information

Article

Deep Learning Approach for Human Action Recognition Using a Time Saliency Map Based on Motion Features Considering Camera Movement and Shot in Video Image Sequences

Abdorreza Alavigharahbagh [1], Vahid Hajihashemi [1], José J. M. Machado [2] and João Manuel R. S. Tavares [2,*]

[1] Faculdade de Engenharia, Universidade do Porto, Rua Dr. Roberto Frias, s/n, 4200-465 Porto, Portugal; up202003516@fe.up.pt (A.A.); up201912327@fe.up.pt (V.H.)
[2] Departamento de Engenharia Mecânica, Faculdade de Engenharia, Universidade do Porto, Rua Dr. Roberto Frias, s/n, 4200-465 Porto, Portugal; jjmm@fe.up.pt
* Correspondence: tavares@fe.up.pt; Tel.: +351-22-041-3472

Abstract: In this article, a hierarchical method for action recognition based on temporal and spatial features is proposed. In current HAR methods, camera movement, sensor movement, sudden scene changes, and scene movement can increase motion feature errors and decrease accuracy. Another important aspect to take into account in a HAR method is the required computational cost. The proposed method provides a preprocessing step to address these challenges. As a preprocessing step, the method uses optical flow to detect camera movements and shots in input video image sequences. In the temporal processing block, the optical flow technique is combined with the absolute value of frame differences to obtain a time saliency map. The detection of shots, cancellation of camera movement, and the building of a time saliency map minimise movement detection errors. The time saliency map is then passed to the spatial processing block to segment the moving persons and/or objects in the scene. Because the search region for spatial processing is limited based on the temporal processing results, the computations in the spatial domain are drastically reduced. In the spatial processing block, the scene foreground is extracted in three steps: silhouette extraction, active contour segmentation, and colour segmentation. Key points are selected at the borders of the segmented foreground. The last used features are the intensity and angle of the optical flow of detected key points. Using key point features for action detection reduces the computational cost of the classification step and the required training time. Finally, the features are submitted to a Recurrent Neural Network (RNN) to recognise the involved action. The proposed method was tested using four well-known action datasets: KTH, Weizmann, HMDB51, and UCF101 datasets and its efficiency was evaluated. Since the proposed approach segments salient objects based on motion, edges, and colour features, it can be added as a preprocessing step to most current HAR systems to improve performance.

Keywords: Human Action Recognition (HAR); deep learning; RNN; time saliency map; camera's movement cancellation

Citation: Alavigharahbagh, A.; Hajihashemi, V.; Machado, J.J.M.; Tavares, J.M.R.S. Deep Learning Approach for Human Action Recognition Using a Time Saliency Map Based on Motion Features Considering Camera Movement and Shot in Video Image Sequences. *Information* **2023**, *14*, 616. https://doi.org/10.3390/info14110616

Academic Editor: Vincenzo Moscato

Received: 27 September 2023
Revised: 7 November 2023
Accepted: 8 November 2023
Published: 15 November 2023

1. Introduction

Actions are physical behaviours through which humans can, for example, interact with their environment or communicate with each other. The development of artificial intelligence, machine vision, and robotics has significantly expanded Human Action Recognition (HAR) systems, which are now used in several applications, such as robotics and security. For instance, smart surveillance systems are employed in security facilities and smart cities to detect inappropriate and dangerous actions. Typically, visual data is the main input of a HAR system, along with other inputs such as sound or infrared data. A HAR system must be trained based on the user's actions, or it should include a pre-trained algorithm for action recognition. Machine learning methods in action recognition are usually categorised

as knowledge- and data-based, with most current methods belonging to the latter group. In terms of processing steps, HAR systems typically include three distinct steps: image preprocessing, feature extraction, and action recognition. Preprocessing is usually designed according to the application, mainly to remove distracting and unnecessary input elements, which increases the system's efficiency and reduces the error and processing time. When a HAR system distinguishes people from the scene and focuses on them in the input video image sequence, it can achieve better and faster results than when it tries to process the entire scene.

HAR also has challenges, such as variable ambient light, noise, and camera and/or scene movement, which increase the complexity of the problem. In this article, a HAR method for urban scenes is proposed that uses camera movement cancellation and motion detection to identify the regions of interest. Active contour and colour segmentation are then used to segment the human and important parts of the scene. Finally, the method uses the motion vector of key points for HAR; therefore, by eliminating useless points and keeping just the key points, the computational cost and memory requirements are minimised, and the system's accuracy is maintained.

2. State of the Art

Most HAR systems emphasise motion features because HAR is a motion-dependent process. On the other hand, in most HAR research, optical flow is the most common technique used to determine the direction and magnitude of the motion. For example, Caetano et al. [1] used a combination of optical flow and Co-occurrence matrices to build features for event detection. Their method considers both the direction and intensity of the motion. The classifier used in their study was a Support Vector Machine (SVM) with a Radial Basis Function (RBF) kernel. Gupta and Balan [2] directly used optical flow for feature extraction. In their study, the output of the Farneback dense optical flow algorithm was converted into colour images and then used to train a Convolutional Neural Network (CNN). Only the frames where an action was present were used in the training and testing phases. Kumar and John [3] used optical flow to build HAR features. They first extracted the foreground of the images, then applied optical flow and used the results to extract features. In this study, the authors only considered frames that contained motion and ignored static image frames, and the used classifier was an SVM with an RBF kernel.

In their HAR system, Rashwan et al. [4,5] used Histograms of Optical flow Co-occurrence (HOF-CO) to build features and CNN and Long Short-Term Memory (LSTM) models as classifiers. Xu et al. [6] used optical flow to extract dense trajectories. Their method used warped optical flow, instead of normal optical flow, to compute a dense trajectory to eliminate camera movement and remove the scene's background. The method also removed redundant features to reduce the memory required for clustering. The authors considered various methods for classification, such as Artificial Neural Networks (ANN), K-nearest neighbours (KNN), and SVM, and the best result was obtained using the SVM classifier.

Liu et al. [7] combined two different systems for action recognition. The first system used a Motion History Image (MHI), whereas the second one used a Kalman filter, and both systems used a VGG-16 CNN for classification. The final result was achieved by combining the results obtained from the two systems. Kumar et al. [8] proposed a method based on a trajectory matrix for feature extraction along with a probabilistic Kalman filter. The extracted features were applied to a Gated Recurrent Unit (GRU) for the final classification. This method used a Gaussian mixture model, graph concepts, and edge features to increase the system's accuracy. Abdelbaky and Aly [9] combined spatial and temporal features with two parallel networks and combined their results. Both parallel networks were Principal Component Analysis Networks (PCANet). In this approach, motion features were extracted using a Short-Time Motion Energy Image (ST-MEI) that calculates the difference between consecutive image frames, and an SVM classifier was used. Guha et al. [10] processed image frames with background subtraction, detected the motion region, and extracted four Histogram of Oriented Gradient (HOG), GLCM, Speeded Up Robust Features (SURF), and

GIST features from the detected region of each image frame. Then, redundant features were removed using a genetic algorithm, and a Multilayer Perceptron (MLP) classifier was used. Dash et al. [11] used a 3D CNN for action recognition. The three-dimensional networks were trained using Scale-Invariant Feature Transform (SIFT) to build the spatial features and the difference of frames to build the motion features, and the key points were identified using edge detection operators. Khan et al. [12] suggested a 26-layer CNN and motion and temporal features for action recognition. The main innovation of their study was the reduction in the number of used features by using a Poisson distribution along with univariate measures (PDaUM).

