmanned Aerial Vehicles (UAV) have revolutionized human action recognition by offering a bird's-eye perspective, thereby unlocking unprecedented potential for comprehensive support within surveillance systems. This paper presents synergistic strategies for enhancing Human Action Recognition (HAR) in UAV imagery. Leveraging a dual-path approach, we propose a novel framework U-ActionNet that integrates FFT-based substance separation and space-time self-attention techniques to improve the accuracy and efficiency of HAR tasks. Catering the server side the first pathway employs a modified C3D model with Fast Fourier Transform (FFT)-based object movement and attention detection mechanisms to effectively extract human actors from complex scenes and capture spatiotemporal dynamics in UAV footage. Moreover, a generalized Region-of-Interest (ROI) module is utilized to concentrate on optimal regions to enhance target recognition. Through extensive experiments on the Drone Action and UAV-human datasets, we demonstrate the effectiveness of our approach, achieving superior performance compared to the state-of-the-art methods with Top-1 accuracies of 94.94% and 95.05%, respectively. Meanwhile, the second pathway employs edge-optimized models, integrating ROI extraction-based frame sampling technique to eliminate static frames while preserving pivotal frames essential for model training. We introduce a lightweight model named U-ActionNet Light which is a combination of MobileNetV2, Fourier module, and BiLSTM models that demand only one-ninth of the parameters compared to the server-side model. Demonstrating its efficacy, this model attains Top-1 accuracies of 80.43% and 84.74% on the same datasets, surpassing the baseline by a significant margin. Experimental results show that the presented frameworks are promising for surveillance, search and rescue, and activity monitoring, where accurate and real-time human action recognition from UAV platforms is essential.

**INDEX TERMS** FFT, action recognition, video surveillance, tracking, optimization.

## I. INTRODUCTION

Unmanned Aerial Vehicles (UAVs) have become a revolutionary tool in the field of human action recognition (HAR), providing unique viewpoints and enabling enhanced surveillance, search and rescue operations, and activity monitoring. By leveraging high-resolution cameras and sophisticated algorithms, UAVs can capture and analyze human movements from a vantage point that ground-based systems cannot achieve. Figure 1 illustrates a conventional workflow diagram of Human Action Recognition (HAR) using UAV systems.

Understanding and interpreting human behaviour has become increasingly crucial in today's rapidly evolving world. Accurate identification and recognition of human actions from video data hold significant potential across a spectrum of applications, spanning from advanced surveillance systems to the realms of social robotics. Traditionally,

The associate editor coordinating the review of this manuscript and approving it for publication was Long Xu.

© 2024 The Authors. This work is licensed under a Creative Commons Attribution 4.0 License.
For more information, see https://creativecommons.org/licenses/by/4.0/

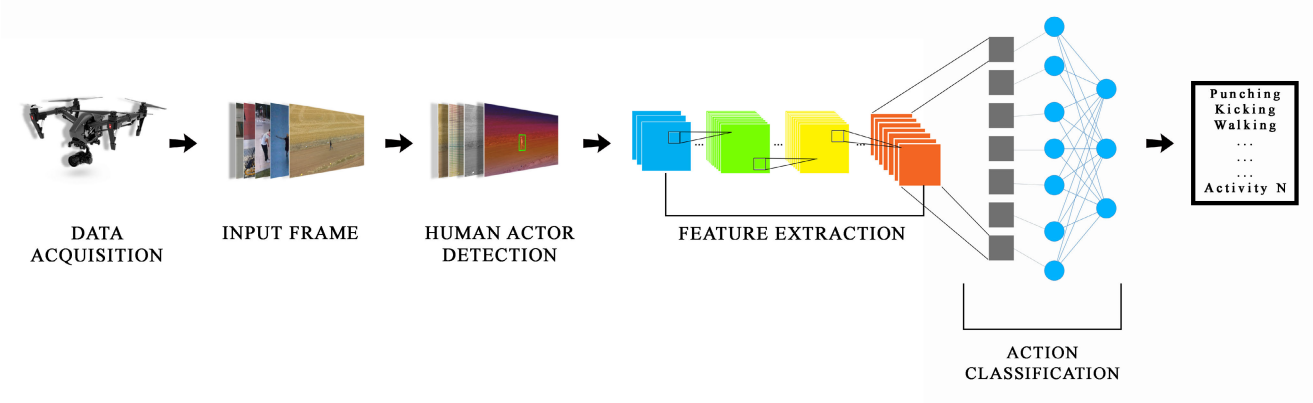



**FIGURE 1.** Diagram illustrates the flow of Human Action Recognition (HAR) employing Unmanned Aerial Vehicle (UAV) technology in surveillance systems.

action recognition methods have relied on fixed camera setups or wearable sensors, which impose limitations in terms of mobility, coverage, and flexibility. Several deep neural network methods [1], [2], [3] have been comprehensively used for ground-based video footage [4], [5] for human activity detection. In spite of their impressive performance on ground videos, their action recognition accuracy significantly drops when applied to UAV videos.

Inflated 3D ConvNet (I3D) is a state-of-the-art action recognition system and powerful deep learning architecture capable of capturing spatiotemporal features from video data, making it well-suited for analyzing surveillance footage [1]. I3D reached 98.0% and 80.9% respectively on two popular ground-based action datasets, UCF-101 [6] and HMDB-51 [7]. However, it only achieved 23.86%, 28.7%, and 16.8% on three aerial action recognition datasets. These datasets were UAV-Human [8], TinyVIRAT [9], and UCF-Aerial. This performance drop is mainly caused by the domain shift brought on by various camera settings and perspective changes. Humans occupy less than 10% of the spatial area in the UAV videos [8], [10] which evidently provides less information than front-view footage captured by ground cameras. Furthermore, the UAV's constant movement may cause the camera's viewing angle to fluctuate, causing blurred motion and occlusion. All these issues combined make it difficult to design robust algorithms for aerial action recognition. Figure 2, provides some illustrations of the performance decline that can be expected from I3D across various datasets.

The neural network models [2], [3] learn more about the background [11] than the human actor due to the size imbalance in UAV videos. While both contextual and background information is essential, it is crucial for the network to differentiate the moving human participant from the background. An effective method to identify humans is by employing the object detection approach [12]. Action detection models depend immensely on the localization of human actors and it is almost impossible to annotate all datasets for object detection. Ground camera videos are

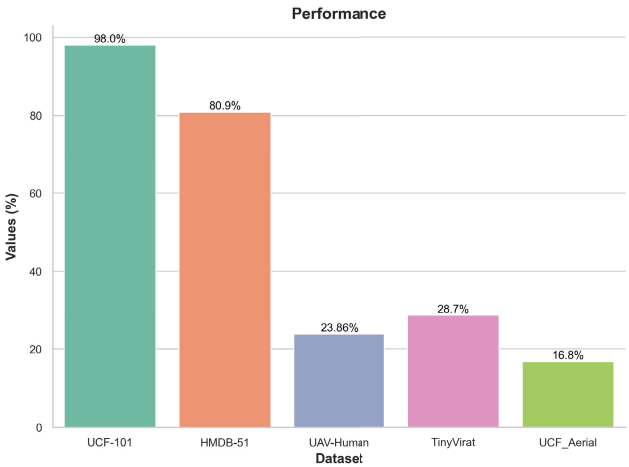



**FIGURE 2.** When switching to aerial data, even the most advanced generic action detection systems perform much worse. The figure contrasts I3D's accuracy on three aerial (UAV-Human, TinyVIRAT, and UCF-Aerial) and two ground-based (UCF101 and HMDB-51) datasets.

unlikely to generalise well because of domain gap issues, as discussed in [13] and [14].

In addition to the task of detecting and separating objects from their backgrounds, the network must also acquire knowledge about the context, object relationships, background associations, and intra-pixel correspondences. The information may be represented using self-attention, as described in the works of Vaswani et al. [15] and Zhang et al. [16]. Self-attention allows for the identification of long-range relationships within an image or video. Previous research on activity recognition [17] employed two different approaches of self-attention for analysis. The network either employed self-attention as a fundamental component or utilized it on the convolutional layers. The calculation of self-attention is computationally intensive due to the large number of matrix multiplications.

Moreover, the majority of existing action recognition methods need substantial memory or power resources.

Consequently, they are usually developed for cloud GPUs or desktop devices. They cannot be used with UAV platforms, mobile devices, or edge devices that have limited memory and lack the capability to do complex mathematical calculations. Presently available mobile platform-based action recognition techniques [4], [9] are not very effective when applied to aerial data because they are designed for ground camera videos. Therefore, better methods are thus required for the identification of actions from UAVs.

Our study introduces two novel deep-learning approaches for identifying actions in UAV videos in order to tackle these concerns. One of the models is U-ActionNet designed for server-side implementation and employs a modified version of C3D as its underlying backbone. It integrates FFT-based substance separation with space-time self-attention mechanisms that significantly improve the accuracy and robustness of action recognition in dynamic environments. Fourier space-time action recognition encodes long-range space-time correlations and separates human actors from the background. By employing FFT-based substance separation, the system effectively isolates relevant motion patterns, reducing noise and enhancing the quality of the data for the recognition process. Given its substantial weight, this model is better suited for backend activities that could handle strong processing power to accurately and precisely identify actions.

On the other hand, U-ActionNet Light is a simplified model designed to reduce computational complexity, making it suitable for edge devices with limited resources. By using region of interest extraction-based frame sampling to remove static frames, it selects only apposite frames for processing, which lowers computational complexity. The lightweight model extracts features and captures contextual information using MobileNetV2, Fourier module, and BiLSTM, with fewer trainable parameters, reducing computational resource needs. This efficient architecture supports the goal of minimizing computational overhead in resource-constrained environments.

The subsequent sections of the manuscript are organized in the following order: Section II provides a literature review that examines relevant methodologies and their limitations. The methodology along with design and mathematical formulations are illustrated in Section III. The experimental results, analysis and comparative studies are outlined in Section IV. Finally, the paper concludes by summarizing the main findings, limitations and their implications for UAV-based human action recognition in Section V.

## II. LITERATURE REVIEW

Activity recognition has benefited immensely from the application of deep learning techniques [3], [18]. Large-scale ground-camera video datasets [5] have emerged, which has prompted the development of deep learning methods for action recognition. For action recognition, three general classes of network designs have been recommended. The first models space and time independently, building on the two streams theory [19], [20], [21]. The second [1], [2], [3] uses 3D CNNs to jointly simulate time and space. Transformer-based architectures are included in the third class [22], [23]. These transformer-based methods have high computational complexity and are based on self-attention [15], [16]. Frame sampling algorithms [24], [25] for video action recognition have been presented with the goal of optimising GPU memory. The problems with action identification in ground-camera videos are the main emphasis of the aforementioned methods. On the other hand, UAV video action identification is far more challenging.

To generate solutions for UAV-based action recognition [26], UAV video databases have been utilized [8], [10]. These solutions aren't reliable in UAV videos as they are developed initially keeping in mind ground camera datasets [27], [28]. Nevertheless, considerable works have been done which were designed particularly for high-altitude action recognition.

