International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Overcoming Occlusion Challenges in Human Motion Detection through Advanced Deep Learning Techniques

Jeba Nega Cheltha*[1], Chirag Sharma[2], Pankaj Dadheech[3], Dinesh Goyal[4]

Submitted: 24/12/2023 **Revised**: 30/01/2024 **Accepted**: 06/02/2024

Abstract: Researchers engaged in Human Motion Detection (HMD) grapple with a primary challenge related to occlusion, where individuals or their body parts become obscured within image or video frames. Occlusion manifests in two distinct forms: Self-occlusion, occurring when one part of the human body hides another, and External occlusion, arising from external objects obstructing humans. This proposed work specifically focuses on self-occlusion and partial-occlusion. To discern human motion from visual data, three fundamental methods are deployed. The initial method, motion segmentation, entails identifying the moving object in a video. The second method, Object Classification, determines whether the moving object is human. The final method, the Tracking algorithm, is employed for identifying human gestures. Occlusion persists as a central concern in HMD. In our proposed methodology, we employ a Mask Region-based Convolutional Neural Network (Mask R-CNN) for motion segmentation to address the occlusion challenge. Object classification utilizes a Recurrent Neural Network (RNN), and for tracking human motion, even during self-occlusion, Multiple Hypothesis Tracking (MHT) is applied. This study presented an innovative hybrid algorithm, the Whale Optimization Algorithm and Red Deer Algorithm (WOA-RDA), demonstrating superior convergence speed coupled with high accuracy. Our HMD approach incorporates an RNN trained with 2D representations of 3D skeletal motion. Diverse datasets, encompassing scenarios with and without occlusion, are integrated into our proposed work. The experimental findings underscore the effectiveness of our approach in accurately identifying human motion under varied conditions, including both with and without occlusion scenarios.

Keywords: *Human motion detection, Occlusion, Recurrent Neural Network, Mask Region-based Convolutional Neural Network, Multiple Hypothesis model, WOA-RDA*

1 Research Scholar, School of Computer Science & Engineering, Lovely Professional University, Punjab-144411, India; Assistant Professor, Department of Computer Science & Engineering, Swami Keshvanand Institute of Technology, Management & Gramothan, Jaipur-302017, India
ORCID ID : 0000-0003-3662-3472
Email: nega.cheltha@skit.ac.in
2Associate Professor, School of Computer Science & Engineering, Lovely Professional University, Punjab-144411, India
ORCID ID : 0000-0002-6423-0230
Email: chiragsharma1510@gmail.com
3Professor, Department of Computer Science & Engineering, Swami Keshvanand Institute of Technology, Management & Gramothan, Jaipur, Rajasthan-302017, India
ORCID ID : 0000-0001-5783-1989
Email: pankajdadheech777@gmail.com
4Professor, Department of Computer Science & Engineering, Poornima Institute of Engineering and Technology, Jaipur, Rajasthan-302020, India
ORCID ID : 0000-0001-5442-5870
Email: dinesh8dg@gmail.com
** Corresponding Author Email: nega.cheltha@skit.ac.in*

1. Introduction

Human Motion Detection (HMD), within the broader domain of computer vision, involves processing video or image inputs to comprehend and identify human movement. Envision a computer program capable of interpreting and recognizing human actions within pictures or videos – this capability encapsulates the essence of Human Motion Detection (HMD). Essentially, it endows the computer with the ability to perceive and comprehend human behaviour visually. This facet of computer vision has become a focal point for researchers, reflecting the intricate nature of extracting meaningful insights from visual data [1]. The primary objective of HMD is to discern and categorize human motion within visual data. However, this proposed work is confronted with a notable challenge, namely occlusion. Occlusion manifests in two distinct forms: self-occlusion and external occlusion. Self-occlusion pertains to instances where a part of the human figure conceals another part of its own body within visual data, posing a significant hurdle in deciphering human gestures and postures in computer vision. External occlusion occurs when external objects obstruct parts of the human body in visual data. This research specifically directs its attention to the complexities associated with self-occlusion and partial occlusion, aiming to enhance the understanding and mitigation of these challenges within the realm of Human Motion Detection.

HMD finds diverse applications across various domains. Examples include extracting human gestures from visual data, enabling autonomous vehicles to perceive the road environment, healthcare applications such as fall detection, identifying abnormal human activities from visual data, discerning player actions in sports, and enhancing surveillance capabilities [2]. In this study, our emphasis is on discerning human motion in scenarios both with and without occlusion. To achieve this, we employ three distinct methods for detecting human motion.

A. Motion Segmentation
B. Object Classification
C. Tracking Algorithm

In visual data, we encounter various objects, some stationary and others in motion. Motion segmentation, a crucial step, helps identify only the elements that are moving, like birds, humans, or fans. In our proposed method, we use Mask RCNN to precisely pinpoint these moving objects. Once identified, we focus exclusively on humans through object classification. In our approach, Recurrent Neural Networks (RNN) trained with 2D representations of 3D skeletal motion are employed for human recognition. Next, we need to track the movements of these identified humans. However, Human Motion Detection (HMD) faces a significant challenge known as occlusion. Our work specifically addresses issues related to self-occlusion and partial occlusion. To handle these challenges during occlusion, we employ Multiple Hypothesis Tracking (MHT), a method adept at managing uncertainties by maintaining and updating hypotheses about human identities and states. In our proposed method, we use the MediaPipe library in Python to estimate key points in the skeleton for each instance in the frame. However, a tricky problem arises known as "occlusion." Occlusion occurs when something hides a person in a picture or video. It could be the person themselves hiding parts of their own body, known as "self-occlusion," or other elements in the environment hiding people, termed "external occlusion." This hiding makes it challenging for computers to see and understand a person's full motion. Our goal in this work is to address partial occlusion and self-occlusion, as these challenges impact the accuracy and performance of Human Motion Detection (HMD). To enhance the robustness of our approach, we incorporate insights from various datasets, especially those involving self and partial occlusion.

In our research, we address the challenge of occlusion within the realm of Human Motion Detection (HMD). Our method incorporates the use of Mask R-CNN, an advanced computer program designed to identify moving objects within video sequences. Additionally, we employ a Recurrent Neural Network (RNN), a sophisticated computational model, to discern whether the detected movement corresponds to a human. This work implements the optimized hybrid Whale Optimization Algorithm and Red Deer Algorithm (WOA-RDA) in conjunction with a Recurrent Neural Network (RNN) to enhance the classification process for determining human presence. For continuous tracking of individuals, even during instances of self-occlusion and partial occlusion, we leverage the capabilities of Multiple Hypothesis Tracking (MHT). An important aspect of our methodology involves training the computer to understand human motion by exposing it to images that depict the skeletal structure, including bones and joints. This training process enables the computer to effectively learn and recognize various moves or gestures, contributing to an enhanced interpretation of human actions.

We evaluated our approach using diverse sets of images and videos, encompassing scenarios involving both self-occlusion and partial occlusion in the visual data. For this research, we utilized Python 3.10.8, the MediaPipe library, and an Intel Core i5 processor. Our experiments demonstrate the effectiveness of our computer program in accurately identifying and interpreting human movements, even when individuals are partially concealed. This research contributes to the advancement of systems capable of comprehending human actions across a range of situations.

This paper is organized as follows to ensure a coherent presentation: Section II provides background information and situates the study within the existing research landscape on human motion detection and occlusion handling. Section III delves into the specific components: Mask RCNN, RNN, and Multiple Hypothesis Tracking which provides a detailed exploration. In Section IV, we present the experimental findings along with meticulous analysis. Finally, Section V concludes by summarizing contributions and outlining potential directions for future research work.

2. Related Work

Recently, researchers have shown a significant increase in their interest in exploring human motion detection. Moreover, a multitude of studies in this field have been undertaken, examining scenarios involving human motion detection both with and without occlusion. The subsequent paragraphs will provide an in-depth exploration of the various tracking approaches that researchers have examined within this domain.

2.1 Spatio-Temporal Clustering

Mingjun Sima introduced a novel Key Frame Extraction Algorithm designed for Human Action Videos using dynamic spatiotemporal slice clustering. The algorithm strategically chooses slice positions by analyzing the human mask heat map, facilitating the extraction of meaningful spatiotemporal slice images. Keyframes are then identified based on clustering outcomes. The existing algorithm demonstrated strong performance with a recall of 96%, precision of 93%, and an F1 score of 94% [3]. In our proposed approach, we achieved further improvement, reaching a recall of 98%, precision of 98%, and an enhanced F1 score of 98%.

In a study conducted by Liang et al., the investigation into spatiotemporal slices for video caption extraction was

carried out. Distinctive bar-code-like patterns emerge when caption pixels are present along horizontal or vertical scan lines in a spatiotemporal slice. Incorporating structural information from these patterns in both horizontal and vertical slices allows precise localization of spatial and temporal positions of video captions [4]. However, there is room for enhancing the effectiveness of the results obtained from this approach.