Jaouedi et al. [13] proposed a HAR method using a Gaussian mixture model to detect human motion against a scene background, and the results are then processed using an RNN. Zheng et al. [14] used video sketch as an auxiliary feature to enhance the efficiency of HAR systems. The authors used a ranking-based method to find effective action sketches and extract the pattern of each action. Ramya and Rajeswari [15] extracted silhouette images from video image sequences using correlation coefficient and image background removal and then determined the type of action based on distance and entropy features and using an ANN as a classifier. Haddad et al. [16] used a combination of Gunner Farneback's dense optical flow (GF-OF), Gaussian mixture model, and information divergence to represent human actions accurately. Kullback–Leibler divergence was used as the classification criterion. Snoun et al. [17] extracted the human skeleton for action detection, and dynamic skeleton, skeleton superposition, and body articulation methods were used to increase the accuracy achieved by a Deep Neural Network (DNN) classifier. Abdelbaky and Aly [18] used three deep CNNs for HAR and combined the results using an SVM. The first network was trained based on the original image; the second one was trained to model horizontal changes, and the last one to model vertical changes. Instead of a single 3D CNN, they used three 2D CNNs. Xia and Ma [19] obtained the motion in the input video image frames using optical flow to determine the type of action. The authors calculated the Joint Action relevance and physical characteristics, including Divergence, Curl, and Gradient as features, and used an SVM classifier.

Carrillo et al. [20] used the Riemannian manifold to obtain action patterns. The covariance matrix and a set of improved dense motion trajectories were used to form the Riemannian manifold. Guo and Wang [21] proposed a deep and sparse spatiotemporal Gabor neural network for action detection. Aghaei et al. [22] used Residual Network (ResNet), Conv-Attention- LSTM, and Bidirectional LSTM (BiLSTM) models for the same purpose, using optical flow to extract the motion. Zebhi et al. [23] used Gait History Image (GHI) and gradient to extract spatiotemporal features and the time-sliced averaged gradient boundary to characterise motion, using a VGG-16 classifier. Wang et al. [24] extracted spatial features using gradient and extracted motion features using optical flow, which were then classified using two different 3D CNNs. Khan et al. [25] used DenseNet201 and InceptionV3 to extract features and a Kurtosis-controlled Weighted KNN as the final classifier. Xu et al. [26] proposed a HAR system based on a scene image and human skeleton, which was robust to optical flow changes. The proposed system was a dual-stream model with sparse sampling combined with a scene image classification scheme. Wu et al. [27] used a descriptor-level Improved Dense Trajectory (IDT) and optical flow for feature extraction. El-Assal et al. [28] used the difference between two consecutive image frames and the rate of change of these differences for action detection.

Boualia and Amara [29] submitted video image frames to a 3D CNN to describe the action involved without preprocessing or feature extraction. Mishra et al. [30] identified regions of interest in the input video image frames, then used MHI and motion energy images to extract features. Finally, they performed action classification using an SVM classifier. Ha et al. [31] used image frame difference for motion detection. In their study, three DNNs were trained using motion and spatial features, and the results were combined in a fusion step. Gharahbagh et al. [32] used a combination of temporal and spatial features, such as frame differences and gradient, for the best frame selection in a HAR

system to improve the training efficiency. Hajihashemi and Pakizeh [33] proposed a HAR system based on the gradient in both temporal and spatial domains. The authors clustered extracted features and formed a bag of video words representing each action. Deshpande and Warhade [34] used the HOG for feature extraction and stacked HOG features for motion detection. Ma et al. [35] used a pre-trained ResNet18 for feature extraction and a CNN with the Class Incremental Learning (CIL) method as a classifier. Shekokar and Kale [36] performed various preprocessing steps on the image frames, such as brightness and contrast correction and Z-score normalisation, and submitted the results to a DNN structure. Sawanglok and Songmuang [37] extracted the global branch, fine-grain branch, Temporal Shift Module (TSM), and TSM fine-grain branch for their HAR system, which included an ANN classifier.

Shi and Jung [38] used a slow-fast network structure for action detection, which allowed the system to respond differently to slow and fast motions. Gao et al. [39] extracted the silhouette region of the image according to the background and then specified the skeleton in different image frames using an LSTM. Wang et al. [40] used a dense optical flow field to detect motion in video image sequences and submitted the extracted features to a DNN. Nasir et al. [41] segmented the human body region after normalising the input video image frames and performing background removal. The authors extracted temporal and spatial features, such as the 3D Cartesian plane, Joints MOCAP, and n-point trajectory from the segmented human region. Sowmyayani and Rani [42] proposed a DNN for action detection in video retrieval systems. The image frame difference was used to detect motion and remove static frames. Singh et al. [43] used optical flow and HOG features to specify motion regions. Finally, the extracted features were used to create a sparse-coded composite descriptor to model the action. Mithsara [44] used a Yolo v5 network to segment humans and then used skeleton extraction to determine the action type.

Megalingam et al. [45] used SIFT to extract spatial features and optical flow to extract motion features for their HAR system. In another work, Nair and Megalingam [46] reviewed HAR methods. Bayoudh et al. [47] proposed a method based on a hybrid 2D/3D CNN, an LSTM network, and a visual attention mechanism for HAR. Liang et al. [48] used the DCT of image frames as input to a CNN. Khater et al. [49] suggested a residual inception ConvLSTM network for feature extraction in a HAR system. Motion information can also be extracted from non-video data such as skeleton data or 3D (RGB-D) videos. Using different modalities increases the accuracy of human activity recognition (HAR) systems. Momin et al. [50], Sun et al. [51], and Wu and Du [52] investigated HAR systems based on different data modalities. Ahn et al. [53], Vaitesswar and Yeo [54], and Lee et al. [55] used skeleton data for HAR. Wu et al. [56], Radulescu et al. [57], Yan et al. [58], and Liao et al. [59] are among other studies that used depth as a data method in combination with video for HAR. The main classifier in all the aforementioned methods is deep learning. Most multimodal methods use graph-based convolutional neural networks (GCNN) or graph neural networks (GNN) because they provide a deep learning method for irregular domains in the machine learning community.

Based on our review, HAR methods based on deep learning consist of two main steps: feature extraction before deep neural networks and feature extraction with DNN. In both steps, spatial and motion features have been used. However, camera movement, sensor movement, and scene movement can increase motion feature errors and decrease the HAR systems' accuracy. Another challenge in video-based HAR is the unclear location of the person in the scene. Some studies have used background removal and image extraction in video data as preprocessing steps to solve the above challenges. These methods generally locate a person, isolate the person from the scene's background, and identify the motion region.

The proposed method focuses on preprocessing to compensate for camera movement, sensor movement, and scene movement before DNN. Some steps of the proposed method are based on modifications of previous studies. These parts are shot detection to detect sudden changes in the scene during action detection, camera movement cancellation to separate the

motion of the objects of interest from the movement of the camera, optical flow computation, silhouette image extraction, and active contour and colour-based image segmentation.