Ahmad et al. [29] integrated YOLOv5 with a gradient-boosting classifier [30]. The authors tested different YOLOv5 architectures and evaluated the system's performance on the Okutama-Action dataset [31] for detecting and recognizing actions in drone videos. The model outperformed other methods for individual and primary/secondary actions but as the altitude increased system performance degraded. Along with this, due to motion blurring caused by the drone speed, algorithm performance dropped even further. To ensure competitive performance in low lighting and extreme weather conditions which was not possible in [29], Xian et al. [32] came up with a novel approach for action recognition in UAV videos. They addressed challenges such as occlusion, viewpoint changes, and small human actor size, using mutual information to align regions corresponding to human actions in the temporal domain. However, to reduce computational overhead it was assumed that there were no long-range spatial-temporal relationships between the human actor and the background.

Most deep learning techniques use large and deep architectures for better accuracy, but they require significant computational power. Thus, they are unsuitable for edge or low-power mobile devices and are instead used on desktop or cloud GPUs. Optimizing system architecture is essential due to the high computational cost of video analysis in UAV footage. There is growing interest in using lightweight techniques to enhance video models' performance [33], [34] [35]. YOLO [36] and MobileNet [37] are notable architectures for video modelling on edge devices. Kondratyuk et al. [4] introduced the family of memory and computationally efficient Mobile Video Networks (MoViNets), which are capable of operating on video streams for real-time inference. Lin et al. presented [38] the Temporal Shift Module (TSM), which supports both online and offline video recognition by shifting each channel within the temporal dimension.

Wang et al. introduced a novel approach [39] using auto zoom and temporal reasoning to enhance accuracy and reduce computational overhead. The framework which depends on the localization method, achieved improvements on various

datasets while the design mainly targeted both edge devices and high-end GPUs. To optimize for edge devices it is necessary to discard irrelevant frames and only work with informative frames as represented by Xian et al. in [40]. Their novel algorithm adaptively selects informative frames by utilizing patch mutual information score, shifted leaky ReLu, and cumulative distribution function to ensure comprehensive coverage of essential segments with high motion salience. Although it is robust to background noises and effectively distinguishes motion salient frames, it has limitations in consistent motion scenarios and during motion static periods.

Fast Fourier Transform (FFT) has lately been utilized for image [41] and video analysis [42] via deep learning systems. To increase accuracy for image detection, classification, and ground-camera action recognition, FFT has been utilised across neural network layers [43] in place of CNNs to transfer feature space into the frequency domain and facilitate the general mixing of information. In recent times, FAR [44] developed modifying feature maps using a frequency-based technique to separate moving objects. The authors separated the human agent from the background and captured contextual information and long-range dependencies. FAR transforms feature maps, preventing the network from learning disentangled features in a differentiable way. Inevitably, Kothandaraman et al. introduced a novel strategy [45] utilizing frequency domain representations. They built a differentiable static-dynamic frequency mask which empowered the network to learn isolated features. Even though the method is computationally expensive and requires multiple GPUs, it achieved significant improvement in HAR for drone footage.

However, the existing methods did not consider the Region-of-Interest (ROI) selection which has a great potential for improving the performance of video action classification. Therefore, we introduce an ROI module in the pre-processing stage in developing classification models. Based on this idea, we present our approaches as follows:

- **Serverside Model:** A reliable serverside model named U-ActionNet that can detect human actions from UAV footages in demanding scenarios in a manner that is flexible, steady, and adaptable for surveillance. The method demonstrates the ability to significantly improve recognition accuracy and efficiency by identifying the human actors and eliminating irrelevant background scenarios through the implementation of Fourier space-time action recognition using a modified version of C3D as its backbone.
- **Lightweight Model:** We developed a lightweight model named U-ActionNet Light for action recognition so that it can be deployed on edge devices requiring little computation. We utilized MobileNetV2, Fourier module, and BiLSTM to comprehend human actions. Our system can identify human actions in real time without dropping significant accuracy.

## III. METHODOLOGY

### A. DATA CURATION & DATASET PREPARATION

We explored two state-of-the-art datasets namely UAV Human [8] and Drone Action [10] which are specifically curated for human action recognition using UAVs. The experiments encompassed different scenarios, action classes, and environmental conditions to ensure the robustness and generalizability of the proposed systems.

#### 1) UAV HUMAN

UAV Human [8] provides a new benchmark for human behaviour understanding with UAVs which contain 119 subjects and 67,428 multi-modal video sequences for action recognition, 22,263 frames for attribute recognition, 1,144 identities and 41,290 frames for person re-identification, and 22,476 frames for pose estimation. The dataset was acquired over the course of three months by a flying UAV in numerous urban and rural districts during both daytime and nighttime. As a result, it covers a wide variety of subject matter, backgrounds, illuminations, weather conditions, camera motions, UAV flying attitudes, and occlusions.




**FIGURE 3.** UAV-Human dataset example videos demonstrating action [8].

In our study of human action recognition using UAVs, we encountered a challenge with the size of the dataset. The dataset available for this task was extremely large and contained 155 action classes. However, we realized that using the entire dataset for our specific research focus would be unnecessary and could potentially lead to increased computational complexity and training time. To address this issue, we made a strategic decision to narrow down the dataset and focus on a subset of classes that were particularly relevant to our research objective. Specifically, we selected ten action classes that were characterized by their association with violent actions. By limiting our analysis to these ten selected classes, we aimed to lay the groundwork for action recognition in the context of UAV surveillance.

The selected 10 action classes were: punching someone, kicking someone, pushing someone, slap someone on the back, threat someone with a knife, hold someone hostage, threat someone with a gun, drag someone, call for help, and stab someone with a knife. Here, the dataset consisted of a total of 10 action classes, with each class containing 30 videos. To ensure an appropriate division of the dataset for training and testing purposes, a split ratio of 80:20 was employed. This means that 80% of the videos within each class, amounting to 24 videos, were utilised for training the action recognition model, while the remaining 20%, which corresponded to 6 videos, were reserved for testing and evaluating the performance of the model. By employing this split strategy, a total of 240 videos were allocated for training the model.

#### 2) DRONE ACTION

Drone Action [10] is an action recognition dataset that was captured in an outdoor environment in order to close this gap and facilitate research in more general application areas. Thirteen dynamic human behaviours were recorded using a free-flying drone. There are 240 high-resolution video clips totalling 66,919 frames in the collection. To get the best possible resolution and maximal human pose details, all of the videos were shot at low altitudes and slow motion.




**FIGURE 4. Drone action dataset's action classes. To ensure a clear example, the photographs were cropped surrounding the subject and chosen at random [10].**

For the Drone Action dataset, we strategically chose to reduce the dataset and concentrate on a subset of classes that were especially pertinent to our study goal. In particular, we chose ten action groups that were distinguished by their correlation with aggressive behaviours. Since violence is a significant factor to take into account in UAV-based surveillance scenarios, these classes were selected based on their commonalities and applicability to violence.

The selected 10 action classes were: Walking, jogging, running, hitting with bottle, hitting with stick, stabbing, punching, kicking, clapping, and waving hands. Here, the dataset consisted of a total of 10 action classes, with each class containing 10 videos. To ensure an appropriate division of the dataset for training and testing purposes, a split ratio of 80:20 was employed. This means that 80% of the videos within each class, amounting to 8 videos, were utilised for training the action recognition model, while the remaining 20%, which corresponded to 2 videos, were reserved for testing the performance of the model. By employing this split strategy, a total of 100 videos were allocated for training the model.

The methodology consists of two discrete segments, each of which is customized to fulfil particular computational demands. The initial portion of the methodology, which centers on the construction of a resilient and comprehensive model that is optimized for execution on the server, will be elaborated in the beginning. The primary objective of this robust model is to leverage the computational capabilities of centralized servers to facilitate complex analysis and detailed interpretation of human actions captured in UAV videos.

### B. SERVER-SIDE MODEL

The Server-side system is designed to deal with heavy workloads and leverage the available computational power for human action detection. Figure 5 illustrates the dynamics of the system. Initially, input video frames are processed using YOLOv8 [46] for object localization. The localized objects are passed through modified C3D for feature extraction. Subsequently, these features are processed through two Fourier blocks in parallel. The Fourier Substance Separation block isolates human actors from the background, while the Fourier Self-Attention block captures spatiotemporal relationships. Eventually, the high-level features are fed through dense layers for human action recognition. The following sections provide a detailed explanation of each stage.

#### 1) REGION-OF-INTEREST

In the domain of UAV-based action recognition, a significant portion of UAV data frames is inundated with background noise. This clutter is often extraneous to the specific tasks and objectives at hand. In essence, it does not contribute to finding the causal effect of any action on the design and training of neural network models aimed at classifying and recognizing certain tasks or activities.

To deal with this issue, we used the ''Region-of-Interest'' (ROI) strategy to declutter the intricate background and zoom in on the 10% to 15% region that is the actual cause of the action. Figure 6 illustrates the process for extracting the Region-of-Interest. Each video file is passed through the YOLOv8 [46] model which is trained on the COCO dataset [47]. As a result, the pre-trained model YOLOv8x, which is used here, is capable of detecting 80 classes of objects in the frame. As we are dealing with human activity recognition using UAV footage, varieties of object classes other than humans are unnecessary. Consequently, a mask is created and using that the unnecessary objects are filtered out so that the detections only contain all the humans in the frame with appropriate confidence scores. In a single frame, there

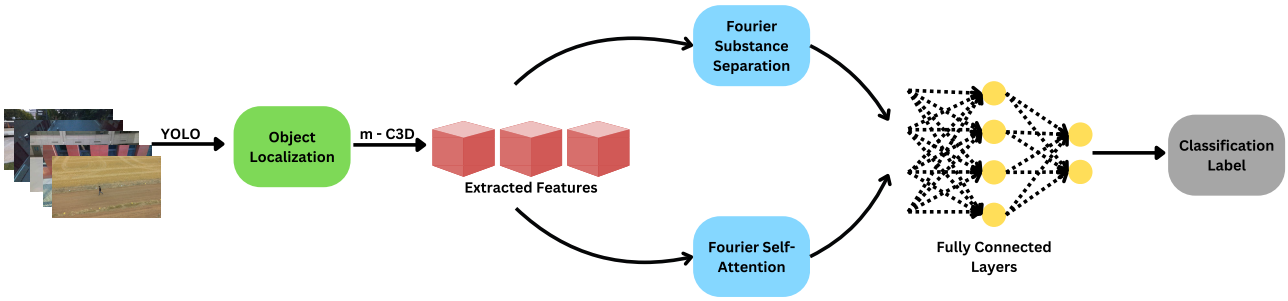



**FIGURE 5.** Proposed U-ActionNet architecture incorporates an object localization block (green), m-C3D block for feature extraction (red), Fourier substance separation block (blue), Fourier Self-Attention block (blue), and two dense layers (yellow).