2.2 Tracking Via Machine Learning Algorithms

Kocabas et al. introduced EpipolarPose, a pioneering self-supervised learning technique tailored for 3D human pose estimation. The methodology involves the estimation of 2D poses from multi-view images, utilizing epipolar geometry to derive both 3D poses and camera geometry. The obtained 3D pose and camera geometry data are integral to training a 3D pose estimator [5]. In our proposed methodology, we have successfully enhanced the percentage of key points compared to the existing approach, showcasing the efficacy of our improvements. Chen et al. presented an innovative unsupervised learning methodology for recovering 3D human pose from 2D skeletal joints obtained from a single image [6]. The approach includes a lifting network that takes 2D landmarks as input and generates an accurate 3D skeleton estimation. During training, the recovered 3D skeleton is reprojected from random camera viewpoints, generating synthetic 2D poses. The lifting of synthetic 2D poses back to 3D and re-projecting them in the original camera view allows for the definition of self-consistency loss in both 3D and 2D spaces. In our proposed work, we have achieved a superior percentage of key points compared to this existing approach, underscoring the effectiveness of our enhancements.

Wang et al. conducted an in-depth investigation into 3D Human Pose Machines using Self-supervised Learning [7]. Their research proposed an efficient self-supervised correction mechanism designed to acquire a comprehensive understanding of intrinsic human pose structures. The mechanism encompassed two dual learning tasks: the 2D-to-3D pose transformation and the 3D-to-2D pose projection. These tasks served as a crucial linkage between 3D and 2D human poses, facilitating a form of "free" self-supervision to enhance the precision of 3D human pose estimation. The percentage of key points achieved in this approach was found to be lower than in our proposed work, highlighting the robustness and effectiveness of our proposed methodology. Their method involved leveraging estimated 2D confidence heat maps of key points and integrating an optical-flow consistency constraint to filter out unreliable estimations of occluded key points. In instances of occlusions, incomplete 2D key points were utilized, feeding them into 2D and 3D temporal convolutional networks to enforce temporal smoothness, ultimately yielding a comprehensive 3D pose. Significantly, by using incomplete 2D key points rather than complete yet incorrect ones, their networks demonstrated reduced susceptibility to error-prone estimations of occluded key points. Training the occlusion-aware 3D Temporal Convolutional Network (TCN) required annotated pairs of a 3D pose and a 2D pose with occlusion labels. To facilitate training, they projected the model onto a 2D plane from various viewing angles, enabling the acquisition and labeling of occluded key points, thus creating a rich dataset for training. Importantly, our proposed methodology surpasses the accuracy of the existing approach, showcasing advancements in occlusion handling for 3D human pose estimation in video.

Ghazal et al. introduced a method incorporating 2D skeleton data and supervised machine learning for human activity recognition [8]. While their approach is notable, the percentage of keypoints achieved in their study is lower than what is demonstrated in our proposed work. This discrepancy underscores the advancements and superior performance of our proposed methodology in the domain of human activity recognition.

2.3 Tracking Via Deep Learning Algorithms

Wandt et al. investigated titled 'RepNet: Weakly Supervised Training of an Adversarial Reprojection Network for 3D Human Pose Estimation.' Their research centered on employing a projection network (RepNet) to transform a distribution of 2D poses into a distribution of 3D poses through adversarial training. Additionally, the network estimates camera parameters, establishing a network layer for reprojection from estimated 3D poses back to 2D, introducing a reprojection loss function [9]. However, the achieved percentage of key points in their study falls short of the performance demonstrated in our proposed approach.

Changai et al. conducted a study aimed at refining Key Frame Extraction for Sports Training through advancements in deep-learning techniques. The investigation focused on selecting crucial video frames from sports training videos to highlight specific actions during the training process. Their methodology involved the use of a fully convolutional network (FCN) to extract the region of interest (ROI) for pose detection in frames. Subsequently, a convolutional neural network (CNN) was employed to estimate the pose probability of each frame. Notably, they introduced a distinctive key frame extraction method that considered probability differences among neighboring frames [10]. In contrast to existing approaches, our proposed methodology showcased superior accuracy, sensitivity, specificity, and classification rates.

Meng et al. undertook a thorough investigation with a specific focus on applying Deep Key Frame Extraction for Sports Training. The research introduced an innovative deep key frame extraction methodology tailored to address the inherent complexities in such sports training videos. To mitigate the challenges posed by intricate backgrounds, the researchers employed Fully Convolutional Networks (FCN) to accurately isolate the region of interest (ROI). Following this, Convolutional Neural Networks (CNN) were utilized to estimate the pose probability of each frame within the identified ROI. Additionally, the study introduced a unique

variation-aware key frame extraction approach, considering the differences in probabilities among neighboring frames [11]. Although achieving a recall rate of 95% and a precision rate of 92%, these results were comparatively lower than the outcomes observed in our proposed methodology.

The R-CNN algorithm is employed to scrutinize video frames, which are subsequently transformed into a 3D space. Following this transformation, 3D space coordinates are acquired and utilized with trained dataset models to discern human motion [12]. For human motion analysis in sports competitions, an algorithm based on KELM-MFF is devised [13]. Time templates are utilized to analyze Deep Learning features within video sequences, although this approach falls short of thoroughly examining color characteristics. Consequently, a focus on color characteristics becomes imperative. The application of edge detection eliminates redundant image areas, and CNN is employed for classification. While enhancing accuracy, recognition rate, specificity, and sensitivity, it's noteworthy that the study was conducted on a limited-scale dataset [14]. Nitin et al. introduce the utilization of the Temporal Convolutional Network (TCN) architecture for activity recognition based on smartphone-collected sensor data. TCN's adaptability in handling input sequences of varied lengths and capturing long-term dependencies results in superior activity recognition accuracy compared to other deep learning methods [15]. Yair A. Andrade et al. propose a novel approach using the Temporal Convolutional Neural Network (TCNN) for human activity analysis and classification [16]. The TCN architecture prioritizes minimized computational requirements, rendering it compact and rapidly trainable compared to other networks. This makes it suitable for real-time analysis and recognition of human activities, particularly in resource-limited settings. Alzahrani et al. proposed fall detection using Microsoft Kinect v2[17]. However, the percentage of keypoints achieved in their study is lower than what is demonstrated in our proposed work. Franco et al. proposed a multimodal approach for human activity recognition based on skeleton and RGB data [18]. Nevertheless, the percentage of keypoints achieved in their study is lower than what is demonstrated in our proposed work.

As asserted by Gaud et al. [19], a predominant consensus among researchers advocates the utilization of potent deep learning techniques, including Convolutional Neural Networks (CNN), Inception CNN, Long Short-Term Memory (LSTM), Bidirectional LSTM (Bi-LSTM), and hybrid approaches for various applications. It is noteworthy that the human walking pattern serves as a reflective indicator of an individual's health condition.

Yang et al. conducted an exploration into geometric structure information by employing a two-layer convolutional network in conjunction with a correlation filter [20]. Jack et al. introduced an end-to-end algorithm that combined correlation filters and neural networks. Correlation filter tracking offers advantages in tracking targets with minimal prior knowledge and high-speed tracking capabilities. On the other hand, deep Convolutional Neural Networks (CNNs) exhibit robust representation abilities by extracting deep features. The fusion of correlation filters and deep CNNs, known as deep correlation filter tracking, has gained attention in target tracking. However, the real-time performance of deep correlation filter tracking is often hindered by the intricate network structure and the computational intensity of deep CNNs. To address this, Yue Yuan et al. utilized ResNet to generate response maps, fused through the AdaBoost algorithm along with a scale filter. Despite its merits, the computational burden of deep CNNs poses challenges in achieving real-time performance, especially when objects share similarities, impacting the overall detection system [21]. An enhanced Particle Filter is employed by Yuan et al. to extract high-order features, combining color features of the target with template-derived features using convolutional networks [22]. This fusion ensures a comprehensive representation of observed data, incorporating the target's color information with distinct visual characteristics extracted through convolutional networks. Hayat et al. utilized Long Short-Term Memory (LSTM) to recognize Human Activity in Elderly People. Despite achieving a commendable accuracy of 95.04%, the Long Short-Term Memory Network's performance falls short of the accuracy demonstrated in our proposed work [23]. Rui Ma et al. employed a deep convolutional generation confrontation network for human motion pose recognition, utilizing a deep convolutional stacked hourglass network to precisely extract the positions of key joint points in the image [24]. However, the percentage of keypoints achieved in their study is lower than that demonstrated in our proposed work. Angelini et al. introduced a Human Activity Recognition (HAR) approach relying on OpenPose for pose extraction [40]. Their methodology involved extracting both low- and high-level features from body poses using a 1D CNN and an LSTM for classification. To simulate real-world scenarios, they utilized actual CCTV data with partial body occlusion and generated synthetic occluded data by removing specific body parts. Through experimentation, they incorporated occluded samples in training, demonstrating that this strategy significantly enhances performance in scenarios involving both occlusion and missing data.