Various methods have been proposed for detecting scene changes in video image sequences. For example, Bi et al. [60] used coherent spatio-temporal patterns, Lu and Shi [61] performed candidate segment selection and Singular Value Decomposition (SVD), Mishra [62] employed a complex dual-tree wavelet with Walsh–Hadamard transform. Rashmi and Nagendraswamy [63] used a block-based cumulative approach. However, all of these methods are slow when used for action detection. For camera movement cancellation, Hu et al. [64] used spatial feature matching within image frames to distinguish object motion from camera movement. After this step, the scene background and foreground were separated, and the moving objects were extracted. Some approaches for removing camera movement, which are also employed in creating panoramic images, include image frame registration and stitching methods [65]. Moore et al. [65] proposed a matching approach that uses spatial patterns and a Principal Component Analysis (PCA)-based cost function to remove camera movement. Zhang et al. [66] proposed an optical flow method to separate camera movement from object motion in a scene. They used the difference between the angles of the flow vector in static and moving objects. Once the camera movement is determined, their method corrects the optical flow. The aforementioned camera movement cancellation methods were proposed for other applications and must be modified for a HAR system.

The extraction of silhouette images has been used as one of the useful tools in action detection. For example, Ahammed et al. [67] used silhouette images to detect human walking. They extracted silhouettes from video foregrounds. Lam et al. [68] separated background from foreground in video image sequences by filtering and extracting silhouettes. Jawed et al. [69] used silhouette images to develop a human gait recognition system. In this study, silhouette images are obtained using background and frame difference, using statistical features such as variance and covariance of scene brightness to extract the background. Maity et al. [70] proposed a HAR system using silhouette images, built by making a background history from consecutive image frames and subtracting it from the image frames. Vishwakarma and Dhiman [71] also used silhouette images for action detection based on Gaussian filters, video motion energy, and image texture for the background and foreground segmentation. The proposed method includes active contour and colour-based segmentation. Active contour methods are generally divided into two categories [72]: parametric and geometric active models. In parametric models, the edge of an object is extracted based on its shape and energy features of the image. The traditional models of this method include the original snake model [73], balloon active contour [74], gradient vector flow [75], and vector field convolution [76]. Parametric models offer high processing speed but are sensitive to the initial point and object geometry. On the other hand, geometric models have lower sensitivity to the initial point and object geometry but have higher computational costs. Examples of geometric models include the Mumford–Shah model [77], geodesic active contour [78], the Chan–Vese model [79], and LBF model [80]. Some studies have also employed image co-segmentation methods for human segmentation that require training [81,82]. Merdassi et al. [83] reviewed several image co-segmentation methods and highlighted their strengths and weaknesses.

In the field of colour-based image segmentation, Anitha et al. [84] adjusted colour thresholds of different RGB channels to maximise Otsu's criterion and Kapur's entropy. Jing et al. [85] used a Fully Convolutional Neural Network (FCN) for image segmentation. They selected the architecture of this DNN by combining different models [86–90]. Kabilan et al. [91] used the fastmap method for colour-based image segmentation, where the results of fastmap and PCA were combined and optimised in an iterative process. Their method incorporated the Sobel gradient, median filter, and Gaussian smoothing. Abualigah et al. [92] analyzed different image segmentation criteria using heuristic methods and compared the results. In addition to Otsu's class variance criterion and Kapur's entropy criterion, the authors also studied fuzzy entropy, Tsallis entropy, and Renyi's entropy criteria [92]. Sathya et al. [93] used three criteria

for image segmentation: Kapur, Otsu, and Minimum Cross Entropy, and combined the results of these methods to increase the segmentation accuracy.

None of the above studies have been used in HAR systems and should be modified to be used. From the above, the innovations of the proposed method are the following:

- The cancellation of the camera movement based on optical flow;
- The use of optical flow and image frame difference to determine motion regions and processing only the selected regions in the later steps, which reduces about 90% of the required computation;
- The use of active contour and colour-based image segmentation in the motion regions for more accurate human segmentation from the scene background;
- The use of previously computed optical flow vectors at selected key points as final features to save computation time.

3. Materials And Methods

In this study, a new hierarchical method for HAR is proposed. This section describes different parts of the proposed method, which is depicted in Figure 1. Some of the shown steps may be discarded, i.e., not considered, depending on the input video image sequences. For example, the first step, which detects shots in the video and identifies scene changes, can be discarded in videos with static scenes. After this step, the temporal processing block detects moving regions in the input video image frame. This block includes camera movement cancellation, optical flow, and image frame difference. Finding regions of movement is significant to reduce the computation required in the following steps. The spatial processing block improves the segmentation of humans and other moving objects based on the results of temporal features. The final features are extracted from key points of the segmented region. The final step is the classification block based on an LSTM.

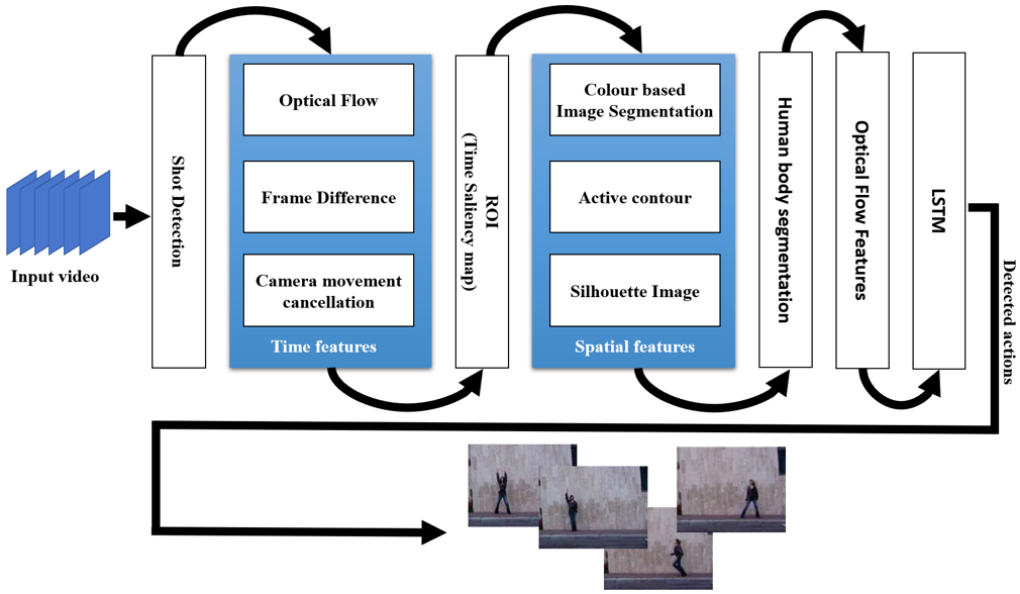

Figure 1. Block diagram of the proposed method.

3.1. Shot Detection

In some videos, sudden changes in the scene during action detection can cause discontinuities and motion detection errors. A HAR system typically relies on segmenting moving regions in the input image frames, which means that scene changes must be taken into account during the feature extraction to avoid errors. The proposed method aims to detect shots and address these errors during the feature extraction step.

In a HAR system, finding a video shot is typically not very complex and can be achieved using simpler methods. In this study, image optical flow and frame difference are

used to detect scene changes. When the scene suddenly changes, the image optical flow and frame difference increase suddenly. Between the optical flow and frame difference, the optical flow is more suitable for shot detection because it is more stable than the frame difference. When a shot occurs, the optical flow values in several consecutive frames are higher than normal. When a shot is detected, the proposed method excludes some frames before and after the shot from the feature extraction and action detection processes.