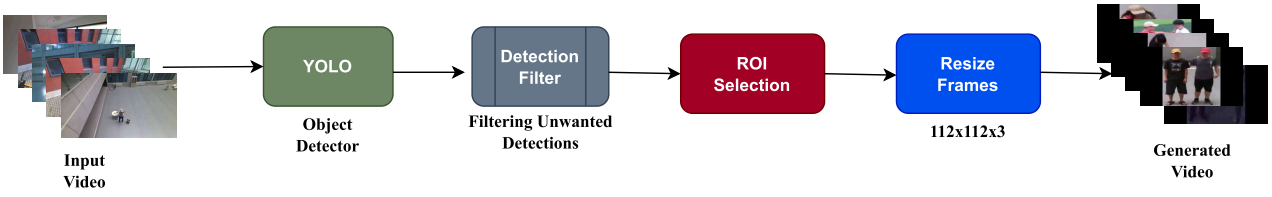



**FIGURE 6.** Region-of-Interest module incorporates YOLOv8 (green) as object detector, Detection filter (grey) for removing unwanted detections, Region-of-Interest selection (red), Resize frames (blue) to 112 × 112 × 3, and finally generated video with only Region-of-Interest.

can be multiple humans and to determine the ideal person of interest, filtering was done based on two criteria [48]. If detected persons' are consecutively present in 32 frames and have detection confidence scores continuously above 0.60, bounding boxes are generated around them signalling the Region-of-Interest for that particular frame. The bounding boxes are resized to $112 \times 112 \times 3$ to ensure uniformity and to simplify subsequent processing steps. All of the regions of interest from each frame of particular videos are joined together to generate videos that will serve as the foundational dataset for the human activity recognition model.

### 2) MODIFIED CONVOLUTIONAL 3D (m-C3D) MODEL

C3D (Convolutional 3D) [49] is a state-of-the-art deep learning model for video analysis and classification. C3D architecture is organized into several layer groups, each comprising convolutional layers followed by max pooling operations, and employs the rectified linear unit (ReLU) activation function to introduce non-linearity.

To optimize human action recognition, we have modified C3D by incorporating the Fourier module after the sixth layer of the C3D network and redesigned the two final dense layers. The Fourier module is divided into two parts and both of them run simultaneously. This module isolates human actors from the background and remembers contextual information regarding the action.

By using the results up to the sixth layer of C3D, we capture more general features that are beneficial for classification while avoiding potential overfitting issues. Moreover, dropout regularization with a rate of 0.5 is applied after each fully connected layer, promoting generalization and enhancing model robustness.

### 3) FOURIER SUBSTANCE SEPARATION

One part of the Fourier module is the Fourier Substance Separation (FSS) which automatically isolates the human actor from the background in surveillance scenarios. To detect and characterize movement, we transformed the lower-level feature maps generated from the sixth layer of m-C3D into the temporal frequency space. This transformation allows us to examine the signal's behaviour across different temporal frequencies and extract valuable information regarding the presence of movement in the scene.

In our approach, we utilize a 1D Fourier transform along the temporal dimension to perform the necessary computations. Let feature maps be represented by $f(c, t, h, w) \in C \times T' \times H' \times W'$ on which the Fourier Substance Separation (FSS) method is applied. Here, $C$ denotes the number of channels, while $T'$ and $(H' \times W')$ denote the temporal and spatial dimensions of the feature maps, respectively.

At frequency $-2\pi k/N$, the amplitude of the temporal Fourier transform is calculated as follows:

$$\mathcal{A}_{\mathcal{T}}(f)(k) = \sum_{n=0}^{n=T'} f(c, t, h, w) \times e^{-2\pi kn/N}. \tag{1}$$

To efficiently compute this transform, we employ the Fast Fourier Transform (FFT) algorithm [50], which provides an optimized solution for this task.

For each spatial and channel location in the feature map $f$, the amplitude of the temporal signal is captured mathematically as $\mathcal{A}_{\mathcal{T}}(f)(k)$. To simply put, higher frequencies in the temporal dimension correspond to movement, while lower frequencies indicate static regions in the scene. Consequently, areas associated with the human actor's motion should

exhibit higher amplitudes in the Fourier transform at higher frequencies.

To locate the moving human actor in multiple locations, we build the frequency-amplitude relationship by multiplying the squared L2-norm of the frequency with the squared L2-norm of the amplitude at each frequency. L2-norm guarantees positive frequencies and amplitudes. By squaring the L2-norm, we emphasize higher amplitudes in the Fourier transform at high frequencies and attenuate lower amplitudes at low frequencies. This process aids in separating dynamic regions of the scene that are associated with motion. The frequencies are: $fr_k = [e^{-2\pi k/N}]$, $k = 1 \ldots .T'$, respectively.

It is important to note that the frequencies utilized in the Fourier Substance Separation (FSS) technique are independent of the input video. Consequently, we can express the dynamic mask $\mathcal{D}_{FS}$ as:

$$D_{FS} = \|\mathcal{A}_{\mathcal{T}}(f)(k)\|_2^2 \times \|fr_k\|_2^2 \tag{2}$$

where $\|fr_k\|_2^2$ denotes the squared L2-norm of a vector. The dynamic mask $D_{FS}$ serves to separate or amplify the regions in the scene that correspond to moving pixels. It is important to note that these regions may include both the moving background (including camera motion) and the moving human actor. Our subsequent task involves using $D_{FS}$ to differentiate the moving object pixels from the moving background pixels.

We use the model's activation maps $f$ to isolate the moving actor. Although not flawless, these activations tend to be higher in salient regions of the scene compared to non-salient regions. As a result, the final representation of the separated object can be obtained by taking the dot product of $D_{FS}$ and the network features $f$, thereby amplifying the dynamic and salient regions of the scene. Mathematically,

$$\nabla_{FS} = f \odot D_{FS}. \tag{3}$$

The separated substance representation, denoted as $\nabla_{FS}$, can be obtained by element-wise multiplication (Hadamard product) of the network features $f$ and the dynamic mask $D_{FS}$, as shown in Equation 3. This formulation results in the amplification of dynamic salient regions while heavily suppressing static non-salient regions. Therefore, eventual representations of disentangled objects have the following order of amplitudes: static-non-salient < dynamic-non-salient < static-salient < dynamic-salient. This hierarchy mirrors the importance of different regions in action recognition for decision-making.

#### 4) FOURIER SELF-ATTENTION

Space-time self-attention has been proven effective in extracting knowledge for video action recognition. We have used Fourier Self-Attention (FA) to gain insights into the long-range space-time relationships present in video data. FA offers an efficient approximation of self-attention by leveraging the power of Fourier transformation in a computationally efficient manner [16].

The key, query, and value components in the self-attention mechanism are obtained through $1 \times 1$ convolution layers applied to the input feature maps. According to Vaswani et al. [15], self-attention is computed by adding up the weights allocated to each value based on a compatibility function that evaluates how closely the query and the key matchup are. This compatibility function plays a crucial role in determining the relevance of each value for the given query. Mathematically, let x denote the input feature maps, and $\odot$ represent matrix multiplication. The attention computation can be expressed as follows:

$$\text{Attention} = \text{Value(x)} \odot [\text{Query(x)}^T \odot \text{Key(x)}]^T \tag{4}$$

The Fourier self-attention method operates in the following manner. Initially, a representation analogous to the key-query computation is obtained, referred to as the *Fourier sub-attention*. The concept of Fourier sub-attention draws inspiration from autocorrelation, which quantifies the correlation coefficient between different segments of a given signal.

Sub-attention in the Fourier domain is defined as the element-wise product of the complex conjugate of the Fourier transform of the feature maps with the original feature maps. To determine the space-time Fourier sub-attention, the video feature maps $f$ are translated to the frequency domain through a 2D Fourier transform along the spatial and temporal axes, yielding a 3D representation $C \times T' \times (HW)$. This transformation is performed using the equation:

$$\mathcal{T}_{\mathcal{ST}}(f)(k,l) = \sum_{h,w} f(c,t,h,w) e^{-2\pi kh/K} e^{-2\pi lw/L}, \tag{5}$$

where $k$ and $l$ represent the frequency indices, and $K$ and $L$ are the dimensions of the spatial and temporal axes respectively.

FFT allows for extensive global interactions between distinct spatial and temporal regions in the video by expressing the signal across a wide range of frequencies. Multiplying $\mathcal{T}_{\mathcal{ST}}$ by its complex conjugate $\mathcal{T}_{\mathcal{ST}}{}^*$ yields the space-time Fourier sub-attention $\mathcal{F}_{\mathcal{ST}}$ in the Fourier domain, as shown in Equation 6:

$$\mathcal{F}_{\mathcal{ST}} = \mathcal{T}_{\mathcal{ST}} \times \mathcal{T}_{\mathcal{ST}}{}^* \tag{6}$$

To obtain the correlations in the time domain, we compute the inverse Fast Fourier Transform ($\mathcal{IF}$) of the space-time Fourier sub-attention $\mathcal{F}_{\mathcal{ST}}$. The resulting correlation maps are then reshaped to match the dimensions of the input feature maps ($C \times T' \times H' \times W'$).

The final space-time Fourier attention maps $\psi_{FA}$ are then computed by taking these sub-attention "weights" and using them in a dot product with the input feature maps $f$. To scale the Fourier attention maps, a scaling factor $\lambda_{FA}$ is empirically chosen to be 0.01, which is then fused with the input feature maps. The following equation describes the combination of the input feature maps and the scaled attention maps:

$$\psi_{FA} = \mathcal{IF}(\mathcal{F}_{\mathcal{ST}}) \times \lambda_{FA} + F \tag{7}$$

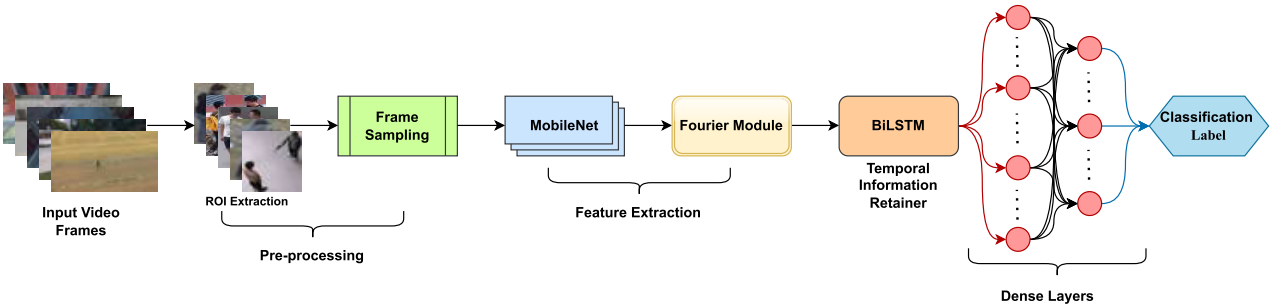



**FIGURE 7.** Proposed U-ActionNet light architecture incorporates Pre-processing block (ROI extraction and frame sampling), Feature extraction block (MobileNet and fourier module), Temporal information retainer, and a few dense layers.

#### *a: TIME COMPLEXITY*

Traditional self-attention [15] involves two matrix multiplications. The query matrix ($THW \times C$) is multiplied by the key matrix ($C \times THW$), resulting in a time complexity of $O(C \times THW \times THW)$. The value matrix ($C \times HWT$) is then multiplied by the attention matrix ($HWT \times HWT$), also with a time complexity of $O(C \times HWT \times HWT)$. Thus, the overall time complexity for space-time self-attention is $O(HWT \times HWT \times C)$ [17].