2. 4 Tracking Using Filter

Fakhreddine et al. made significant strides in Human Pose Estimation by harnessing the capabilities of a Catadioptric Sensor in Unconstrained Environments through the implementation of an Annealed Particle Filter [25]. This study stands out for its integration of variations to compute gradients on spherical images, leading to the development of a robust descriptor. When coupled with an SVM classifier for human detection, this descriptor notably contributes to improved accuracy. In comparison to the existing approach, our proposed methodology demonstrates a substantial enhancement in accuracy. Particle filters play a pivotal role

in human tracking, addressing challenges such as occlusion and ensuring smooth and continuous imagery [26]. In this pursuit, the system selects highly weighted samples for effective human motion tracking. Notably, particle filters find application in both single-target and multiple-target tracking scenarios [27]. This adaptive approach showcases the adaptability and effectiveness of particle filters in managing diverse challenges inherent in human pose estimation and tracking within dynamic environments.

The incorporation of the Correlation method leads to significant improvements in object detection accuracy and success rates, albeit with a marginal reduction in running speed [28]. Somayyeh et al. introduced a hybrid algorithm that combines a particle filter and genetic algorithm for efficient target tracking [29]. This research introduces novel concepts, including "marking" (where users define the target in the first frame of a video sequence) and "image size reduction." The implementation of these concepts results in a reduced number of particles, decreased processing time for each frame, and an overall reduction in tracking time. Additionally, the marking idea notably enhances the performance of the proposed RPFGA method in addressing occlusion challenges. Results from various challenges, including occlusions (OCC), demonstrate an improvement in F-measure in our proposed work compared to the existing approach. While correlation filter-based object tracking enhances efficiency [30], our proposed work prioritizes accuracy and outperforms the existing approach in this regard.

3. Proposed Work

In this research endeavor, our central objective is to propel the field of Human Motion Detection (HMD) forward by addressing a persistent challenge—occlusion. Occlusion arises when portions or the entirety of an individual are obscured in images or video frames, presenting a significant impediment to precisely identifying and tracking human motion. To surmount this challenge, we advocate for a comprehensive approach that integrates cutting-edge techniques in motion segmentation, object classification, and tracking algorithms. Our proposed methodology harnesses the capabilities of modern computer vision and machine learning methodologies to augment the resilience and precision of HMD systems, particularly in scenarios involving occlusion. The fundamental components of our approach include:

We employ the Mask Region-based Convolutional Neural Network (Mask R-CNN) for motion segmentation, leveraging this advanced deep learning model's proficiency in precisely identifying and delineating moving objects within video streams. By integrating Mask R-CNN into our framework, our objective is to enhance the accuracy of identifying and isolating human motion, particularly in scenarios with occlusion. Object classification is a pivotal step in distinguishing whether a moving entity is a human or another object. To address this, we propose the use of Recurrent Neural Networks (RNN), trained with 2D representations of 3D skeletal motion. This approach enables the system to learn and recognize human-specific movement patterns, contributing to more accurate object classification even in occluded scenarios. The Multiple Hypothesis Tracking (MHT) plays a crucial role in our approach to tackle occlusion during tracking. MHT enables the system to maintain multiple hypotheses about the identity and location of tracked objects, proving particularly valuable in scenarios of self-occlusion, where parts of the human body may temporarily hide others. To assess the effectiveness of our proposed methodology, we conduct experiments using diverse datasets that encompass scenarios both with and without occlusion. This allows us to evaluate the system's performance in challenging real-world conditions.

A distinctive aspect of our approach involves the utilization of 2D representations of 3D skeletal motion. We investigate the efficacy of this representation in enhancing the system's understanding of human actions, contributing to improved object classification and tracking, particularly in occluded scenarios. Through our proposed work, we aim to contribute to the advancement of Human Motion Detection (HMD) systems, enhancing their resilience in scenarios with occlusion. The integration of cutting-edge technologies and novel representations is anticipated to result in improved accuracy and reliability, paving the way for enhanced applications in surveillance, gaming, healthcare, and beyond. Algorithm 1 provides a detailed explanation of our proposed work, outlining the processing steps for detecting instances of human motion within an input stream of video frames.

The key inputs include the individual frame "I_t," the current frame index "t," the total number of frames "n," and the Video Dataset "V." "M_t^i" signifies the segmented masks pertaining to individual instances, while "IoU" represents the Intersection over Union, and "IoU_t" symbolizes the IoU threshold. The algorithm commences by initializing critical thresholds, including IoU threshold ("IoU_t"), Euclidean Distance threshold ("D_t"), Statistical Threshold ("ST"), and Hypothesis Probability Threshold ("HPT"). For each frame iteration within the range of the total frames, the algorithm undergoes a sequence of essential steps. It commences by preprocessing the current frame "I_t" and applying the Modified Mask R-CNN technique to derive segmented masks "M_t^i" for individual instances. Within this loop, the algorithm evaluates each instance "i" to determine its consistency across frames using IoU calculations as follows

$$IoU\left(M_t^i, M_{t-1}^i\right) = \frac{|M_t^i \cap M_{t-1}^i|}{|M_t^i \cup M_{t-1}^i|} \quad (1)$$

Instances exhibiting IoU greater than the threshold are considered as consistent entities. Statistical Thresholding is used in the proposed work. Subsequently, Euclidean Distance between centroids is computed to gauge instance movement, with those surpassing the Euclidean Distance

threshold marked as "in motion." Euclidean distance is calculated as follows

$$D_i = \sqrt{\left(x_t^i - x_{t-1}^i\right)^2 + \left(y_t^i - y_{t-1}^i\right)^2} \quad (2)$$

Further, the algorithm calculates Motion Magnitude ("M_m") and derives Z-scores for instances based on Motion Magnitude, Mean Motion ("M_{em}"), and Standard Motion ("S_m"). The motion magnitude is calculated as the Euclidean distance between the centroids of an object or region in consecutive frames. The formula for calculating the mean motion (M_{em}) is:

$$M_{em} = \frac{\left(\sum M_{m_i}\right)}{N} \quad (3)$$

Where: Σ represents the sum across all instances in the dataset. M_{m_i} is the motion magnitude of instance i.N is the total number of instances in the dataset. Standard Motion ("S_m") is calculated as follows

$$S_m = \frac{\sqrt{\left(\sum M_{m_i} - M_{em}\right)^2}}{N} \quad (4)$$

Where:Σ represents the sum across all instances in the dataset. $M_{m_i}$is the motion magnitude of instance i. M_{em} is the mean motion magnitude of the dataset. N is the total number of instances in the dataset. Z-score is calculated as follows.

$$\text{Z-score} = \frac{\left(M_{m_i} - M_{em}\right)}{S_m} \quad (5)$$

If the absolute value of the Z-score exceeds the Statistical Threshold ("ST"), the algorithm then feed segmented mask M_t^i into the RNN for classification and then generates hypotheses using the Multiple Hypothesis Tracking ("MHT"). The algorithm computes a Hypothesis Probability ("P_h") for each hypothesis generated (H_i), and if this probability exceeds the Hypothesis Probability Threshold ("HPT"), the hypothesis is tagged as a legitimate motion instance. Finally, the algorithm presents the observed occurrences of human motion via output graphics. This technique is a complete framework for human motion detection that employs Mask R-CNN and a variety of associated thresholds and hypothesis generating procedures to detect motion instances even in complex settings.

3.1 Mask R-CNN

In our proposed methodology, we leverage the Mask Region-based Convolutional Neural Network (Mask R-CNN) to tackle the intricate task of motion segmentation, particularly in the context of human activities. Mask R-CNN stands out as a formidable deep learning model renowned for its prowess in instance segmentation, a nuanced process involving the precise identification and delineation of individual objects within an image or video sequence. To delve into the specifics, the Mask R-CNN architecture commences its operations with a backbone Convolutional Neural Network (CNN), and in our innovative approach, we opt for ResNeXT as the chosen backbone. ResNeXT represents an evolution of the ResNet architecture, tailored to optimize training efficiency and elevate performance in image classification tasks. The distinctive features of ResNeXT encompass the incorporation of "cardinality" and the introduction of "cardinality groups" within a ResNeXT block. Here, cardinality refers to the count of independent pathways or groups within a given ResNeXT block. This departure from the conventional approach of merely increasing the number of filters involves the integration of parallel pathways, each with its set of filters. These parallel pathways, organized into cardinality groups, enable the network to learn diverse features more comprehensively. In simpler terms, each cardinality group specializes in capturing specific features, and their outputs are thoughtfully combined through concatenation, resulting in a richer and more nuanced representation of the input data. The ResNeXT block, constituting multiple parallel branches with individual convolutional layers and filters, operates in concert to contribute to the final output. The brilliance of this design lies in its ability to harness the power of parallel processing, allowing the network to capture a wide range of features simultaneously. The inclusion of a shortcut connection further streamlines the flow of information within the network, enhancing its overall efficiency. In essence, our implementation of Mask R-CNN, coupled with ResNeXT as the backbone, is geared towards elevating the accuracy and efficiency of human motion segmentation, especially when confronted with challenges like occlusion. This innovative fusion of cutting-edge technologies in deep learning aims to contribute significantly to the advancement of Human Motion Detection (HMD) systems in real-world scenarios.

In the Mask R-CNN framework, the Region Proposal Network (RPN) plays a pivotal role in generating a collection of bounding box proposals, commonly referred to as anchor boxes. These proposals are determined based on the features extracted by the underlying Convolutional Neural Network (CNN). Anchor boxes serve as potential representations of object locations across various scales and aspect ratios. The RPN assigns scores to each of these proposals, reflecting the likelihood that a given proposal encompasses an object. For every anchor box, the RPN makes two distinct predictions for each spatial location within the feature map F: the probability of the box containing an object (P_obj) and the refined coordinates of the box (Δbox).