3.2. Camera Movement Cancellation

The objective of the temporal processing block is to separate moving objects from the static components of the input video image sequences. There are two types of motions in videos with actions: the motion of objects and the movement of the camera. Usually, in this domain, the motion of objects is associated with action, and camera movement is uninteresting. Consequently, camera movement should be removed to ensure that the feature extraction is not affected; therefore, the temporal block consists of camera movement cancellation, optical flow, and frame difference sub-blocks. Regarding the priority, first, camera movement should be cancelled, and then, optical flow and frame difference should be calculated.

In our method, static and moving objects were separated by comparing the flow angles. Static objects, as the main part of an image frame, have similar flow angles due to camera movement, and moving objects have different flow angles. A cluster containing many similar or denser components is considered to contain camera movement, while the remaining clusters represent object motion. The camera movement in the horizontal and vertical directions is determined according to the angle and intensity of the optical flow in the centre of the denser cluster. Subsequently, all flow vectors are adjusted according to the camera movement. In this way, the static components of the scene that previously had a motion vector due to the camera movement become static. Moreover, it is not necessary to recalculate the optical flow, and the flow results can be corrected by subtracting the centre of the denser cluster from all flows. In the next step, the frame difference is calculated based on the corrected frames. After the camera movement cancellation, the image optical flow and frame difference provide a more accurate representation of the object's motion. In cases where the dataset is acquired with a static camera, this step is unnecessary and can be discarded.

3.3. Optical Flow

The proposed method uses two different approaches for motion detection to enhance its efficiency and reduce errors. The first approach is optical flow. One limitation of this method is its inability to distinguish between the movements caused by the camera and the actual motion in the scene. This limitation has been addressed through the camera movement cancellation step. In addition, the optical flow method has a relatively high computational cost because of its spatial and temporal derivatives. Among the various current optical flow algorithms, the Lucas-Kanade and Horn-Shunck algorithms have been mainly used in HAR systems. The difference between these two methods lies in how they compute the derivative and some initial assumptions. Both methods use a similar approach to estimate the motion vector. In a two-dimensional (2D) grey video image sequence, it is assumed that the motion in x, y directions is δx, δy at time δt, and the image pixel values of the first image frame are $E(x, y, t)$, and of the second is $E(x + \delta x, y + \delta y, t + \delta t)$, thus:

$$E(x + \delta x, y + \delta y, t + \delta t) = E(x, y, t) + \frac{\partial E}{\partial x}\delta x + \frac{\partial E}{\partial y}\delta y + \frac{\partial E}{\partial t}\delta t + \varepsilon, \quad (1)$$

where ε represents the error that occurs due to the nonlinear form of the motion. The motion in a short time between two image frames can be assumed to be linear:

$$\varepsilon \approx 0. \quad (2)$$

The brightness of objects usually does not change significantly between two consecutive image frames, so each object has approximately the same brightness; hence:

$$E(x + \delta x, y + \delta y, t + \delta t) = E(x, y, t). \tag{3}$$

According to Equations (2) and (3), one can obtain:

$$\frac{\partial E}{\partial x}\delta x + \frac{\partial E}{\partial y}\delta y + \frac{\partial E}{\partial t}\delta t = 0. \tag{4}$$

After dividing by ∂t and simplifying the previous equation, one obtains:

$$\begin{array}{l} \frac{\partial E}{\partial x}\frac{\delta x}{\delta t} + \frac{\partial E}{\partial y}\frac{\delta y}{\delta t} = -\frac{\partial E}{\partial t}, \\ \frac{\partial E}{\partial x}V_x + \frac{\partial E}{\partial y}V_y = -\frac{\partial E}{\partial t}, \end{array} \tag{5}$$

which can be rewritten in the form of:

$$\begin{array}{l} \nabla_y E.V_x + \nabla_y E.V_y = -\frac{\partial E}{\partial t}, \\ \nabla E.\begin{bmatrix} V_x \\ V_y \end{bmatrix} = -\frac{\partial E}{\partial t}, \end{array} \tag{6}$$

where V_x and V_y are the velocities of the object in the horizontal and vertical directions, which are unknown, and ∇ is the gradient of motion in the spatial domain. Because Equation (6) involves two unknowns, some constraints must be introduced to solve it. Assuming that the motion of the objects between two image frames is smooth in all directions, one obtains:

$$\frac{\partial V_x}{\partial x}, \frac{\partial V_y}{\partial y}, \frac{\partial V_x}{\partial y}, \frac{\partial V_y}{\partial x} \rightarrow 0, \tag{7}$$

or

$$\begin{array}{l} \left(\frac{\partial V_x}{\partial x}\right)^2 + \left(\frac{\partial V_x}{\partial y}\right)^2 \rightarrow 0, \\ \left(\frac{\partial V_y}{\partial y}\right)^2 + \left(\frac{\partial V_y}{\partial x}\right)^2 \rightarrow 0. \end{array} \tag{8}$$

Solution methods such as Farneback, Horn–Schunck, Lucas–Kanade, and Lucas–Kanade derivative of Gaussian have been suggested to determine Vx and Vy while minimising both terms in Equation (8) and satisfying Equation (6). The Lucas–Kanade algorithm, implemented in parallel processing mode [94], is used for optical flow calculation in the present study.

3.4. Time Analysis Block

In summary, in the first step of the temporal block, camera movement is removed using optical flow data obtained in the shot selection step. The image frame difference is calculated after the camera movement cancellation. The motion vector obtained using optical flow is adjusted according to the camera movement vector and then used as a saliency map in combination with the result of the frame difference. The relationship used in the combination of optical flow and frame difference is:

$$Saliency\ Map = (\text{opticalflow})^{\alpha}(Frame\ \ difference)^{\beta}, \tag{9}$$

where values of α and β are positive numbers greater than 1 (one) and, here, were calculated in a simulation step (see Section 5.1). Therefore, because the optical flow achieves a better motion pattern than the frame difference, it has a greater influence than the frame difference in Equation (9). The spatial processing block has three stages: silhouette image extraction, active contour segmentation, and colour-based image segmentation.

Because moving parts usually occupy a small region in the image frame, spatial feature extraction can be performed quickly with relatively few calculations.

3.5. Silhouette Image Extraction

In the current study, silhouette extraction is performed by assuming that the background contains motionless regions or is far from the motion region. First, the gradient of the image frame is calculated using the Sobel operator. It is assumed that the silhouette region is rectangular and contains motion, as well as parts that overlap with the motion region. This includes the entire body, i.e., incorporating both moving and non-moving parts, to differentiate between human actions accurately. The maximum dimension of the rectangular window for the silhouette image is assumed to be 1.5 times the salient region specified in the temporal processing block. As the motion region has already been identified in the preceding steps, the silhouette extraction is more accurate and faster than the general methods.

3.6. Active Contour Segmentation

Silhouette extraction approximates the boundaries of an object; therefore, accurate segmentation methods must then be used to detect the entire object of interest. Active contour is one of the powerful image segmentation methods based on spatial features. Because pre-trained models are not acceptable and geometric methods are more efficient, the proposed solution includes a geometric method. Among the geometric methods, [79,80] showed a lower error and good performance; thus, the model suggested in [80] is used in the proposed solution.