In contrast, our Fourier attention employs one 2D FFT and one 2D iFFT to address this issue. For a matrix of dimensions $HW \times T$, 2D FFT is computed. The number of channels ($C$) determines the number of 2D FFTs required, leading to a complexity of $O(C \times HWT \log(HWT))$. Both 2D FFT and 2D iFFT share this complexity, resulting in an overall time complexity of $O(C \times HWT \log(HWT))$ for Fourier Attention. Therefore, Fourier Attention is significantly more efficient than self-attention. In terms of accuracy, space-time Fourier Attention performs comparably to space-time self-attention.

#### 5) CREATING THE END-TO-END MODEL

Finally, the end-to-end model is constructed by combining the modified C3D and Fourier modules with additional Dense layers to adapt it to the specific task at hand. Modified C3D works as a feature extractor which collects the lower-level features from the localized video data and passes it down to the Fourier module. It enhances the feature representation by separating temporal and spatial information and emphasizing spatially coherent patterns. Eventually, two fully connected layers map the enhanced features to the output classes of the task. The end-to-end model is compiled using an appropriate loss function (sparse categorical cross-entropy) and optimizer (Adam).

### *C. LIGHTWEIGHT MODEL FOR EDGE DEVICES*

Usually, video surveillance systems are predominantly implemented in a passive manner, with analysis performed only after a criminal offence has been committed or an atypical behaviour has been reported. This highlights the need for proactive video surveillance systems, which can recognize and prevent potential security threats in real-time. Current approaches often suffer from limitations such as low efficiency and high computational complexity, which limits their real-time applicability. In tandem with the server-side approach, the second part of the methodology introduces a lightweight model designed to operate seamlessly on edge devices. The lightweight model is tailored to strike a balance between computational efficiency and recognition precision, making it well-suited for deployment on resource-constrained UAVs or other edge devices. Figure 7 illustrates the design of the proposed model.

#### 1) ROI EXTRACTION-BASED FRAME SAMPLING

The proposed lightweight model is a combination of an ROI extraction-based frame sampler, feature extractor based on MobileNetV2 and Fourier Module, temporal information retainer based on BiLSTM, and fully connected layers. Each video contains a large number of frames and in plenty of frames, there is no substantial movement which can result in any form of action. So, efficient scene extraction is essential for creating a representative dataset while managing computational resources effectively.

Initially, we applied a Region-of-Interest (ROI) extraction to eliminate background clutter and focus exclusively on human activity in each frame as we did on the server-side model. Using YOLOv8, trained on the COCO dataset [47], we filtered out irrelevant objects, isolating only human detections based on consecutive frame presence and confidence thresholding. Following ROI extraction, we employed FFmpeg's scene change detection to perform efficient frame sampling, optimizing for keyframes that represent significant movement and action without redundant information. FFmpeg [51] is a multimedia framework, which is mainly used for pre-processing videos.

We employed FFmpeg's scene change detection mechanism to optimize the selection of keyframes from UAV video footage. We have empirically determined an optimal scene change threshold of 0.01, ensuring that only frames depicting substantial changes are retained. This threshold value was chosen after extensive experimentation to strike a balance between retaining enough frames for comprehensive action representation and eliminating redundant frames that

do not contribute additional information. This streamlines the dataset creation process and also ensures that the retained frames are highly relevant and informative about our objective.

#### 2) FRAME PREPROCESSING

Let the extracted frames be denoted as $\mathcal{F} = \{f_1, f_2, \ldots, f_N\}$, where $N$ is the total number of frames. Each frame $f_i$ is resized to a fixed height and width of $112 \times 112$. The resized frame is denoted as $f_i^{\text{resized}}$.

$$f_i^{\text{resized}} = \text{Resize}(f_i,\ \text{height} = 112,\ \text{width} = 112) \tag{8}$$

After resizing, mean normalization is applied to each frame to ensure the data has a mean of zero and unit variance. Let the resized frame $f_i^{\text{resized}}$ have pixel values $p_{ij}$, where $j$ indexes the pixels in the frame. Now, $\mu_i$ denotes the mean and $\sigma_i$ denotes the standard deviation of the pixel values in $f_i^{\text{resized}}$. Each pixel value $p_{ij}$ is then normalized:

$$p_{ij}^{\text{normalized}} = \frac{p_{ij} - \mu_i}{\sigma_i} \tag{9}$$

Therefore, the normalized frame $f_i^{\text{normalized}}$ is given by:

$$f_i^{\text{normalized}} = \left\{ \frac{p_{ij} - \mu_i}{\sigma_i} \mid j = 1, 2, \ldots, M \right\} \tag{10}$$

where $M$ is the total number of pixels in the frame (in this case, $M = 112 \times 112$).

#### 3) FEATURE EXTRACTION

The preprocessed frames $\mathcal{F}$ are subsequently passed through MobileNetV2 for feature extraction. MobileNetV2 is a lightweight convolutional neural network that performs efficient feature extraction at lower computational costs, making it well-suited for our application. To balance efficiency and performance, we freeze the initial layers of the MobileNetV2 model and fine-tune only the final ten layers. This selective training allows the model to leverage the pre-trained weights to extract features from earlier layers while adapting the later layers to our specific task.

To prevent overfitting during the training process, we incorporate L2 regularization into the weights of the dense layers. The regularization term $\lambda \sum_i w_i^2$ (where $\lambda$ is the regularization coefficient and $w_i$ are the weights) is added to the loss function, encouraging the model to learn more robust and generalizable features. We have empirically set the regularization coefficient to 0.0001.

Given the sequential nature of the frames, we utilize a TimeDistributed layer to handle the sequence of frames $\mathcal{F}$. The TimeDistributed layer applies the MobileNetV2 model to each frame individually, resulting in a sequence of feature maps. Mathematically, if $\phi$ represents the MobileNetV2 model, then for each frame $f_i$, the feature map $\phi(f_i)$ is generated, and the sequence of frames $\mathcal{F} = \{f_1, f_2, \ldots, f_N\}$ is transformed into a sequence of feature maps $\mathcal{F}' = \{\phi(f_1), \phi(f_2), \ldots, \phi(f_N)\}$. Therefore,

$$\mathcal{F}' = \text{TimeDistributed}(\phi)(\mathcal{F}) \tag{11}$$

where $\phi$ is the MobileNetV2 model.

The low-level features are then passed on to the Fourier module to extract further high-level features. Similar to the serverside model, low-level features pass through two streams at the Fourier block simultaneously. The lightweight model's Fourier module follows a homogenous policy compared to the server side. Fourier substance separation as illustrated in section III-B3, isolates dynamic salient regions from the static non-salient region. Fourier Self-Attention as explained in section III-B4, captures long-range space-time relationships present in feature space.

#### 4) TEMPORAL INFORMATION RETAINER

To capture temporal dependencies within the sequence of feature maps $\mathcal{F}'$, we incorporate a Bidirectional Long Short-Term Memory (BiLSTM) layer. This layer processes the sequence in both forward and backward directions, capturing context from past and future frames. Mathematically,

$$\mathbf{H} = \text{BiLSTM}(\mathcal{F}') \tag{12}$$

Let $\overrightarrow{h}_t$ and $\overleftarrow{h}_t$ be the hidden states of the forward and backward LSTMs at time step $t$. The output sequence of the BiLSTM layer is given by $\mathbf{H} = \{(\overrightarrow{h}_t, \overleftarrow{h}_t) | t = 1, 2, \ldots, N\}$.

#### 5) END-TO-END MODEL

Finally, we add two fully connected (Dense) layers to capture higher-level representations from the hidden states. Dropout regularization is applied to these Dense layers to further prevent overfitting. We employ a categorical cross-entropy loss function to measure the discrepancy between predicted and true labels. Along with it Stochastic Gradient Descent (SGD) optimizer is used to minimize the loss.

The output of these layers is passed through a final dense layer with a softmax activation function, producing the probability distribution over the classes. Mathematically, the model architecture can be represented as:

$$\mathbf{H}' = \text{Dense}(\text{Dropout} \ldots (\mathbf{H})) \tag{13}$$

$$\mathcal{Y} = \text{Dense}(\text{softmax})(\mathbf{H}') \tag{14}$$

Here, $\mathcal{Y}$ denotes the final output probabilities for the classes.

By leveraging MobileNetV2 and Fourier module for efficient feature extraction, BiLSTM for temporal dependency capture, and Dense layers with dropout for higher-level representation, our model effectively classifies human actions in UAV video sequences.

## IV. EXPERIMENTAL RESULTS & ANALYSIS

### A. SERVER-SIDE MODEL RESULTS

The server-side model performs a series of tasks. Initially, through a Region-of-Interest (ROI) generation step, specific regions within these videos containing the persons of interest are located and isolated. As a result of this process, the memory usage for each video is remarkably reduced, by as much as 80%. Thereafter, features were retrieved from the dataset using modified C3D. The mid-level features were then