The mathematical expressions for these predictions are articulated as follows: P_obj (Probability of objectness for an anchor box):

$$P\_obj = \text{sigmoid}(F\_cls) \quad (6)$$

where F_cls represents the RPN's classification score for the anchor box. Δbox (Bounding box regression values for an anchor box):

$$\Delta box = F\_reg \quad (7)$$

High-scoring proposals are singled out for subsequent analysis. Region of Interest (RoI) Align comes into play, extracting features from the suggested regions by utilizing the features acquired from the underlying Convolutional Neural Network (CNN). This guarantees precise alignment of features with the spatial positions within each proposed region. The RoI head then engages in predicting class probabilities for each proposed region through the application of a softmax function. Let C denote the number of classes; thus, for every RoI,

$$P\_class = softmax(F\_cls) \quad (8)$$

where F_cls denotes the classification score for each class.

The instance head comprises two parallel branches: one dedicated to bounding box regression and the other to object classification. Bounding box regression fine-tunes the initially proposed boxes to better match the true shapes of objects, while the object classification branch assigns a class label to each proposed region, identifying the object type it represents. Concurrently, the mask head takes on the responsibility of predicting segmentation masks for each proposed region, generating binary masks for each class that precisely delineate the object's boundaries within the proposed region. To ensure accurate mask predictions, particularly for small or irregularly shaped objects, Mask R-CNN employs a pixel-to-pixel alignment strategy. The model undergoes training using multiple loss functions, encompassing bounding box loss, classification loss, and mask loss. This combination of losses ensures that the model learns to predict bounding boxes, object classes, and segmentation masks with high accuracy. In our proposed approach, the pre-trained Mask R-CNN model is employed on each frame of a video sequence. The model effectively segments and delineates regions corresponding to moving humans, producing precise masks that highlight the spatial extent of each person's motion. Mask R-CNN exhibits notable robustness to occlusion, accurately segmenting visible portions of humans, even in scenarios involving partial occlusion by other objects or the individuals' own body parts. The architecture of Mask R-CNN is illustrated in the following Figure 1.

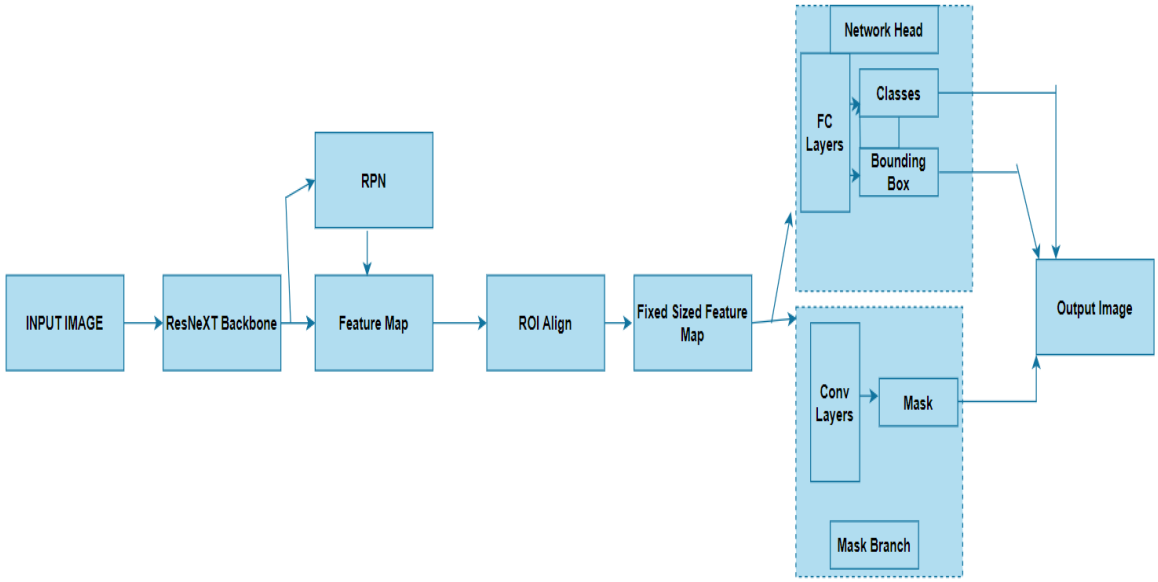


Fig.1: Architecture of Modified Mask RCNN with ResNeXT Backbone

3.2 RNN

In our proposed approach, Recurrent Neural Networks (RNNs) play a pivotal role in the classification task. RNNs are a specialized type of artificial neural network designed to handle sequential data, making them particularly effective for tasks involving temporal dependencies, such as the classification of human motion. In the realm of human motion, RNNs excel at learning patterns and relationships over time, enabling them to identify and classify diverse types of movements or behaviors. The unique strength of RNNs lies in their ability to process sequences of data, making them well-suited for tasks where the order of input information holds significance. In the context of human motion classification, the sequential nature of motion data, captured over time, is crucial for comprehending various actions. Within an RNN, memory cells store information about preceding time steps in the sequence, allowing the network to maintain context and capture temporal dependencies. The hidden state of the RNN serves as a memory repository, enabling the network to remember patterns from earlier time steps. To tackle the challenge of learning long-term dependencies, advanced RNN variations like Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) have been introduced. These

architectures incorporate mechanisms to selectively remember or forget information, enhancing their capability to capture more extensive temporal dependencies. In the context of human motion classification, the input to the RNN comprises sequences of features representing motion data, such as joint angles, positions, velocities, or other relevant information characterizing the motion. The RNN processes these input sequences over time, updating its hidden state at each time step, which acts as an internal representation of the network's understanding of the input sequence up to the current time step. The output layer of the RNN is responsible for predicting the class or label associated with the input motion sequence. This layer can utilize the final hidden state or aggregate information from multiple time steps to make a robust classification decision.

RNNs undergo training utilizing the Backpropagation Through Time (BPTT) algorithm, an extension tailored for feedforward neural networks. BPTT computes gradients across the entire sequence, enabling the network to learn from temporal dependencies within the training data. The loss function quantifies the variance between predicted class probabilities and true labels. Throughout the training, the network refines its parameters to minimize this loss, thereby enhancing its capacity to accurately categorize human motions. Applications of RNN in Human Motion Classification encompass diverse domains: Gesture Recognition, Activity Classification, and Biomechanical Analysis. RNNs excel in classifying hand gestures or body movements, finding applications in human-computer interaction and gesture-based control systems. RNNs demonstrate proficiency in classifying intricate activities by discerning patterns in sequential data. This capability is valuable in surveillance or healthcare for monitoring human actions. In sports science or rehabilitation, RNNs play a pivotal role in classifying and analyzing human motion patterns, offering insights into biomechanical aspects. RNNs present a robust approach to human motion classification, harnessing their capability to capture temporal dependencies in sequential data. Detailed formulas are provided below for a comprehensive understanding. The hidden state (h_t)in a basic RNN at each time step (t) is updated using the input at the current time step (x_t) and the hidden state from the previous time step (h_{t-1})

$$h_t=\boldsymbol{\sigma}(W_{hx} \,.\, x_t + W_{hh} \,.\, h_{t-1} + b_h) \quad (9)$$

Where W_{hx} denotes weight matrix for the input, W_{hh} is the weight matrix for the hidden state, b_h is the bias term, and $\boldsymbol{\sigma}$ is an activation function, often it is the hyperbolic tangent (tanh).

The LSTM includes more sophisticated mechanisms to capture long-term dependencies. The LSTM hidden state (h_t) and cell state (C_t) are updated at each time step (t) using input (x_t), previous hidden state (h_{t-1}), and previous cell state (c_{t-1}).

$$f_t=\boldsymbol{\sigma}(W_f \,.\, [h_{t-1},x_t] + b_f) \quad (10)$$

$$i_t=\boldsymbol{\sigma}(W_i \,.\, [h_{t-1},x_t] + b_i) \quad (11)$$

$$ć_t=\tanh(W_c \,.\, [h_{t-1},x_t] + b_c) \quad (12)$$

$$c_t=f_t \,.\, c_{t-1} + i_t \,.\, ć_t \quad (13)$$

$$o_t=\boldsymbol{\sigma}(W_o \,.\, [h_{t-1},x_t] + b_o) \quad (14)$$

$$h_t= o_t \,.\, \tanh(c_t) \quad (15)$$

Where f_t, i_t, o_t are the forget, input and output gates, $ć_t$ is the candidate cell state, W_f, W_i, W_c, W_o are weight matrices, b_f, b_i, b_c , b_o are bias vectors and $\boldsymbol{\sigma}$ is the sigmoid activation function.