3.7. Colour-Based Image Segmentation

The objective of this block is to extract temporal features based on colour information, which, if the location of the person or object is known, can accurately distinguish the person or object from the scene background. Here, the Sathya et al. [93] method based only on Minimum Cross Entropy criteria is used. In the proposed method, colour-based segmentation is just applied to the silhouette region, and the final segmentation is limited to the silhouette region dilated 1.5 times. The outcome of this step, which is an accurate segmentation of humans and moving objects, is then transferred to the final feature extraction step.

3.8. Final Feature Extraction

After segmenting the motion region, which includes humans and other foreground objects, appropriate features must be extracted. The final feature is a vector, including x and y as the location and motion vector of some selected points of a segmented human. The main challenge in this step is that despite all the steps taken to extract spatial and temporal features, there are still segmentation errors. Moreover, the angle between the person and the camera plays a crucial role in feature extraction. In a first attempt, a 15-point model was employed for feature extraction, which can model simple motion if the view is assumed from the front, Figure 2.

In the 15-point model, the front view is assumed to be a complete view. However, a challenge arises when a body part is missing during segmentation because it leads to incomplete feature extraction. To address this issue, several assumptions were made:

- A 15-point model was used in the dataset, including the whole body view. In cases where any of these 15 points are missing, the corresponding position is considered empty.
- In cases where a full body view is unavailable, the feature vector is created by selecting 65 points along the extracted contour, with equal distances between them, as key points.
- In the full-body model, the extracted feature includes the location of the points relative to point 9 as an approximation to the body centroid and the intensity and direction of the motion vector in the key points. If the 15-point model is not available, the point with the centre of the segments is considered the coordinate origin.
- In general, 65 points are selected, each with four features, including x and y locations and motion vector.

It should be noted that the input of the DNN classifier is a vector with a dimension of $65 \times 4 \times$ Nf when full body view is not available, and $15 \times 4 \times$ Nf when full body view is available, where Nf is the number of image frames. When an action is repeated in a video image sequence, these time intervals are used as two samples of the same action in the training step. In this step, the calculated optical flow at selected points is used as features, and there is no need to extract new features.

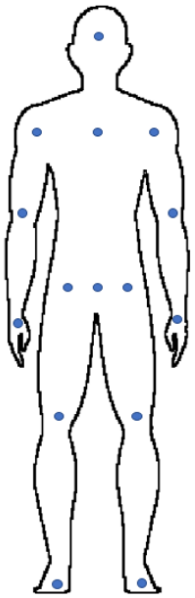

Figure 2. The 15-point human body model used in the proposed method.

3.9. Long Short-Term Memory

In the last step, an RNN is used as a classifier. Among DNNs, RNNs are the most suitable structures for processing time series data, such as actions, where each sample depends on previous samples. In a basic RNN, due to its simple structure, memory limitations in the hidden layers, and weight update algorithm that uses gradient-based methods, the system cannot be properly trained when the pattern in the data, such as action data, spans a long period. In gradient-based updating, the influence of the data in the more distant samples decreases compared to the closer samples until it eventually becomes 0 (zero). LSTM is a modified RNN used for learning long-term patterns. Figure 3 shows the structure of an LSTM.

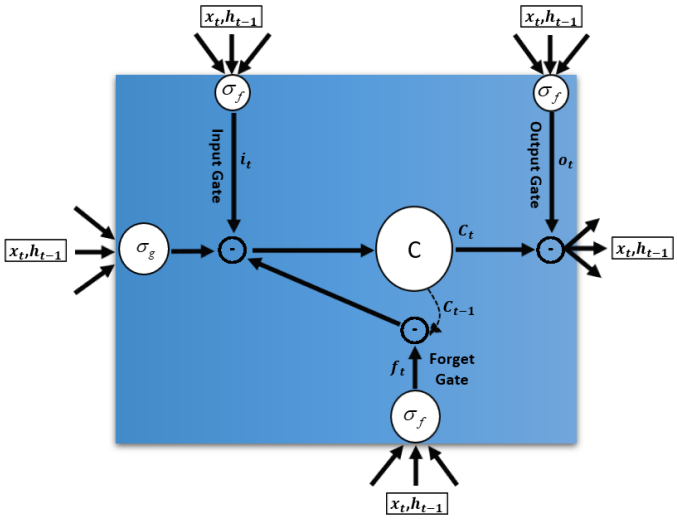

Figure 3. Traditional LSTM structure.

The LSTM equations are as follows:

$$\begin{aligned} f_t &= \sigma_f\left(W_{xf}x_t + W_{hf}h_{t-1} + b_f\right), \\ g_t &= \sigma_g(W_{xc}x_t + W_{hc}h_{t-1} + b_c), \\ i_t &= \sigma_f(W_{xi}x_t + W_{hi}h_{t-1} + b_i), \\ c_t &= f_t \odot c_{t-1} \odot g_t \odot i_t, \\ o_t &= \sigma_f(W_{xo}x_t + W_{ho}h_{t-1} + b_o), \\ h_t &= o_t \odot c_t, \end{aligned} \tag{10}$$

where x_t is the input, h_{t-1} the previous states, c_t the previous states within the forget gate, W the weight and b the bias of each inner block of the LSTM, σ is the activation function, Hedmard multiplication ($\odot$) corresponds to pointwise multiplication, and the final output is h_t. The forget gate determines which inputs and previous states affect the output and which should be discarded. This enables the LSTM to learn long-term patterns.

4. Experimental Setting

4.1. Datasets

Several action datasets with different conditions are available for evaluating HAR methods. Five datasets were used in this study: BVSD, KTH, Weizmann, HMDB51, and UCF101 datasets, Figure 4. Details of these datasets are given in the following sections.

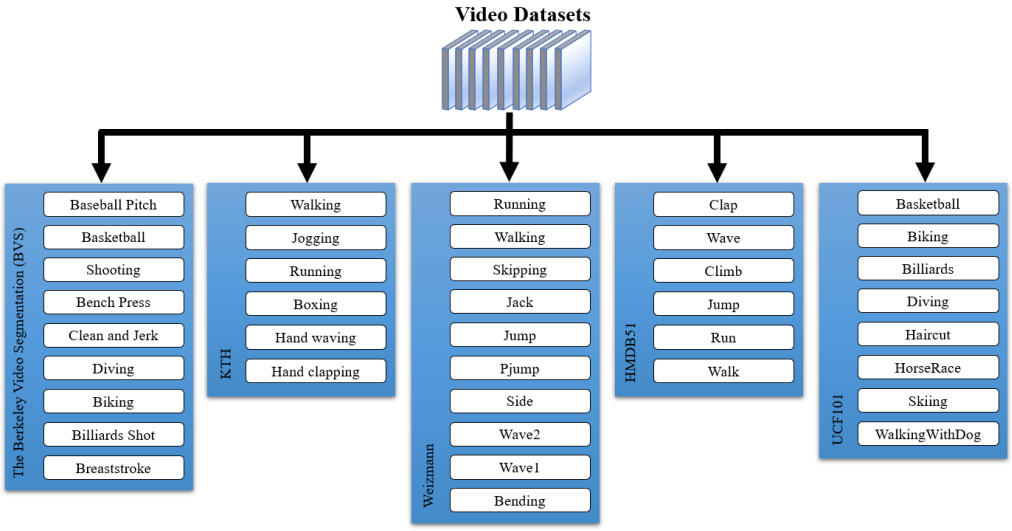

Figure 4. Datasets used in the evaluation of the proposed method.