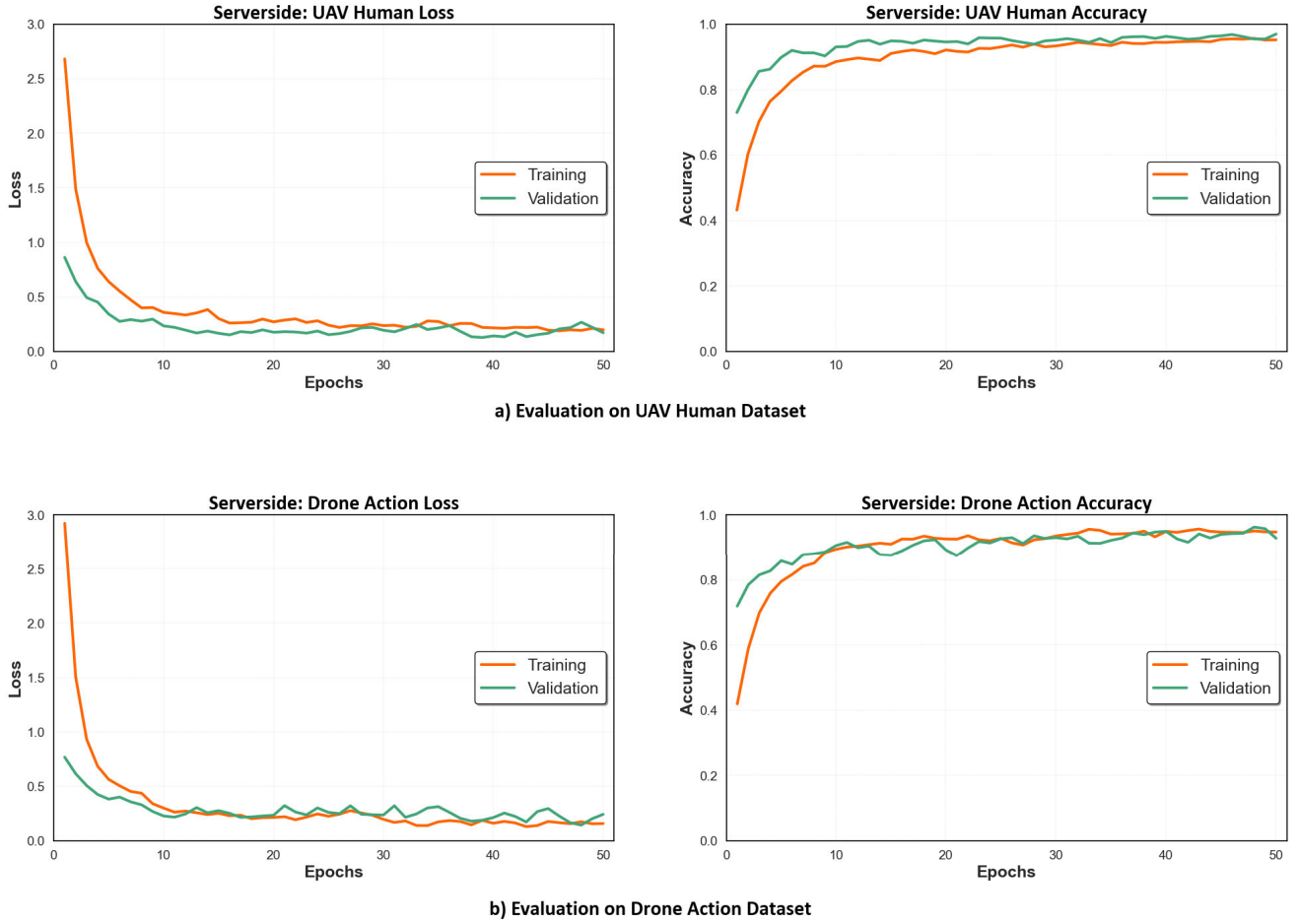



**FIGURE 8.** Loss and accuracy curve for the proposed U-ActionNet model on the UAV Human [8] and the drone action [10] dataset.

transferred to the network's Fourier block to facilitate Fourier substance separation and Fourier self-attention. Finally, two dense neural network layers were applied to the combined high-level feature maps generated from the Fourier block to classify the actions being performed.

For the server-side model, we employed the ReLU activation function for all layers except the final layer, which utilized a softmax activation function. This configuration allows the model to handle non-linearities effectively and ensures proper probability distribution over the output classes. The loss function used was sparse categorical cross entropy, suitable for multi-class classification problems with integer labels. The model was optimized using the Adam optimizer with a learning rate of 0.001, providing efficient and adaptive gradient descent. The training was performed over 50 epochs with a batch size of 16, balancing computational efficiency while ensuring adequate model convergence.

The training and validation loss curves for the UAV Human and Drone Action datasets, as shown in Figure 8, exhibit a clear convergence trend. In both datasets, the loss decreases rapidly within the initial epochs, indicating that the model quickly learns the underlying patterns. For the UAV Human dataset, the training loss stabilizes at approximately 0.1924, while the Drone Action dataset achieves a slightly lower training loss of 0.1592. The validation loss for both datasets follows a similar pattern, converging close to the training loss, suggesting minimal overfitting.

Shifting the focus to the accuracy curves, illustrated in Figure 8, the model exhibits strong performance across both datasets. The UAV Human dataset achieves a training accuracy of 95.05%, while the Drone Action dataset reaches 94.94%. Both training and validation accuracy curves show a rapid increase in the initial epochs, plateauing as they approach their respective maximum values. The validation accuracy remains closely aligned with the training accuracy, further indicating the model's generalization capability.

Building on the accuracy curve results, the classification reports for both datasets provide a more granular view of the model's performance across individual action classes. The server-side model's classification performance for the UAV Human and Drone Action datasets, detailed in Table 1, shows exceptional precision, recall, and F1 scores across all action

**TABLE 1.** Classification report for server-side model (U-ActionNet) for UAV human and drone action datasets.

| Experimental Datasets | Actions | Precision | Recall | Accuracy | F1 Score |
|---|---|---|---|---|---|
| UAV Human | Punching someone | 0.92 | 0.90 | 0.98 | 0.91 |
| | Kicking someone | 0.90 | 0.94 | 0.98 | 0.92 |
| | Pushing someone | 0.93 | 0.88 | 0.98 | 0.91 |
| | Slap someone on the back | 0.99 | 0.99 | 0.99 | 0.99 |
| | Threat someone with a knife | 0.94 | 0.92 | 0.99 | 0.93 |
| | Hold someone hostage | 0.97 | 0.97 | 0.99 | 0.97 |
| | Threat someone with a gun | 0.94 | 0.98 | 0.99 | 0.96 |
| | Drag someone | 0.98 | 0.97 | 0.99 | 0.97 |
| | Call for help | 0.99 | 0.99 | 0.99 | 0.99 |
| | Stab someone with a knife | 0.95 | 0.97 | 0.99 | 0.96 |
| Drone Action | Clapping | 0.92 | 1.00 | 0.99 | 0.96 |
| | Hitting Bottle | 0.78 | 0.97 | 0.97 | 0.87 |
| | Hitting Stick | 1.00 | 1.00 | 1.00 | 1.00 |
| | Jogging | 1.00 | 0.83 | 0.99 | 0.90 |
| | Kicking | 0.98 | 0.98 | 0.99 | 0.98 |
| | Running | 0.78 | 1.00 | 0.99 | 0.88 |
| | Stabbing | 1.00 | 0.88 | 0.98 | 0.94 |
| | Walking | 1.00 | 1.00 | 1.00 | 1.00 |
| | Waving Hands | 1.00 | 0.95 | 0.99 | 0.98 |
| | Punching | 1.00 | 0.89 | 0.99 | 0.94 |

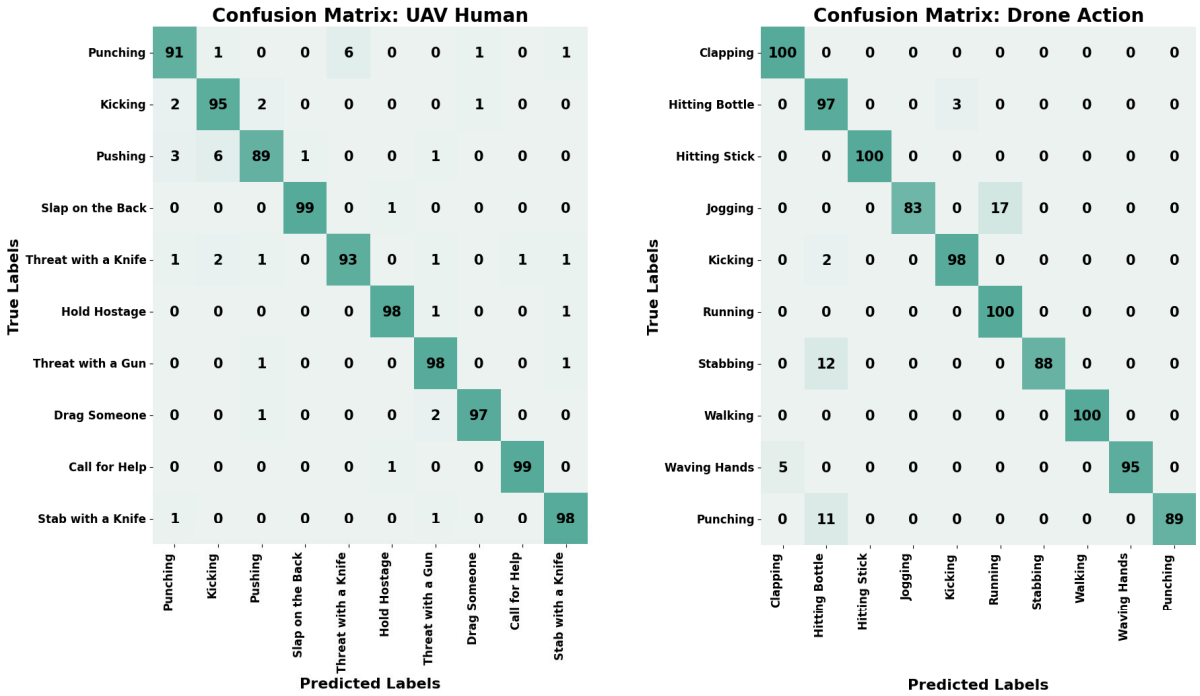



**FIGURE 9.** Confusion matrix of the serverside model (U-ActionNet) on the UAV human dataset (on the left) [8] and on the drone action dataset (on the right ) [10].

classes. Notably, ''Slap someone on the back'' and ''Call for help'' both achieved almost perfect scores of 0.99, while ''Hold someone hostage'' and ''Drag someone'' attained F1-scores of 0.97. Other actions, such as ''Threat someone with a gun'' and ''Stab someone with a knife'', also demonstrated high performance with F1-scores of 0.96, indicating the model's robust capability in accurately detecting diverse human activities.

Similarly, the report presented in Table 1 for the Drone Action dataset highlights the model's outstanding accuracy. Actions like ''Hitting Stick'', ''Walking'', and ''Waving Hands'' achieved perfect F1 scores of 1.00. ''Kicking''

**TABLE 2. Evaluation between different models and proposed U-ActionNet model.**

| Method | Dataset | Loss | Top-1 Accuracy | Top-3 Accuracy |
|---|---|---|---|---|
| C3D + FC [49] | Drone Action | 0.3774 | 78.52% | 86.71% |
| FFTUAVNet [52] | | 0.2554 | 88.77% | 98.19% |
| **U-ActionNet** | | **0.1592** | **94.94%** | **98.87%** |
| X3D [3] | UAV Human | 1.977 | 31.33% | 45.60% |
| C3D + FC [49] | | 0.3497 | 88.70% | 94.44% |
| FFTUAVNet [52] | | 0.992 | 64.86% | 83.37% |
| **U-ActionNet** | | **0.1924** | **95.05%** | **98.64%** |

also performed exceptionally well with an F1 score of 0.98, and ''Punching'' and ''Stabbing'' attained strong F1 scores of 0.94. While ''Hitting Bottle'' and ''Running'' showed slightly lower precision, the model maintained high overall performance, with all F1-scores exceeding 0.91. This consistency underscores the model's effectiveness in accurately classifying a wide range of actions in both datasets.

Following the detailed classification report, the confusion matrices for the UAV Human and Drone Action datasets offer a comprehensive visual representation of the model's performance across different action classes, as illustrated in Figure 9. The confusion matrix for the UAV Human dataset reveals high accuracy in most classes, with actions such as ''Slap on the Back'' and ''Call for Help'' showing minimal misclassifications, achieving near-perfect diagonal values, respectively. However, there are minor confusions, such as ''Punching'' and ''Pushing'' being occasionally misclassified. Similarly, the confusion matrix for the Drone Action dataset also demonstrates strong performance, with actions like ''Hitting Stick,''''Clapping,'' and ''Waving Hands'' showing perfect or near-perfect diagonal values. Nevertheless, there are notable confusions such as ''Jogging'' being misclassified as ''Running'', and ''Punching'' being misclassified as ''Hitting Bottle''. These matrices accentuate the model's strengths in correctly identifying the majority of actions while also pointing out specific areas where improvements could be made to reduce misclassifications.