The output layer (y'_t) of the RNN is responsible for predicting the class or label associated with the input motion sequence. It can use the final hidden state or aggregate information from multiple time steps:

$$y'_t= \text{Dense}(h_t) \quad (16)$$

Dense is a fully connected layer that maps the hidden state to the output space. RNNs are trained using the BPTT algorithm, an extension of the backpropagation algorithm for feedforward neural networks. BPTT calculates gradients over the entire sequence.

$$\frac{\delta L}{\delta \theta} = \sum_{t=1}^{T} \frac{\delta L_t}{\delta \theta} \quad (17)$$

Where L_t is the loss at time step t, θ represents the model parameters. The loss function measures the difference between the predicted class probabilities (y'_t) and the true labels (y_t).

$$L_t(y_t,y'_t) = - \sum_i y_{t,i} \,.\, \log(y'_{t,i}) \quad (18)$$

Where $y_{t,i}$ and $y'_{t,i}$ are the true and predicted probabilities for class I at time step t. During training, the network adjusts its parameters (θ) using gradient descent to minimize the overall loss.

$$\theta_{t+1} = \theta_t - \eta \,.\, \frac{\delta L}{\delta \theta} \quad (19)$$

Where η is the learning rate. RNNs offer a powerful approach to human motion classification by capturing temporal dependencies in sequential data, providing valuable insights in various domains.

3.3 Hybrid RDA-WOA

The advantages of two nature-inspired algorithms are combined in a revolutionary meta-heuristic technique called the hybrid red deer and whale optimization algorithm (or RDA and WOA, respectively). The RDA explores the search space and identifies the optimal solutions by imitating red deer's natural behaviors, such as fighting, shouting, and mating. To identify the most promising areas and improve the solutions, the WOA mimics the humpback whale's hunting tactics, which include encircling, bubble-netting, and spiral attacks. In order to balance population diversity

and convergence, the hybrid algorithm uses the RDA for global exploration and the WOA for local exploitation. Through the incorporation of the suggested hybrid WOA-RDA optimized RNN technique, there is a notable improvement in classification accuracy, precision, recall, and F1 score, accompanied by swift convergence speed.

3.4 Multiple Hypothesis Tracking

In our proposed methodology, we employ Multiple Hypothesis Tracking (MHT) to navigate the challenges of uncertainty, partial occlusion, and self-occlusion prevalent in object-tracking scenarios. MHT, a proven tracking method in computer vision and target tracking applications, specializes in addressing the complexities associated with occlusion, including self-occlusion and partial occlusion. This method operates by concurrently managing and evaluating multiple hypotheses related to the identity and location of the tracked objects. This adaptability proves invaluable in scenarios where objects may encounter temporary concealment or overlap. In the MHT framework, each tracked object is linked with several hypotheses, with each hypothesis representing a potential assignment of an object to a track or detection. The maintenance of multiple hypotheses is crucial to accommodating uncertainty, particularly in occlusion scenarios. MHT operates through a prediction-update framework. During the prediction step, the existing hypotheses are projected forward based on the anticipated motion of the object. Subsequently, in the update step, new detections or observations contribute to the evaluation and adjustment of the existing hypotheses. The primary challenge in tracking lies in associating observations (such as detections or measurements) with existing tracks or establishing new tracks for unmatched observations. MHT addresses this challenge by scrutinizing various hypotheses to ascertain the most probable association between observations and established tracks. Notably, MHT excels in handling self-occlusion, where one part of an object may briefly obscure another part. The incorporation of multiple hypotheses allows the tracker to consider diverse possibilities, ensuring accurate associations when the occluded part becomes visible again. Additionally, when external occlusion causes partial visibility of an object, MHT maintains multiple hypotheses concerning the object's location and identity. This flexibility enables the tracker to adapt to situations where the complete object is not observable in every frame.

MHT initiates by establishing numerous hypotheses for each tracked object, rooted in initial detections. These hypotheses undergo forward projection, aligning with the anticipated motion of the object. MHT systematically assesses associations between existing tracks and incoming observations, computing the likelihood of each observation being linked to the track for every hypothesis. To streamline computational processes and prioritize more viable hypotheses, those with low likelihoods are pruned. In instances where observations don't align with existing tracks, new hypotheses and tracks may emerge. The continual updating of hypotheses, informed by fresh observations, enhances the precision of object location and identity estimates. The adaptive nature of the multiple hypotheses framework empowers the tracker to navigate occlusion scenarios adeptly, ensuring resilient tracking even when objects experience temporary concealment. MHT, demonstrating robustness in scenarios marked by heightened uncertainty, such as occlusion, strategically explores diverse possibilities and adjusts to evolving conditions. Crucially, MHT is well-suited for real-time implementation in tracking systems, rendering it applicable across domains like surveillance, robotics, and beyond.

3.5 MediaPipe

MediaPipe, an open-source framework developed by Google, offers a comprehensive suite of pre-built solutions and tools for creating applications with perception features, encompassing face detection, hand tracking, pose estimation, and more. This framework streamlines the development of applications involving computer vision and machine learning tasks by providing user-friendly APIs and ready-made models. In our proposed work, we employ MediaPipe for skeleton images. MediaPipe features a pre-trained pose estimation model designed for real-time estimation of the human body's pose. This model adeptly detects and tracks key body landmarks, including key points on the head, torso, arms, and legs. The pose model in MediaPipe identifies 33 key points in total. The ready-to-use pre-trained pose estimation model from MediaPipe is easily integrated into applications. Trained on a diverse range of human poses, the model is optimized for real-time performance. Typically, the pose estimation model is utilized within the broader MediaPipe Pose solution, incorporating components for landmark detection, pose tracking, and rendering of the estimated poses. Fitness applications leverage pose estimation for tracking users' body movements during exercises, providing valuable feedback on form and posture. In retail, pose estimation facilitates virtual try-on experiences, enabling users to visualize how clothing items fit on their bodies. In the gaming domain, pose estimation creates interactive experiences by translating users' body movements into in-game actions. Pose models also contribute to accessibility features, offering gesture-based control options for individuals with mobility challenges. Figure 2 illustrates pose models, as depicted by Google Developers [41]. Figure 2 shows the pose model of MediaPipe and Figure 3 shows the workflow of the proposed work.

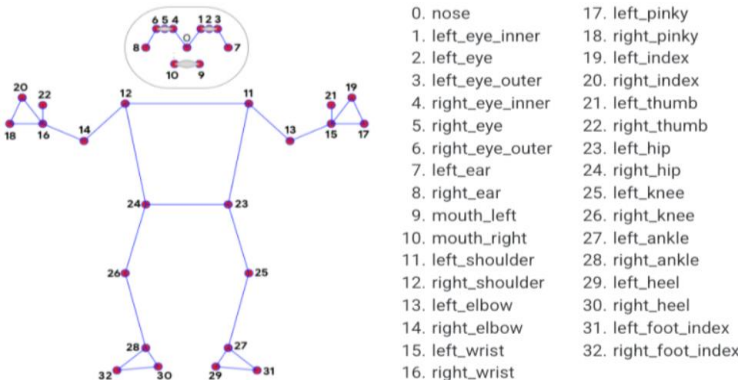


Fig 2. Pose Model of MediaPipe

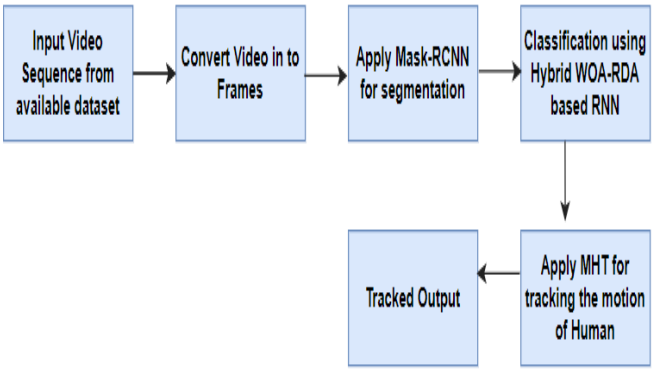


Fig 3. Workflow of Proposed Work

Algorithm : Human Motion Detection during Self and Partial Occlusion

Input : Input: Best Frames I_t

Output : Human motion Detection

I_t : Video Frame

t : index of the current frame being processed

n : total number of frames in the video

M_t^i : Segmented masks

i : Individual instances

IoU : Intersection over Union

IoU_t : IoU threshold Coordinates of two centroids $(x_t^i, y_t^i), (x_{t-1}^i, y_{t-1}^i)$

D_i : Euclidean Distance

D_t : Euclidean Distance Threshold

M_m : Motion Magnitude

M_{em} : Mean Motion

S_m : Standard motion

ST : Statistical Threshold

H : Hypotheses

MHT : Multiple Hypothesis Tracking

H_i : For each hypothesis

P_h : Hypothesis based on observed data

HPT : Hypothesis Probability Threshold

KP : Detected Keypoints on the human body

Begin

(1) Initialize IoU_t, D_t ,ST, HPT

(2) for t=2 to n do :

(a) Preprocess frame I_t

(b) Apply Mask R-CNN to each I_t to obtain M_t^i for i

(c) For each instance I in {1,2,..,k} do:

(i) Calculate $IoU\left(M_t^i, M_{t-1}^i\right) = \frac{|M_t^i \cap M_{t-1}^i|}{|M_t^i \cup M_{t-1}^i|}$