4.1.1. Berkeley Video Segmentation Dataset

The Berkeley Video Segmentation Dataset (BVSD) was the first dataset used to determine the powers in the saliency map function (Equation (9)). This dataset consists of high-resolution video image frames labelled for evaluating video segmentation methods [95], and was later extended to evaluate motion detection methods [96].

Although this dataset contains some errors and incorrect segmentations in several frames, it is a useful benchmark for evaluating methods that separate moving objects from a scene. Consequently, it has been widely used in many studies [97–99].

4.1.2. KTH Dataset

The KTH Dataset contains 600 video image sequences of six different actions [100]. There are a total of 2391 frames in the videos, all acquired in black and white at a resolution of 120 × 160, and only one person is seen in each video. The actions included in this dataset are walking, jogging, running, boxing, hand waving, and hand clapping. The videos were acquired from a fixed distance, and the person's body is fully visible in them, and the angle

and direction of the scene are fixed. However, in some cases, the human scale changes. The clothing of the people in the scene also changes. Because the camera angle and scene are fixed, and there are no sudden changes in the scene or background, and therefore no shot detection or camera movement cancellation is required for this dataset. It has been considered a good benchmark in many studies [101–104]. The low resolution of the videos also reduces the required computations.

4.1.3. Weizmann Dataset

The Weizmann dataset [105] was acquired in a static scene with simple backgrounds as a HAR dataset for outdoor environments. There is only one person in each action, and there are 10 actions involved, including Running, Walking, Skipping, Jack, Jump, Pjump, Side, Wave2, Wave1, and Bending, so it includes more actions than the KTH dataset. The number of videos in this dataset is 90, and the resolution is 144×180, which is higher than that of the KTH dataset. Since there are no sudden scene changes or camera movement in this dataset, shot detection and camera movement cancellation are not required. The subject is fully positioned in the video, and therefore, the proposed 15-point model can be used for feature extraction. This dataset has been extensively used to evaluate HAR methods [106–108]. In this study, the Weizmann dataset was used as the second dataset to calculate the powers of the saliency map function (Equation (9)). Because the saliency map function contained only two variables, it was unnecessary to use optimisation methods to calculate the optimal values, and the selection was performed simply by assigning two different intervals for the powers and checking all the possible values.

4.1.4. HMDB51 Dataset

The HMDB51 dataset [109] was acquired under real conditions and contains all the challenging factors a HAR system may encounter, including camera movement, complete absence of humans in many videos, i.e., considerably long periods without action or people, sudden scene changes, different video qualities, and fluctuating lighting conditions. According to these challenging factors, all parts of the proposed algorithm are required to process this dataset, which contains 51 different actions tagged in about 7000 different clips. The videos have varying lengths, with a total duration of 101 min. The actions of this dataset are classified into the following five groups :

- General facial actions;
- Facial actions with object manipulation;
- General body movements;
- Body movements with object interaction;
- Body movements with human interaction.

Additionally, it should be noted that each category includes different actions.

4.1.5. UCF101 Dataset

The UCF101 dataset, similar to the HMDB51 dataset, was acquired under real-world conditions and includes camera movement, scene changes, and lighting changes [110].

This dataset contains 101 different actions, such as Baseball Pitch, Basketball, Shooting, and Bench Press. The videos in this dataset are all in colour, with varying duration, and higher resolution than the Weizmann and KTH datasets. Table 1 presents details on this dataset.

Table 1. Characteristics of the UCF101 dataset.

Actions	101
Clips	13,320
Groups per Action	25
Clips per Group	4–7

Table 1. *Cont.*

Mean Clip Length	7.21 s
Total Duration	1600 min
Min Clip Length	1.06 s
Max Clip Length	71.04 s
Image Frame Rate	25 fps
Image Resolution	320 × 240
Audio	Yes (51 actions)

4.2. Evaluation Metrics

In a HAR system, if the system is binary, i.e., it only distinguishes between the presence or absence of an action, the following criteria are used to evaluate the system:

- True positive: Action exists and is correctly detected;
- True negative: Absence of action is correctly detected;
- False positive: Action does not exist, but the system mistakenly indicates its existence;
- False negative: Action is present but incorrectly detected as absent.

In multi-class mode, i.e., where the number of actions is more than two, the confusion matrix is used to evaluate the system. In this matrix, each row or column represents a class, the main diagonal elements show the number of correctly recognised samples of that class, and the other elements indicate the following:

- If the element is interpreted based on the action associated with its row, it indicates the number of samples of row classes that were mistakenly assigned to other classes;
- If the element is interpreted based on the action associated with its column, it indicates the number of samples of other classes that were mistakenly assigned to the column class.

The sum of the main diagonal elements divided by the total number of samples indicates the system's accuracy. Confusion matrices can be used to evaluate the performance of HAR systems in each class separately. The percentage reported at the end of each row of the confusion matrix is the true positive rate for the row class. The percentage reported at the end of each column of the confusion matrix is the precision of the column class. Precision is the number of true positives divided by the total number of elements assigned to that class by the system.

5. Experimental Results And Discussion

A series of simulations were performed to determine the parameters of the different parts of the proposed method and evaluate its competence. In this section, details of the results are presented and discussed.

5.1. Time Features

Firstly, the values of α and β of Equation (9) should be calculated. An optimisation approach using the GA method was used to determine these values. The α and β were chosen as limited values between 1 (one) and 10, and the aim was to obtain the minimum value of these variables to find the moving objects between frames. All Weizmann and BVSD samples are input for the GA cost function. The default values of the MATLAB optimisation toolbox were used to implement the GA. Finally, the values of α and β were obtained as equal to 1.32 and 1, respectively. The results of the optical flow and frame difference blocks before and after cancelling the camera movement are shown in Figure 5. In the selected frames, the person moves slightly, and the camera moves simultaneously.