Transitioning from the analysis of the confusion matrices, we now evaluate the performance of various models, including our (proposed) **U-ActionNet** method as presented in Table 2. Table 2 comprehensively compares the loss, Top-1 accuracy, and Top-3 accuracy of several models, i.e. C3D [49], FFTUAVNet [52], X3D [3], and our U-ActionNet method on the Drone Action and UAV Human datasets. For the Drone Action dataset, our proposed method outperforms the other models, achieving the lowest loss of 0.1592, a Top-1 accuracy of 94.94%, and a Top-3 accuracy of 98.87%. In comparison, FFTUAVNet shows a loss of 0.2554 with a Top-1 accuracy of 88.77% and a Top-3 accuracy of 98.19%, while C3D + Fully Connected Layers (FC) records a higher loss of 0.3774, with Top-1 and Top-3 accuracies of 78.52% and 86.71%, respectively.

For the UAV Human dataset, the superiority of our U-ActionNet method is again evident, with a notable loss reduction to 0.1924, a Top-1 accuracy of 95.05%, and a Top-3 accuracy of 98.64%. The closest competitor, C3D + FC, achieves a loss of 0.3497, with Top-1 and Top-3 accuracies of 88.70% and 94.44%, respectively. FFTUAVNet, while performing adequately with a loss of 0.992 and accuracies of 64.86% (Top-1) and 83.37% (Top-3), and X3D, with much poorer results of a 1.977 loss and 31.33% Top-1 accuracy, lag significantly behind. These results underscore the enhanced effectiveness and precision of our U-ActionNet model in detecting and classifying not only the primary human actions but also the top alternatives within both datasets. The evaluation metrics clearly indicate that our proposed Server-side method, U-ActionNet, significantly outperforms existing models in terms of lower loss and higher accuracy rates for both the Drone Action and UAV Human datasets.

This robust performance highlights the potential of our approach for practical applications in human action detection and classification, as substantiated by the comparative data presented. The comparative analysis underscores the robustness and generalizability of our approach across different scenarios, outperforming both the traditional C3D model and the FFTUAVNet [52] in several key metrics.

### B. LIGHTWEIGHT MODEL RESULTS

To generate a lightweight model it is necessary to keep in mind to ignore the static unnecessary frames and use only the essential regions of interest. Using a Region-of-Interest (ROI) module, relevant areas containing individuals were located and isolated. Keyframes were selected with an FFmpeg-based frame sampler, applying a scene change detection filter where the scene change factor exceeded 0.01, reducing the frame count by up to 30%. These frames were then processed by MobileNetV2 for feature extraction, followed by a Fourier block for substance separation and self-attention. Extracted features were stored with memory mapping to minimize computational costs and fed to a BiLSTM module to capture temporal context. Finally, two fully connected layers classified the actions performed.

For the lightweight model, we utilized the ReLU activation function for all layers except the final one, which used softmax to generate a probability distribution over the classes. The model was trained using the categorical cross-entropy loss function, which is suitable for one-hot encoded labels. The optimizer selected was SGD (Stochastic Gradient Descent) with a learning rate of 0.01, ensuring a gradual and controlled optimization process. The training was performed over 50 epochs with a batch size of 16, balancing computational efficiency and effective model learning.

The loss and accuracy curves for the lightweight model on the UAV Human and Drone Action datasets are presented in Figure 10. For the UAV Human dataset [8], the training loss begins at approximately 2.5 and progressively decreases, while the validation loss follows a similar downward trend. Similarly, for the Drone Action dataset, the training loss

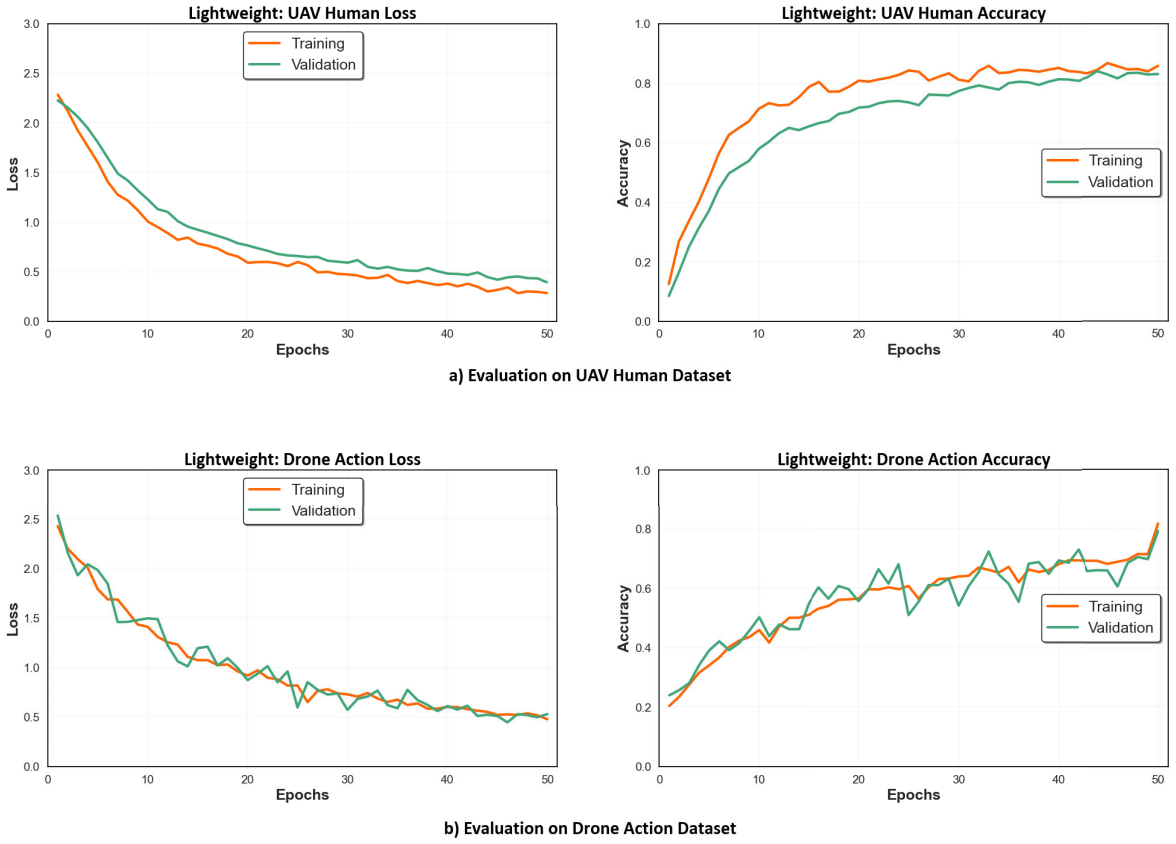



**FIGURE 10.** Loss and accuracy curve for the proposed U-ActionNet light model on the UAV human [8] and the drone action [10] dataset.

**TABLE 3.** Classification report for the lightweight model (U-ActionNet light) for UAV human and drone action datasets.

| Experimental Datasets | Actions | Precision | Recall | Accuracy | F1 Score |
|---|---|---|---|---|---|
| UAV Human | Punching someone | 0.75 | 0.88 | 0.96 | 0.81 |
| | Kicking someone | 0.83 | 0.90 | 0.98 | 0.86 |
| | Pushing someone | 0.83 | 0.63 | 0.96 | 0.72 |
| | Slap someone on the back | 0.83 | 0.86 | 0.98 | 0.84 |
| | Threat someone with a knife | 0.71 | 0.93 | 0.96 | 0.81 |
| | Hold someone hostage | 0.88 | 0.68 | 0.97 | 0.77 |
| | Threat someone with a gun | 0.95 | 0.87 | 0.97 | 0.91 |
| | Drag someone | 0.78 | 0.90 | 0.96 | 0.84 |
| | Call for help | 0.94 | 0.82 | 0.97 | 0.87 |
| | Stab someone with a knife | 0.95 | 0.90 | 0.98 | 0.93 |
| Drone Action | Clapping | 0.75 | 0.90 | 0.97 | 0.82 |
| | Hitting Bottle | 0.74 | 0.86 | 0.94 | 0.79 |
| | Hitting Stick | 0.96 | 0.85 | 0.98 | 0.90 |
| | Jogging | 1.00 | 0.50 | 0.95 | 0.67 |
| | Kicking | 0.92 | 0.96 | 0.99 | 0.94 |
| | Running | 0.67 | 0.29 | 0.97 | 0.40 |
| | Stabbing | 0.80 | 0.60 | 0.95 | 0.69 |
| | Walking | 0.57 | 1.00 | 0.93 | 0.72 |
| | Waving Hands | 0.92 | 0.80 | 0.97 | 0.86 |
| | Punching | 0.83 | 0.89 | 0.97 | 0.86 |

starts around 2.5 and consistently reduces to a final value of 0.5081. The validation loss, although showing more variability, mirrors the general decrease observed in the training loss.

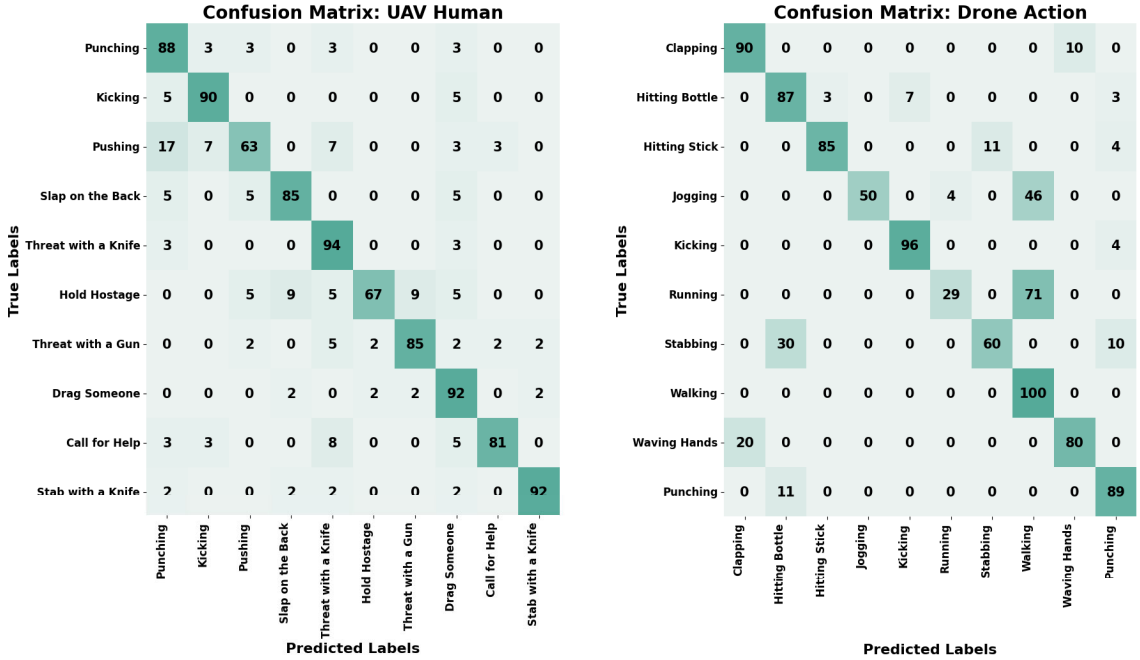



**FIGURE 11.** Confusion matrix of the lightweight model (U-ActionNet light) on the UAV human dataset (on the left) [8] and on the drone action dataset (on the right ) [10].