(ii) if $IoU\left(M_t^i, M_{t-1}^i\right) > IoUt$ then consider i as the same object across the frames

(iii) Calculate $D_i = \sqrt{\left(x_t^i - x_{t-1}^i\right)^2 + \left(y_t^i - y_{t-1}^i\right)^2}$

(iv) if $D_i > D_t$ then Mark instance i as in motion

(v) Calculate M_m for i

(vi) Calculate Z-score for I based on M_m, M_{em}, S_m

(vii) if |Z-score_i|>ST then

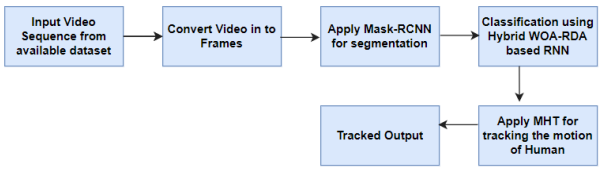
a. Extract KP from segmented mask M_t^i
b. Feed KP into the WOA-RDA RNN for classification
c. generate H using MHT

(viii) for each H in H_i calculate P_h

(a) if P_h > HPT then mark h as a valid motion instance

(3) Output Visualization

End



4. Experiments

4.1 Dataset

The datasets employed in this research encompass a diverse range of sources, namely WEIZMANN, UCF101, VIRAT, HMDB51, KTH, and the HumanEva Dataset. Table 1 provides an overview of the resolutions associated with the different datasets, while Table 2 outlines few tracking challenges inherent in specific activities within these datasets. Of note, HMDB51 encompasses 51 distinct human motions.

The WEIZMANN dataset is composed of ten distinct human motions [32], comprising a total of 93 videos with a resolution of 180 × 144. UCF101, on the other hand, features 101 diverse human motions and boasts an extensive collection of 13,320 videos sourced from YouTube. These videos maintain a frame rate of 25 frames per second, and the resolution of the UCF101 dataset stands at 320 × 240 [31].

The VIRAT dataset, known as the Video Image Retrieval and Analysis Tool (VIRAT), captures recordings from elevated vantage points with a resolution of 640 × 480 This dataset encompasses multifaceted challenges, encompassing climate variation and the presence of numerous moving objects.

The Human Motion Database (HMDB51) encompasses 51 different human motions [33], spanning across 101 videos. Schuldt's KTH dataset incorporates a comprehensive compilation of 2391 video sequences, featuring 25 actors engaged in the demonstration of six distinct actions. Each action is performed within four distinct scenarios, including outdoors with varying attire, indoor settings, and different outdoor environments. The HumanEva dataset constitutes an amalgamation of six distinct motions, encompassing activities such as walking, gestures, and jogging [34,35]. These motions are executed by four individual subjects and captured through a network of seven cameras, which includes three RGB cameras and four grayscale cameras.

Video Diver Dataset (VDD-C): The datasets employed in this research encompass more than 100,000 carefully annotated underwater images featuring divers in a variety of settings, sourced from both pool environments and the Caribbean region, with a specific focus on the waters off the coast of Barbados. These images have been extracted from video sources and are made freely accessible for use [36].

Table 1. Resolution of Various Dataset

Dataset	Resolution
Weizmann	180 × 144
HMDB51	340× 256
UCF101	320× 240
Virat	640 × 480
KTH	160 × 120

Activity Detected	Tracking Challenges
Biking	Occlusion, Low Resolution, Scale Variation
Side Jump	Occlusion, Background clutters
Skipping	Low Resolution, Rotation
Diving	Deformation, Occlusion, Background Clutters

Table 2. Tracking Challenges in Various Dataset

4.2 Result

We illustrate sample frames from various datasets in Figure 4. Specifically, Figure 4a showcases a sample frame from the VDD-C dataset, Figure 4b from the Virat dataset, Figure 4c from the KTH dataset, Figure 4d and 4e from the UCF101 dataset, and Figures 4f, 4g, and 4h from the Weizmann dataset

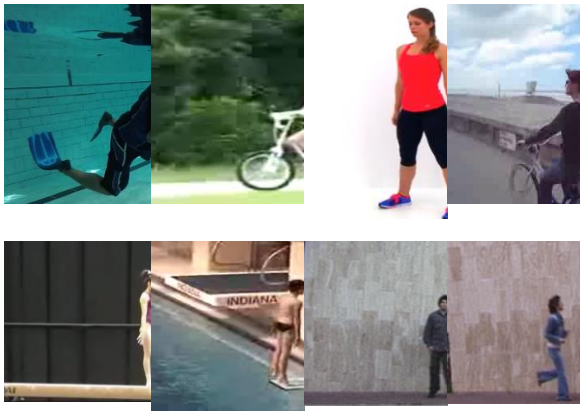


Fig 4a. Frame From VDD-C Dataset, **Fig 4b.** Frame Selected from Virat Dataset, **Fig 3c.** Frame Selected From KTH Dataset **,Fig 4d, 4e.** Frame Selected from UCF101 dataset, **Fig 4f, 3g, 3h.** Frame Selected from Weizmann dataset.

The keypoints extracted from the segmented mask M_t^i are illustrated in the subsequent Figure 5a to 5h. Figure 5a, showcasing keypoint detection from the VDD-C dataset. In Figure 5b, keypoint detection from the Virat dataset is presented, followed by Figure 5c, depicting keypoint

detection from the KTH dataset. Figures 4d and 5e feature keypoint detection from the UCF101 dataset. Figures 5f, 5g, and 5h exhibit keypoint detection from the Weizmann dataset.

Table 3. Correctly and Wrongly Predicted Frames

Activity	Total	Correct	Wrong
Biking	132	131	1
Side Jump	95	93	2
Skipping	102	100	2
Diving	151	149	2
Balancing Beam	123	122	1

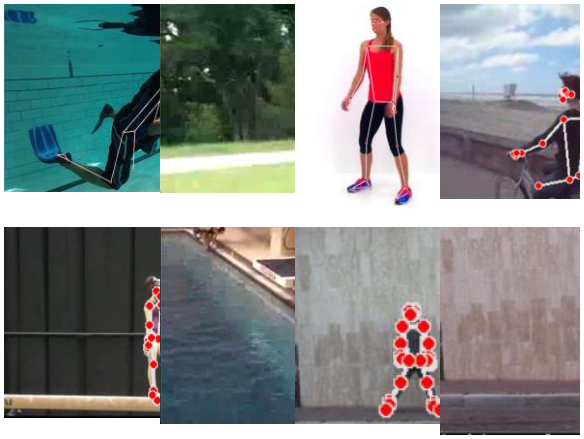


Fig 5a. Keypoint detection result from VDD-C Dataset, **Fig 5b:** Keypoint detection result from Virat Dataset, **Fig 5c:** Keypoint detection result from KTH Dataset **, Fig 5d, 5e:** Keypoint detection result from UCF101 dataset, **Figure 5f, 5g, 5h:** Keypoint detection result from Weizmann dataset.

Fig 6a,6b,6c,6d,6e,6f,6g,6h,6i,6j,6k, and 6l: Keypoint detection results from KTH Dataset during sit-to-stand exercise.

Figures 6a to 6l showcase keypoint detections during sit-to-stand exercises, highlighting the efficacy of our proposed approach in the KTH dataset, specifically addressing challenges such as self-occlusion and partial-occlusion in Human Motion Detection.

The representation of human motion, both with and without occlusion, is illustrated in Figures 7a through 7h. Specifically, Figure 7a displays human motion detection from the VDD-C dataset, Figure 7b depicts detection from the Virat dataset, and Figure 7c exhibits detection from the KTH dataset. Additionally, Figures 7d and 7e showcase human motion detection from the UCF101 dataset, while Figures 7f, 7g, and 7h portray detection from the Weizmann dataset.

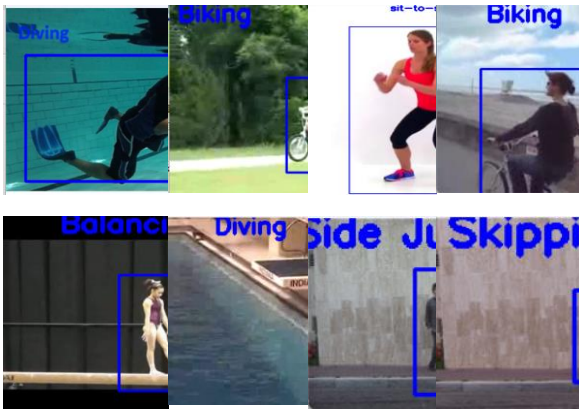


Fig 7a. Human Motion Detection on VDD-C Dataset, **Fig 7b.** Human Motion Detection on Virat Dataset, **Fig 7c.** Human Motion Detection on KTH Dataset **, Fig 7d, 7e.** Human Motion Detection on UCF101 dataset, **Fig 7f, 7g, 7h:** Human Motion Detection on Weizmann dataset.

4.3. Metrics

In the context of assessing human motion detection algorithms, it is imperative to comprehend key

terminologies integral to various evaluation metrics. The following terms, TP (True Positive), FP (False Positive), TN (True Negative), and FN (False Negative), play a pivotal role in quantifying the system's performance in detecting instances of human motion. True Postive (TP)denotes the number of correctly detected instances of human motion.