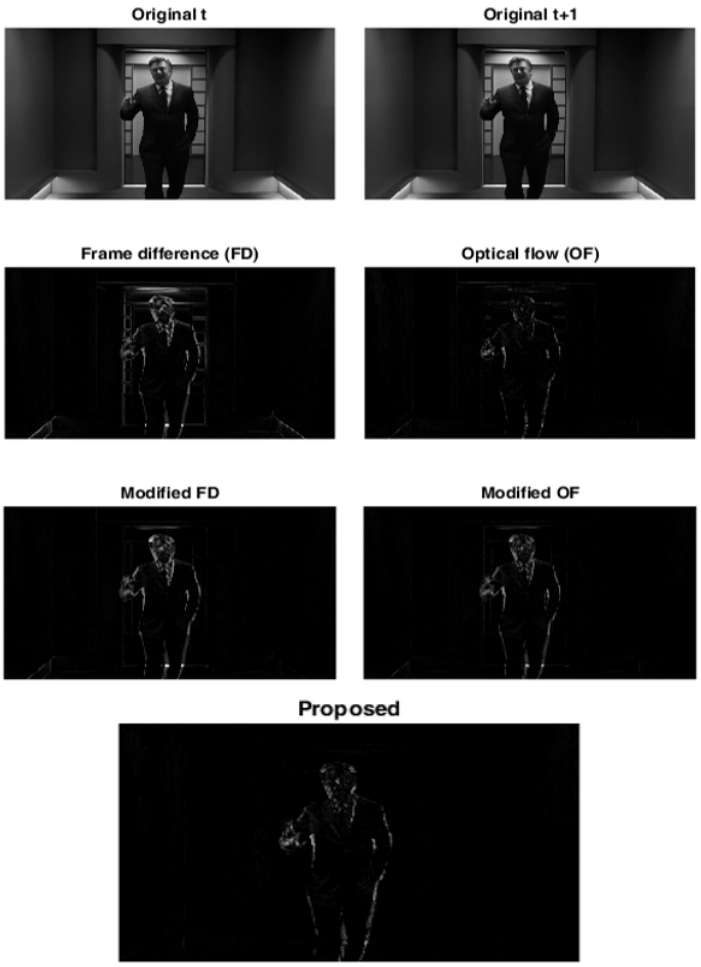

Figure 5. Motion detection before and after the preprocessing and final time saliency map.

As can be seen, the optical flow and frame difference without the camera movement cancellation have relatively high errors. In these two cases, parts of the door and borders are erroneously selected as moving objects. After the cancellation, the optical flow and frame difference have much better results; however, in both cases, some noise can be seen, and the static objects are very faintly visible. The result of the optical flow is better than that of the frame difference with and without the camera movement cancellation. After combining the two results using the proposed function, the border around the image disappeared completely, and the person was clearly separated from the background. The reduction of noise and the absence of borders in the temporal processing step are very important because, in this step, the region of interest was extracted and transferred to the next steps. In Figure 6, there is another example where the camera moves simultaneously with the person under study. Before the camera movement cancellation, the error was significantly high, as can be perceived from the results.

After the camera movement cancellation, the frame difference and optical flow results were much better, although the left margin and some discontinuities are still visible. In the lower part of Figure 6, it can be observed that after applying the proposed approach, the errors disappeared completely, and the part containing the person was completely separated from the rest of the scene. The resulting output was then applied to the spatial processing block as a time saliency map to segment important parts of the frames using spatial features. Figure 7 shows the active contour segmentation result for a specified silhouette region. However, due to the presence of static object edges within the silhouette region that overlaps with the boundaries of the moving object, the active contour did not

yield optimal results. The active contour performs better in cases where only the edges of the moving object are inside the silhouette region without overlapping with other objects.

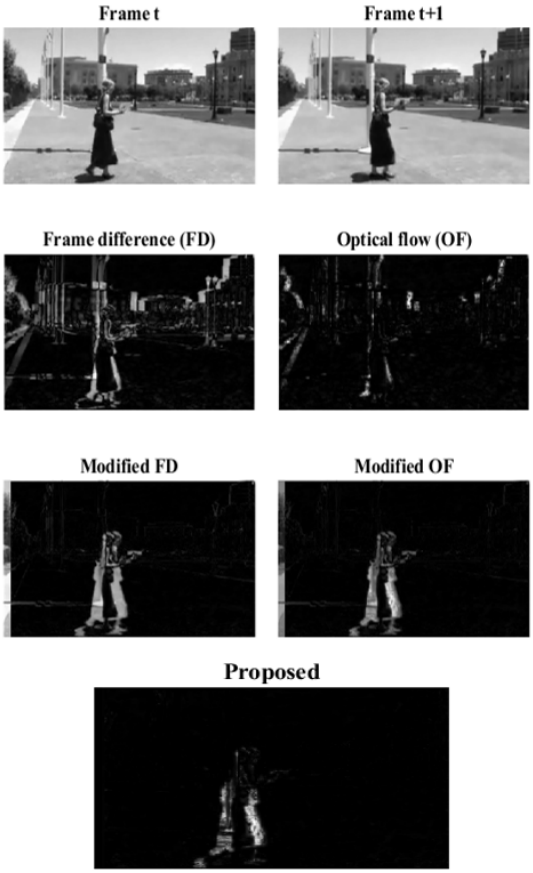

Figure 6. Motion segmentation before and after the camera movement cancellation.

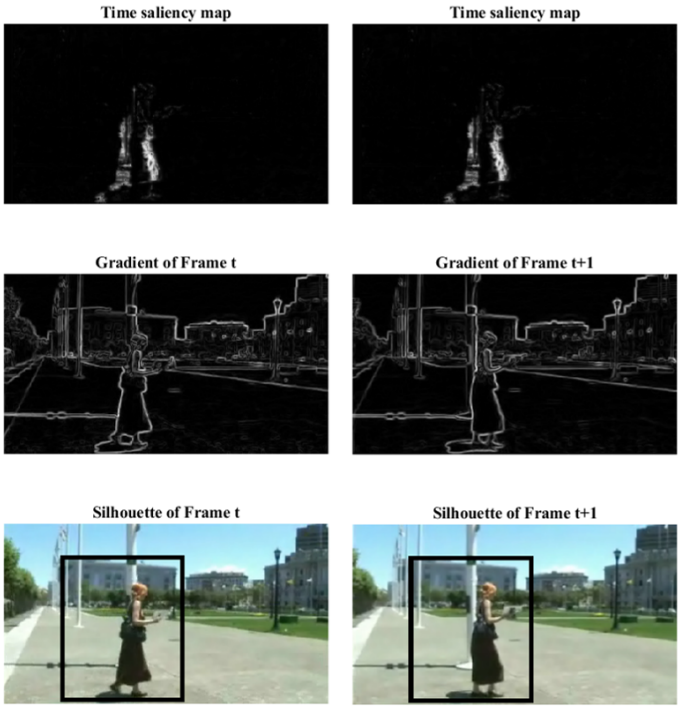

Figure 7. Silhouette image extraction using time saliency map and spatial features.

Figure 8 shows the result of colour-based image segmentation within a silhouette region in green. In the proposed method, the colours of the outer part of the silhouette region and the result of the active contour are chosen as background colours, and the colours of the inner part of the moving region are chosen as the foreground. Since the moving region was accurately separated, the moving person was accurately segmented based on its colour difference from the background. The most important parameter in colour-based image segmentation is the colour difference between moving objects and other objects in the silhouette region. The resulting segmentation is combined with the active contour segmentation method by intersection and then used for feature extraction. Figure 9 shows the final segmented region, which confirms the proposed method provides a very good approximation of the moving person.

Figure 8. Result of the colour image-based segmentation.

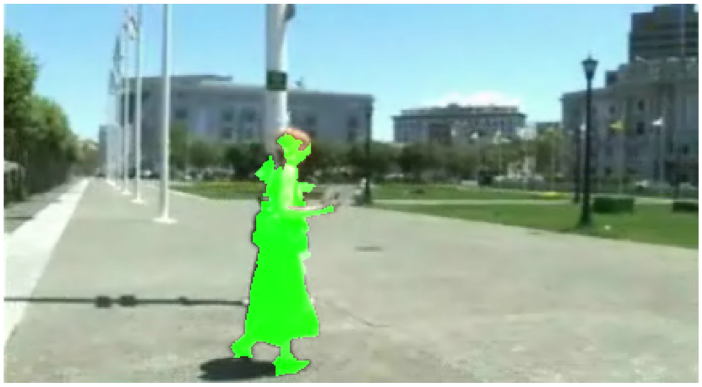

Figure 9. Final segmented region (in green).