Following the analysis of the loss curves, the accuracy curves for both the UAV Human and Drone Action datasets provide further insights into the performance of the lightweight model, as depicted in Figure 10. The training and validation accuracy curves exhibit a steady upward trend. For the UAV Human dataset, the training accuracy starts below 20% and increases consistently, culminating in a final accuracy of 84.74%. The validation accuracy also shows a similar trend, reflecting the model's ability to generalize well to unseen data. In the case of the Drone Action dataset, the training accuracy follows a comparable pattern, beginning near 20% and reaching a final accuracy of 80.43%. Despite the greater variability observed in the validation accuracy curve, it too converges towards the training accuracy, underscoring the model's robustness.

Expanding upon the findings of the accuracy curve, classification reports for each dataset offer a more detailed perspective of the model's effectiveness for distinct action classes. As shown in Table 3, the lightweight model demonstrates strong performance on the UAV Human dataset with high precision, recall, and F1-scores in most categories. Notably, the class ''Stab someone with a knife'' achieved the highest precision of 0.95 and an F1-score of 0.93, indicating the model's robustness in recognizing this action. Similarly, ''Threat someone with a gun'' has a high F1-score of 0.91, reflecting its effective detection. However, there are areas, such as in ''Pushing someone'', which has the lowest recall of 0.63, which suggests that the model has missed some instances of this class.

Transitioning to the Drone Action dataset, as summarized in Table 3, the lightweight model showcases varying performance levels across different actions. The model excels in identifying ''Kicking'' and ''Hitting Stick'', with both classes achieving near-perfect precision and F1 scores of 0.94 and 0.90, respectively. Conversely, the model struggles significantly with identifying actions such as ''Jogging'' and ''Running'' as both have quite similar forms of action.

Building on the detailed classification report, the confusion matrices as illustrated in Figure 11, offer a comprehensive visualization of the lightweight model's performance. For the UAV Human dataset, the matrix reveals that the model excels in identifying actions such as ''Threat with a Knife'' and ''Stab with a Knife''. However, some confusion is evident in the ''Pushing'' class, which is occasionally misclassified as ''Punching''. On the other hand, the Drone Action dataset's confusion matrix highlights the model's proficiency in detecting ''Hitting Bottle'', ''Hitting Stick'', and ''Kicking'' marked by high correct predictions and minimal errors. Nevertheless, significant misclassifications occur in actions like ''Running'' which is often incorrectly identified as ''Walking''. These insights from the confusion matrices signify our proposed model's strengths in certain classifications.

Following the insights provided by the confusion matrices, Table 4 compares the baseline model with the proposed U-ActionNet Light model for both datasets. The baseline model is defined as a non-frame-sampled architecture that utilizes MobileNet as the feature extractor and BiLSTM for temporal information retention. Our proposed model demonstrates a notable reduction in loss, with a decrease from 0.6037 to 0.5081 for the Drone Action dataset and from 0.5218 to 0.4988 for the UAV Human dataset. In terms

**TABLE 4.** Comparison between baseline and U-ActionNet light. The baseline model is defined as a non-frame-sampled architecture that utilizes MobileNet as the feature extractor and BiLSTM for temporal information retention.

| Method | Dataset | Loss | Top-1 Accuracy | Top-3 Accuracy |
|---|---|---|---|---|
| Baseline | Drone Action | 0.6037 | 74.19% | 86.91% |
| **U-ActionNet Light** | | **0.5081** | **80.43%** | **90.93%** |
| Baseline | UAV Human | 0.5218 | 82.38% | 92.05% |
| **U-ActionNet Light** | | **0.4988** | **84.74%** | **93.41%** |

**TABLE 5.** Percentage improvement over baseline.

| Comparison | Dataset | Top-1 Accuracy Improve (%) | Top-3 Accuracy Improve (%) |
|---|---|---|---|
| **Baseline -vs- Proposed** | Drone Action | 8.41% | 4.63% |
| **Baseline -vs- Proposed** | UAV Human | 2.86% | 1.48% |

**TABLE 6.** Model parameters.

| Method | Trainable Parameters | Non-trainable Parameters | Total Parameters |
|---|---|---|---|
| Lightweight Model | 2,003,018 | 5,525,504 | 7,528,522 |
| Serverside Model | 10,102,794 | 53,214,464 | 63,317,258 |

of accuracy, the U-ActionNet Light model achieves a Top-1 accuracy of 80.43% for Drone Action, surpassing the baseline's 74.19%, and a Top-1 accuracy of 84.74% for UAV Human, compared to the baseline's 82.38%. Additionally, the Top-3 accuracy metrics highlight further improvements, with the proposed model reaching 90.93% for Drone Action and 93.41% for UAV Human, outperforming the baseline figures of 86.91% and 92.05%, respectively.

Expanding on the results of the prior performance assessment, Table 5 presents the percentage improvement in Top-1 and Top-3 accuracies achieved by the proposed U-ActionNet Light model over the baseline. For the Drone Action dataset, the lightweight model shows a significant Top-1 accuracy improvement of 8.41%, alongside a Top-3 accuracy enhancement of 4.63%. Similarly, for the UAV Human dataset, the model demonstrates a 2.78% improvement in Top-1 accuracy and a 1.48% increase in Top-3 accuracy compared to the baseline. Overall, these results confirm the superiority of our U-ActionNet Light model, which provides a compelling solution to real-time action recognition tasks, making it highly suitable for edge applications.

### C. COMPARATIVE ANALYSIS

To optimize human action recognition in UAV videos, we designed two models to meet distinct computational needs. The serverside model exhibits remarkable computational depth, enabling intricate analysis and interpretation of human actions. On the other hand, our frame-sampled lightweight model purposefully architected for deployment on edge devices, enhances computational efficiency and ensures rapid inference time.

**TABLE 7.** Inference times for different models between datasets.

| Method | Dataset | Inference Time (seconds) |
|---|---|---|
| Serverside | UAV Human | 0.291 |
| Lightweight | | 0.164 |
| Serverside | Drone Action | 0.395 |
| Lightweight | | 0.216 |

As shown in Table 6, the lightweight model uses only one-ninth of the parameters of the server-side model. The heavyweight model, designed for server-side processing, boasts a substantial parameter count totalling 63,317,258, comprising both trainable and non-trainable parameters. With 10,102,794 trainable parameters, this model exhibits a rich capacity for capturing complexities within UAV video data. In stark contrast, the lightweight model, strategically designed for edge deployment, maintains a significantly leaner architecture, featuring a total parameter count of 7,528,522. Characterized by 2,003,018 trainable parameters, the lightweight model strikes a balance between computational efficiency and model complexity.

Along with this, Table 7 reveals a remarkable inference time differential between the server-side and lightweight model across different datasets. For the UAV Human dataset, the server-side model demonstrates an average inference time of 0.291 seconds, while the lightweight model significantly reduces this duration to 0.164 seconds. Similarly, for the Drone Action dataset, the server-side model incurs an average inference time of 0.395 seconds, in contrast to the lightweight model's more efficient 0.216 seconds.

The substantial reduction in inference time without a significant compromise on performance highlights the lightweight model's practicality for deployment in resource-constrained environments where latency and computational resources are of critical consideration. Consequently, the choice between models should be guided by the specific operational demands, where the lightweight model offers a pragmatic solution for scenarios necessitating rapid inference and lower computational overheads.

## V. CONCLUSION

In this paper, we have proposed novel solutions to recognize human actions from UAV videos for both server-side and edge devices. For server-side devices, we leveraged the structural and mathematical properties of YOLO and FFT to systematically isolate humans from the background and

capture prolonged space-time dependencies. This method has a major limitation as it is computationally too expensive to deploy on edge devices. To mitigate this issue, our U-ActionNet Light model utilizes an adaptive ROI extraction-based frame sampling strategy to sample only dynamic frames which comprehensively collects significant frames. Subsequently, to recognize and classify human actions, MobileNetv2, Fourier Module, and BiLSTM are utilized which greatly reduces trainable parameters and ensures its deployability on edge devices. Both of our methods demonstrate significant top-1 and top-3 accuracy improvements on SOTA datasets in comparison to prior approaches. Nevertheless, we report a limitation of our methods. Here, we assumed that each input video only shows at max two human actors acting. We would like to investigate the plausibility of expanding our method into multi-actor-based multi-action videos. Nonetheless, for our U-ActionNet Light model, exploring the application of improved video sampling techniques could be worthwhile.

## REFERENCES

[1] J. Carreira and A. Zisserman, "Quo Vadis, action recognition? A new model and the kinetics dataset," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4724–4733.

[2] C. Feichtenhofer, H. Fan, J. Malik, and K. He, "SlowFast networks for video recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6201–6210.

[3] C. Feichtenhofer, "X3D: Expanding architectures for efficient video recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 200–210.

[4] D. Kondratyuk, L. Yuan, Y. Li, L. Zhang, M. Tan, M. Brown, and B. Gong, "MoViNets: Mobile video networks for efficient video recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 16015–16025.

[5] M. Monfort, A. Andonian, B. Zhou, K. Ramakrishnan, S. A. Bargal, T. Yan, L. Brown, Q. Fan, D. Gutfreund, C. Vondrick, and A. Oliva, "Moments in time dataset: One million videos for event understanding," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 502–508, Feb. 2020.

[6] K. Soomro, A. R. Zamir, and M. Shah, "UCF101: A dataset of 101 human actions classes from videos in the wild," 2012, *arXiv:1212.0402*.

[7] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre, "HMDB: A large video database for human motion recognition," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 2556–2563.

[8] T. Li, J. Liu, W. Zhang, Y. Ni, W. Wang, and Z. Li, "UAV-human: A large benchmark for human behavior understanding with unmanned aerial vehicles," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 16261–16270.

[9] U. Demir, Y. S. Rawat, and M. Shah, "TinyVIRAT: low-resolution video action recognition," in *Proc. 25th Int. Conf. Pattern Recognit. (ICPR)*, Jan. 2021, pp. 7387–7394.

[10] A. G. Perera, Y. W. Law, and J. Chahl, "Drone-action: An outdoor recorded drone video dataset for action recognition," *Drones*, vol. 3, no. 4, p. 82, Nov. 2019. [Online]. Available: https://www.mdpi.com/2504-446X/3/4/82

[11] K. Li, Z. Wu, K.-C. Peng, J. Ernst, and Y. Fu, "Tell me where to look: Guided attention inference network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 9215–9223.

[12] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 28, Dec. 2015, pp. 91–99.

[13] M. Wang and W. Deng, "Deep visual domain adaptation: A survey," *Neurocomputing*, vol. 312, pp. 135–153, Oct. 2018.