- True Positive (TP) signifies the instances accurately identified as human motion.
- False Positive (FP) denotes instances where the system erroneously detects human motion in the absence of actual motion.
- True Negative (TN) represents instances where the system correctly identifies the absence of human motion.
- False Negative (FN) refers to instances where the system fails to detect human motion when it is indeed present.

Recall (Sensitivity):

Also known as sensitivity. Table 4 shows the recall of the proposed work along with other methods [37,38,39],. This metric gauges the system's ability to accurately detect human motion and is computed as:

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives} \quad (20)$$

Precision:

Table 4 presents the precision values for the proposed methodology in comparison to other methods [37,38,39]. This metric assess the accuracy of positive detections, precision is calculated as:

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives} \quad (21)$$

F1 Score:

Table 4 presents the F1 score values for the proposed methodology in comparison to other methods [37,38,39]. The F1 score, a harmonic mean of precision and recall, balances the two metrics and is calculated as:

$$\text{Fig Metric} = \frac{2 * Recall * Precision}{Recall + Precision} \quad (22)$$

Accuracy:

Table 4 presents the accuracy rates for the proposed methodology in comparison to other methods [37,38,39]. Accuracy express the ratio of correctly classified instances to the total instances, accuracy is calculated as:

$$Accuracy = \frac{True\ Positives + True\ Negatives}{True\ Positives + True\ Negatives + False\ Positives + False\ Negatives} \quad (23)$$

Percentage of Correct Keypoints:

This metric evaluates the accuracy of a keypoint detection or matching algorithm in locating keypoints relative to ground truth or reference keypoints. Figure 8c shows the Percentage of Correct key of the proposed work along with other methods. Table 3 shows correctly and wrong predicted frames. The formula is:

$$PCK = \frac{Number\ of\ Correct\ Keypoints}{Total\ Number\ of\ Keypoints} * 100 \quad (24)$$

Recognition Rate:

The Recognition rate within the context of human motion detection pertains to the precision or accomplishment level of accurately recognizing and categorizing diverse human motion activities. This metric quantifies the ratio of correctly identified activities in relation to the overall count of activities observed. Figure 8d illustrates the comparison of average recognition rates among various models using different datasets. . The formula utilized to compute the recognition rate in the domain of human motion detection is articulated as follows:

$$Recognition\ Rate = \frac{True\ Positives}{Total\ Number\ of\ Activities} * 100 \quad (25)$$

Average Overlap Rate (AOR):-

In the realm of human motion detection, the term "Average Overlap Rate" characterizes a widely utilized metric for assessing the accuracy of motion detection and tracking algorithms across a sequence of frames. It gauges the alignment between detected human positions and the true human positions within these frames. Figure 8b presents a comparison of the average overlap rates achieved by various methods.

$$AOR = \frac{Number\ of\ True\ Positives}{Total\ Number\ of\ ground\ truth\ positions} * 100 \quad (26)$$

Average Center Error (ACR):-

In the landscape of computer vision, aided by advanced motion tracking algorithms, the achievement of precise Center estimation for human motion has become a tangible goal. Figure 8a illustrates the comparison of average center errors among various models. In this dynamic exploration, the formula employed to calculate the error for a single pair of points (x, y) and (x', y') retains its essence:

$$ACR = \sqrt{(x - x')^2 + (y - y')^2} \quad (27)$$

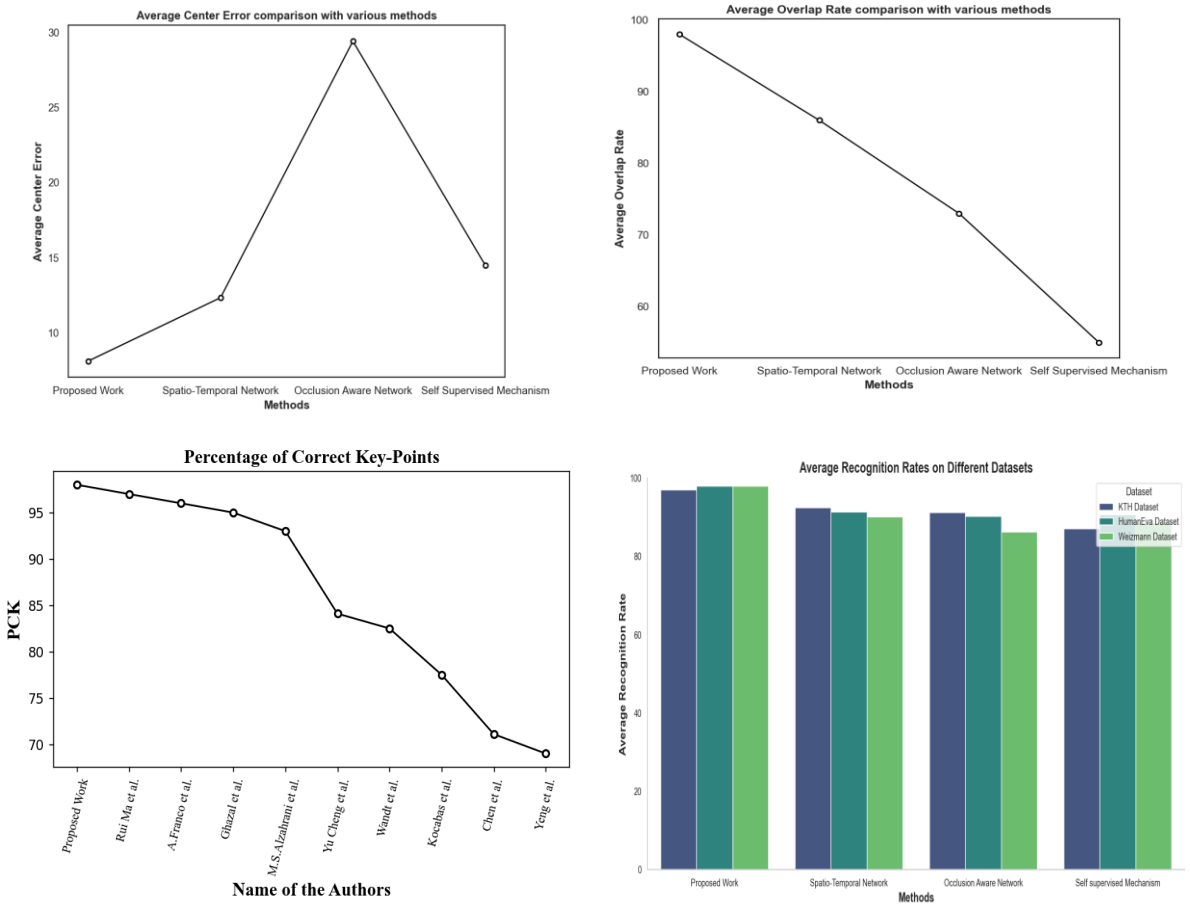


Fig 8a. Comparison of Average Center Error with various models, **Fig 8b.** Comparison of Average Overlap Rate with various methods, **Fig 8c.** Comparison of Percentage of Correct Key-Points with various methods, **Fig 8d.** Comparison of Average Recognition Rate of various models with different datasets

Table 4. Recall and Precision Metrics compared with various Methods

Method	Accuracy	Recall	Precision	Fig Metric
Proposed Work	97%	98%	98%	98%
Long Short Term Memory	96%	97%	97%	97%
Dynamic Spatio Temporal Slice	95%	96%	93%	94%
Hybrid Approach	94%	95%	92%	93%
Deep Keyframe Extraction	93%	95%	92%	93%
Spatio Temporal Slice	67%	69%	65%	67%

5. Conclusion

In conclusion, our research tackles a critical challenge in Human Motion Detection (HMD) – the impact of occlusion on accurate motion identification, particularly focusing on self-occlusion and partial occlusion scenarios. We proposed a sophisticated approach that integrates state-of-the-art technologies, including Mask R-CNN for precise motion segmentation, Recurrent Neural Network (RNN) for object

classification trained on 2D representations of 3D skeletal motion by utilizing the novel hybrid WOA-RDA optimized RNN topology with rapid convergence speed, and Multiple Hypothesis Tracking (MHT) for robust motion tracking during occlusion. Through meticulous experimentation on diverse datasets featuring both occluded and unoccluded scenarios, our approach demonstrated exceptional efficacy in identifying human motion. The results highlight the resilience of our method in handling self-occlusion and partial occlusion, showcasing its suitability for real-world applications such as gesture retrieval, healthcare fall detection, sports analytics, and surveillance. This research significantly contributes to the field by presenting a holistic solution to the challenging problem of occlusion in HMD. The amalgamation of advanced neural network models and tracking algorithms enhances the accuracy and reliability of motion detection, especially in scenarios characterized by occlusion complexities. The outcomes of our experiments substantiate the effectiveness of the proposed approach, providing a foundation for the development of more sophisticated systems capable of understanding human actions in diverse and challenging environments. Looking ahead, future research endeavors could explore optimizations for broader datasets and extend the application of our methodology to real-world settings. The presented approach opens avenues for advancements in HMD, addressing a crucial aspect that significantly influences the accuracy and applicability of motion detection systems. In future research, we plan to employ machine learning algorithms to automatically learn and optimize thresholds or make dynamic adjustments based on training data. This approach aims to enhance adaptability and performance, improving the efficiency and accuracy of the system in real-world scenarios. Sudden changes in lighting conditions can affect the performance of segmentation algorithms, leading to errors in mask extraction. Using adaptive thresholding or background modeling techniques that can adapt to changes in lighting conditions may enhance the robustness of the algorithm.