[14] Y. Chen, W. Li, C. Sakaridis, D. Dai, and L. Van Gool, "Domain adaptive faster R-CNN for object detection in the wild," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3339–3348.

[15] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, Jun. 2017, pp. 5998–6008.

[16] H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena, "Self-attention generative adversarial networks," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 7354–7363.

[17] G. Bertasius, H. Wang, and L. Torresani, "Is space-time attention all you need for video understanding?" in *Proc. ICML*, Jan. 2021, p. 4.

[18] I. Hussain, R. Jany, R. Boyer, A. Azad, S. A. Alyami, S. J. Park, M. M. Hasan, and M. A. Hossain, "An explainable EEG-based human activity recognition model using machine-learning approach and LIME," *Sensors*, vol. 23, no. 17, p. 7452, Aug. 2023.

[19] K. Hara, H. Kataoka, and Y. Satoh, "Can spatiotemporal 3D CNNs retrace the history of 2D CNNs and ImageNet?" in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6546–6555.

[20] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Proc. Adv. Neural Inf. Process. Syst.*, Jan. 2014, pp. 1–9.

[21] C. Feichtenhofer, A. Pinz, and A. Zisserman, "Convolutional two-stream network fusion for video action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1933–1941.

[22] R. Girdhar, J. João Carreira, C. Doersch, and A. Zisserman, "Video action transformer network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 244–253.

[23] 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.

[24] S. N. Gowda, M. Rohrbach, and L. Sevilla-Lara, "SMART frame selection for action recognition," in *Proc. AAAI Conf. Artif. Intell.*, May 2021, vol. 35, no. 2, pp. 1451–1459.

[25] B. Korbar, D. Tran, and L. Torresani, "SCSampler: Sampling salient clips from video for efficient action recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6231–6241.

[26] M. Ding, N. Li, Z. Song, R. Zhang, X. Zhang, and H. Zhou, "A lightweight action recognition method for Unmanned-Aerial-Vehicle video," in *Proc. IEEE 3rd Int. Conf. Electron. Commun. Eng. (ICECE)*, Dec. 2020, pp. 181–185.

[27] D. Du, Y. Qi, H. Yu, Y. Yang, K. Duan, G. Li, W. Zhang, Q. Huang, and Q. Tian, "The unmanned aerial vehicle benchmark: Object detection and tracking," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Jan. 2018, pp. 375–391.

[28] P. Mittal, R. Singh, and A. Sharma, "Deep learning-based object detection in low-altitude UAV datasets: A survey," *Image Vis. Comput.*, vol. 104, Dec. 2020, Art. no. 104046.

[29] T. Ahmad, M. Cavazza, Y. Matsuo, and H. Prendinger, "Detecting human actions in drone images using YoloV5 and stochastic gradient boosting," *Sensors*, vol. 22, no. 18, p. 7020, Sep. 2022.

[30] *YOLOv5*. Accessed: Oct. 31, 2022. [Online]. Available: https://github.com/ultralytics/yolov5

[31] M. Barekatain, M. Martí, H.-F. Shih, S. Murray, K. Nakayama, Y. Matsuo, and H. Prendinger, "Okutama-action: An aerial view video dataset for concurrent human action detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2017, pp. 2153–2160.

[32] R. Xian, X. Wang, and D. Manocha, "MITFAS: Mutual information based temporal feature alignment and sampling for aerial video action recognition," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2024, pp. 6611–6620.

[33] Q. Fan, C.-F. Chen, H. Kuehne, M. Pistoia, and D. Cox, "More is less: Learning efficient video representations by big-little network and depthwise temporal aggregation," in *Proc. Adv. Neural Inf. Process. Syst.*, Jan. 2019, pp. 1–10.

[34] S. Bhardwaj, M. Srinivasan, and M. M. Khapra, "Efficient video classification using fewer frames," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 354–363.

[35] X. Li, Y. Wang, Z. Zhou, and Y. Qiao, "SmallBigNet: Integrating core and contextual views for video classification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 1089–1098.

[36] 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.

[37] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*.

[38] J. Lin, C. Gan, and S. Han, "TSM: Temporal shift module for efficient video understanding," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 7082–7092.

[39] X. Wang, R. Xian, T. Guan, C. M. de Melo, S. M. Nogar, A. Bera, and D. Manocha, "AZTR: Aerial video action recognition with auto zoom and temporal reasoning," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2023, pp. 1312–1318.

[40] R. Xian, X. Wang, D. Kothandaraman, and D. Manocha, "PMI sampler: Patch similarity guided frame selection for aerial action recognition," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2024, pp. 6967–6976.

[41] H. Buijs, A. Pomerleau, M. Fournier, and W. Tam, "Implementation of a fast Fourier transform (FFT) for image processing applications," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-22, no. 6, pp. 420–424, Dec. 1974.

[42] Z. Zhang, J. Zhao, D. Zhang, C. Qu, Y. Ke, and B. Cai, "Contour based forest fire detection using FFT and wavelet," in *Proc. Int. Conf. Comput. Sci. Softw. Eng.*, Dec. 2008, pp. 760–763.

[43] L. Chi, B. Jiang, and Y. Mu, "Fast Fourier convolution," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, Jan. 2020, pp. 4479–4488.

[44] D. Kothandaraman, T. Guan, X. Wang, S. Hu, M. C. Lin, and D. Manocha, "FAR: Fourier aerial video recognition," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, Jan. 2022, pp. 657–676.

[45] D. Kothandaraman, M. Lin, and D. Manocha, "DifFAR: Differentiable frequency-based disentanglement for aerial video action recognition," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2023, pp. 8254–8261.

[46] G. Jocher, A. Chaurasia, and J. Qiu. (Jan. 2023). *Ultralytics YOLO*. [Online]. Available: https://github.com/ultralytics/ultralytics

[47] T.-Y. Lin, "Microsoft COCO: Common objects in context," in *Proc. 13th Eur. Conf. Comput. Vis.*, Zurich, Switzerland. Cham, Switzerland: Springer, Jan. 2014, pp. 740–755.

[48] M. M. Hasan, K. Ahn, and O. Chae, "Measuring blockiness of videos using edge enhancement filtering," in *Signal Processing, Image Processing and Pattern Recognition*. Berlin, Germany: Springer, 2011, pp. 10–19.

[49] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3D convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 4489–4497.

[50] M. Frigo and S. G. Johnson, "FFTW: An adaptive software architecture for the FFT," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, vol. 3, May 1998, pp. 1381–1384.

[51] S. Tomar, "Converting video formats with FFmpeg," *Linux J.*, vol. 2006, no. 146, p. 10, Jun. 2006.

[52] A. M. Chowdhury, A. Imran, and M. M. Hasan, "FFT-UAVNet: FFT based human action recognition for drone surveillance system," in *Proc. 5th Int. Conf. Sustain. Technol. Ind. (STI)*, Dec. 2023, pp. 1–6.



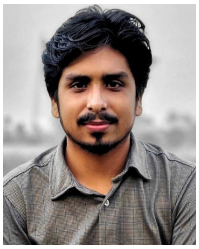
**ABDUL MONAF CHOWDHURY** received the B.Sc. degree in robotics and mechatronics engineering from the University of Dhaka, Bangladesh. He is currently a Research Assistant with MAIM Laboratory, University of Dhaka, working on a Wellcome Leap (In Utero) funded project collaborating with University College Dublin to develop a wearable device to monitor fetal movements to prevent stillbirth. He has previous experience as an engineer in the computer vision and machine learning domain. His research interests include multi-modal learning, continual learning, vision language models, human action recognition, robot vision and other computer vision, and machine learning-based applications.



**AHSAN IMRAN** received the B.Sc. and M.Sc. degrees in robotics and mechatronics engineering from the University of Dhaka, Bangladesh. He is currently a Research Assistant with MAIM Laboratory, University of Dhaka, working on a Wellcome Leap (In Utero) funded project collaborating with University College Dublin to develop a wearable device to monitor fetal movements to prevent stillbirth. His research interests include AI for health with wearables, multi-modal signal processing, human action recognition, and vision language model with RL.



**MD MEHEDI HASAN** received the Ph.D. degree in computer science from the University of New South Wales, Australia, in 2017. He joined as a Postdoctoral with UNSW. Afterward, he worked with the Phenomics and Bioinformatics Research Centre, University of South Australia, as a Postdoctoral Researcher to conduct applied research on aerial and land-based imaging platforms with various geospatial sensors. Later on, he leads as a Machine Learning Engineer (Research and Development) at RME Australia to provide AI-driven industry 4.0 automation systems for Mill Relining Machines. He is currently a Faculty Member with the Department of Robotics and Mechatronics Engineering, University of Dhaka, Bangladesh. Earlier, he was an Associate Professor and the Research Director of SURC, Stamford University, Bangladesh. He is a reviewer of several distinguished journals, including *Briefings in Bioinformatics*, *Plant Methods*, *Frontiers in Plant Science*, *Functional Plant Biology*, and IEEE Access. His research interests include plant phenomics, robotics visual servoing, the IoT, 3D video coding, machine vision, and deep learning.



**RIAD AHMED** received the B.Sc. degree from the Department of Robotics and Mechatronics Engineering, University of Dhaka, Bangladesh. He is currently a Lecturer with the Computer Science and Engineering Department, BRAC University, Bangladesh. Before that, he was a Machine Learning Engineer with Headless Technologies Ltd. and a Computer Vision Engineer with Pranon Global Ltd. He was a Research Assistant with the University of Dhaka. His research interests include computer vision, machine learning, the Internet of Things, industrial automation, activity recognition, video analytics, and artificial intelligence in healthcare.

**AKM AZAD** is a Cancer Data Scientist and an AI Scientist in research. He is currently appointed as a Faculty Member with Imam Mohammad Ibn Saud Islamic University, Faculty of Science. He has an excellent research career with strong and highly focused track record on the development of advanced methodologies and computational resources for multi-omics (e.g., Transcriptomics, Proteomics, and Genomics) data analysis and their relevant drug positioning. Earlier, he worked as a Cancer Data Scientist and Bioinformatician in multiple research institutes in Australia and South Korea, i.e., Children's Medical Research Institute, Australia, Kolling Institute of Medical Research, Australia, University of New South Wales, Australia, and Gwangju Institute of Science and Technology, South Korea. During his research career, he has gained experiences in multi-omics data for identifying network-level biomarkers and analyzing their drug-targets in various cancers, e.g., Glioblastoma, Breast, and Pancreas. His research interests include AI-based biomedicine, data science, Bayesian methods, Bayesian networks, and drug repositioning.



**SALEM A. ALYAMI** (Member, IEEE) received the Ph.D. degree in bio-statistics from Monash University, Australia, in 2017. He is currently an Associate Professor with the School of Mathematics and Statistics, Faculty of Science, Imam Mohammad Ibn Saud Islamic University (IMSIU), Riyadh, Saudi Arabia. He has been contributing and leading several grants in statistics, and serving as the Dean of Scientific Research with IMSIU, since 2020. His research interests include statistical inference, Bayesian statistics, sampling algorithms, statistical simulation, and the applications of statistics in biology and medicine.