Author contributions

All authors have reviewed and concurred with the finalized version of the manuscript as published.

Conflicts of interest

The authors declare no conflicts of interest

References

[1] P. Wang, W. Li, P. Ogunbona, J. Wan, and S. Escalera, "RGB-D-based human motion recognition with deep learning: A survey," Computer Vision and Pattern Recognition, 2017.

[2] P. Pareek and A. Thakkar, "A survey on video-based Human Action Recognition: recent updates, datasets, challenges, and applications," Artif. Intell. Rev., vol. 54, no. 3, pp. 2259–2322, 2021.

[3] M. Sima, "Key frame extraction for Human Action Videos in dynamic spatio-temporal slice clustering," J. Phys. Conf. Ser., vol. 2010, no. 1, p. 012076, 2021.

[4] L.-H. Chen and C.-W. Su, "Video caption extraction using spatio-temporal slices," Int. J. Image Graph., vol. 18, no. 02, p. 1850009, 2018.

[5] M. Kocabas, S. Karagoz, and E. Akbas, "Self-supervised learning of 3D human pose using multi-view geometry," 2019.

[6] Y. Cheng, B. Yang, B. Wang, and R. T. Tan, "3D human pose estimation using spatio-temporal networks with explicit occlusion training," 2020.

[7] K. Wang, L. Lin, C. Jiang, C. Qian, and P. Wei, "3D human pose machines with self-supervised learning," IEEE Trans. Pattern Anal. Mach. Intell., pp. 1–1, 2019.

[8] S. Ghazal, U. S. Khan, M. Mubasher Saleem, N. Rashid, and J. Iqbal, "Human activity recognition using 2D skeleton data and supervised machine learning," IET Image Process., vol. 13, no. 13, pp. 2572–2578, 2019.

[9] B. Wandt and B. Rosenhahn, "RepNet: Weakly supervised training of an adversarial reprojection network for 3D human pose estimation," Computer Vision and Pattern Recognition, 2019.

[10] Lv, J. Li, and J. Tian, "Key frame extraction for sports training based on improved deep learning," Sci. Program., vol. 2021, pp. 1–8, 2021.

[11] M. Jian, S. Zhang, L. Wu, S. Zhang, X. Wang, and Y. He, "Deep key frame extraction for sport training," Neurocomputing, vol. 328, pp. 147–156, 2019.

[12] "Human movement detection using recurrent convolutional neural networks," Special Issue-IJITEE, vol. 8, no. 12S, pp. 348–351, 2019.

[13] L. Zhang, "Applying deep learning-based human motion recognition system in sports competition," Front. Neurorobot., vol. 16, 2022.

[14] Kumar and V. Kumar, "Human Motion Detection Based Upon CNN with Pruning and Edge Detection"," ISSN, pp. 6766–6772, 2021.

[15] N. Nair, C. Thomas, and D. B. Jayagopi, "Human activity recognition using temporal convolutional network," in Proceedings of the 5th International Workshop on Sensor-based Activity Recognition and Interaction, 2018.

[16] Y. A. Andrade-Ambriz, S. Ledesma, M.-A. Ibarra-Manzano, M. I. Oros-Flores, and D.-L. Almanza-Ojeda, "Human activity recognition using temporal

convolutional neural network architecture," Expert Syst. Appl., vol. 191, no. 116287, p. 116287, 2022.

[17] M. S. Alzahrani, S. K. Jarraya, H. Ben-Abdallah, and M. S. Ali, "Comprehensive evaluation of skeleton features-based fall detection from Microsoft Kinect v2," Signal Image Video Process., vol. 13, no. 7, pp. 1431–1439, 2019.

[18] A. Franco, A. Magnani, and D. Maio, "A multimodal approach for human activity recognition based on skeleton and RGB data," Pattern Recognit. Lett., vol. 131, pp. 293–299, 2020.

[19] N. Gaud, M. Rathore, and U. Suman, "Human gait analysis and activity recognition: A review," in 2023 IEEE Guwahati Subsection Conference (GCON), 2023.

[20] G. Yang, C. Li, and H. Chen, "Research on deep correlation filter tracking based on channel importance," EURASIP J. Adv. Signal Process., vol. 2022, no. 1, 2022.

[21] Y. Yuan, J. Chu, L. Leng, J. Miao, and B.-G. Kim, "A scale-adaptive object-tracking algorithm with occlusion detection," EURASIP J. Image Video Process., vol. 2020, no. 1, 2020.

[22] H. Chu, K. Wang, and X. Xing, "Target tracking via particle filter and convolutional network," J. Electr. Comput. Eng., vol. 2018, pp. 1–9, 2018.

[23] A. Hayat, F. Morgado-Dias, B. Bhuyan, and R. Tomar, "Human activity recognition for elderly people using machine and Deep Learning approaches," Information (Basel), vol. 13, no. 6, p. 275, 2022.

[24] R. Ma, Z. Zhang, and E. Chen, "Human motion gesture recognition based on computer vision," Complexity, vol. 2021, pp. 1–11, 2021.

[25] F. Ababsa, H. Hadj-Abdelkader, and M. Boui, "3D human pose estimation with a catadioptric sensor in unconstrained environments using an annealed particle filter," Sensors (Basel), vol. 20, no. 23, p. 6985, 2020.

[26] L. C. Chang et al., "An intelligent automatic human detection and tracking system based on weighted resampling particle filtering," Big Data Cogn. Comput., vol. 4, no. 4, p. 27, 2020.

[27] R. Verma and A. K. Verma, "Particle Filter Based Visual Tracking: A Review," International Journal of Advances in Engineering and Management, vol. 3, pp. 3660–3680, 2021.

[28] S. Liu, D. Liu, G. Srivastava, D. Połap, and M. Woźniak, "Overview and methods of correlation filter algorithms in object tracking," Complex Intell. Syst., 2020.

[29] S. S. Moghaddasi and N. Faraji, "A hybrid algorithm based on particle filter and genetic algorithm for target tracking," Expert Syst. Appl., vol. 147, no. 113188, p. 113188, 2020.

[30] X. Guo, A. Hamdulla, and T. Tohti, "Research on target tracking algorithm based on correlation filtering," J. Phys. Conf. Ser., vol. 2024, no. 1, p. 012043, 2021.

[31] M. Ramesh and K. Mahesh, "Sports video classification framework using enhanced threshold based keyframe selection algorithm and customized CNN on UCF101 and Sports1-M dataset," Comput. Intell. Neurosci., vol. 2022, pp. 1–15, 2022.

[32] N. Tasnim and J.-H. Baek, "Deep learning-based human action recognition with key-frames sampling using ranking methods," Appl. Sci. (Basel), vol. 12, no. 9, p. 4165, 2022.

[33] G. A. S. Surek, L. O. Seman, S. F. Stefenon, V. C. Mariani, and L. dos S. Coelho, "Video-based human activity recognition using deep learning approaches," Sensors (Basel), vol. 23, no. 14, p. 6384, 2023.

[34] M. Rapczyński, P. Werner, S. Handrich, and A. Al-Hamadi, "A baseline for cross-database 3D human pose estimation," Sensors (Basel), vol. 21, no. 11, p. 3769, 2021.

[35] S. Chen, Y. Xu, and B. Zou, "Prior-knowledge-based self-attention network for 3D human pose estimation," Expert Syst. Appl., vol. 225, no. 120213, p. 120213, 2023.

[36] K. de Langis, M. Fulton, and J. Sattar, "Video Diver Dataset (VDD-C), 100,000 annotated images of divers underwater." Data Repository for the University of Minnesota (DRUM), 2021.

[37] M. Nithya and K. S. Gautam, "An alternative frame selection approach for motion detection," Int. J. Adv. Sci. Technol., vol. 29, no. 7, pp. 13408–13415, 2020.

[38] Y. Cheng, B. Yang, B. Wang, and R. T. Tan, "3D human pose estimation using spatio-temporal networks with explicit occlusion training," Proc. Conf. AAAI Artif. Intell., vol. 34, no. 07, pp. 10631–10638, 2020.

[39] Y. Cheng, B. Yang, B. Wang, Y. Wending, and R. Tan, "Occlusion-aware networks for 3D human pose estimation in video," in 2019 IEEE/CVF International Conference on Computer Vision (ICCV), 2019.

[40] F. Angelini, Z. Fu, Y. Long, L. Shao, and S. M. Naqvi, "2D pose-based real-time human action recognition with occlusion-handling," IEEE Trans. Multimedia, vol. 22, no. 6, pp. 1433–1446, 2020.

[41] "Pose landmark detection guide," Google for Developers. [Online]. Available: https://developers.google.com/mediapipe/solutions/vision/pose_landmarker. [Accessed: 27-Jan-2024].