5.2. Effect of the Different Steps of the Proposed Method

In this section, different parts of the proposed method are added to a HAR system, and their impact on the system's efficiency is evaluated. KTH, Weizmann, and HMB51 datasets were used to evaluate the proposed method. According to the preprocessing steps' structure, the training and testing data selection was done only in the DNN training (Section 5.2), and the rest of the steps were done using the whole data. In the DNN training, for each dataset, only its data was used for training and testing. The selected percentages for training and testing at this step were 70 and 30, respectively.

In the first evaluation, the proposed time saliency map (Equation (9)) was compared with the conventional optical flow in KTH and Weizmann datasets. The two implementations were performed under similar conditions regarding the training and testing data, classifier structure, and DNN parameters. The only difference between the two implementations was the time saliency map, with one being based on the optical flow and the other on the proposed method. Tables 2–5 show the results obtained for KTH and Weizmann datasets. Considering that all comparison conditions were the same, it can be concluded that using the proposed temporal saliency function, the performance was enhanced even when there was no movement in the scene.

Table 2. Confusion matrix obtained for the HAR system using optical flow on the KTH dataset.

	Walking	Jogging	Running	Boxing	Hand Waving	Hand Clapping	Accuracy (%)
Walking	27	3	0	0	0	0	90
Jogging	1	27	2	0	0	0	90
Running	0	3	27	0	0	0	90
Boxing	0	0	0	28	0	2	93
Hand waving	0	0	0	1	27	2	90
Hand clapping	0	0	0	0	0	29	100
Accuracy (%)	96	82	93	97	100	88	

Table 3. Confusion matrix obtained for the HAR system using the proposed method on the KTH dataset.

	Walking	Jogging	Running	Boxing	Hand Waving	Hand Clapping	Accuracy (%)
Walking	30	0	0	0	0	0	100
Jogging	0	30	0	0	0	0	100
Running	0	1	29	0	0	0	96.67
Boxing	0	0	0	30	0	0	100
Hand waving	0	0	0	0	30	0	100
Hand clapping	0	0	0	0	0	30	100
Accuracy (%)	100	96.77	100	100	100	100	

Table 4. Confusion matrix obtained for the HAR system using optical flow on the Weizmann dataset.

Activity	Running	Walking	Skipping	Jack	Jump	Pjump	Side	Wave2	Wave1	Bending	Accuracy (%)
Running	3	0	0	0	0	0	0	0	0	0	100
Walking	1	2	0	0	0	0	0	0	0	0	66.67
Skipping	0	0	3	0	0	0	0	0	0	0	100
Jack	0	0	0	2	0	0	0	1	0	0	66.67
Jump	0	0	0	0	3	0	0	0	0	0	100
Pjump	0	0	0	0	0	2	1	0	0	0	66.67
Side	0	1	0	0	0	0	2	0	0	0	66.67
Wave2	0	0	0	0	0	0	0	2	1	0	66.67
Wave1	0	0	0	0	0	0	0	0	3	0	100
Bending	0	0	0	0	0	0	0	0	0	3	100
Accuracy (%)	75	66.67	100	100	100	100	66.67	66.67	75	100	

Table 5. Confusion matrix obtained for the proposed HAR system on the Weizmann dataset.

Activity	Running	Walking	Skipping	Jack	Jump	Pjump	Side	Wave2	Wave1	Bending	Accuracy (%)
Running	3	0	0	0	0	0	0	0	0	0	100
Walking	1	2	0	0	0	0	0	0	0	0	66.67
Skipping	0	0	3	0	0	0	0	0	0	0	100
Jack	0	0	0	3	0	0	0	0	0	0	100
Jump	0	0	0	0	3	0	0	0	0	0	100
Pjump	0	0	0	0	0	3	0	0	0	0	100
Side	0	0	0	0	0	0	3	0	0	0	100
Wave2	0	1	0	0	0	0	0	2	0	0	66.67
Wave1	0	0	0	0	0	0	0	0	3	0	100
Bending	0	0	0	0	0	0	0	0	0	3	100
Accuracy (%)	75	66.67	100	100	100	100	100	100	100	100	

In general, the accuracy of the conventional optical flow was 92.18%, which was increased to 99% by using the proposed method. The confusion matrices for the test data are shown in Tables 2 and 3. The KTH dataset contained 30 samples per class in the test data. In the Weizmann dataset, there are 9 or 10 different samples for each action. Here, 3 samples were used for testing and the rest for training. In this dataset, the scene is static, and the camera does not move; therefore, it was possible to accurately evaluate the feature extraction part of the proposed method without the effect of the camera movement cancellation. Tables 4 and 5 show the results obtained by the conventional and proposed methods, respectively. All the other simulation conditions remained unchanged.

The results demonstrate that the proposed method achieved an accuracy of 92%, while the conventional method achieved an accuracy of 83%. The final evaluation was performed using the HMDB51 dataset. Table 6 shows the number of samples and actions considered using this dataset.

Table 6. Total number of samples considered for the HMDB51 dataset for "Training" and "Testing" in terms of the action involved.

	Total	Train	Test
Clap	130	91	39
Wave	68	76	32
Climb	151	66	45
Jump	232	162	70
Run	548	384	164
Walk	64	73	31

The results reported here refer to the test data. As shown in Table 7, the overall accuracy of a HAR system using optical flow without camera movement cancellation on the HMDB51 dataset was $195/381 \cong 51\%$, which was calculated as the sum of the main diagonal elements divided by the total number of samples (see Section 4.2). Table 8 indicates that the system's overall accuracy was $271/381 \cong 71\%$. Therefore, by adding camera movement cancellation as a preprocessing step at the input of the used HAR system, the feature extraction efficiency was increased, and the system error decreased.

Table 7. Confusion matrix obtained for the HAR system using optical flow without camera movement cancellation on the HMDB51 dataset.

Activity	Clap	Wave	Climb	Jump	Run	Walk	Accuracy (%)
Clap	22	10	3	2	1	1	56.41
Wave	7	20	1	3	1	0	62.50
Climb	0	6	22	10	5	2	48.89
Jump	8	10	12	32	3	5	45.71
Run	8	8	12	15	82	39	50.00
Walk	0	2	0	3	9	17	54.84
Accuracy (%)	48.89	35.71	44.00	49.23	81.19	26.56	

Table 8. Confusion matrix obtained for the proposed HAR system on the HMDB51 dataset.

Activity	Clap	Wave	Climb	Jump	Run	Walk	Accuracy (%)
Clap	27	9	0	2	1	0	69.23
Wave	4	22	0	1	4	1	68.75
Climb	3	0	35	7	0	0	77.78
Jump	0	2	8	54	3	3	77.14
Run	3	6	4	9	108	34	65.85
Walk	0	0	4	0	2	25	80.65
Accuracy (%)	72.97	56.41	68.63	73.97	91.53	39.68	

5.3. Comparison with Other Methods

Considering that the proposed method focuses on the cancellation of camera movement and accurate moving object extraction, it was tested using a simple DNN structure. The reported percentages were obtained based on the test set in all comparisons. Table 9 and Figure 10 show the accuracy of recent state-of-the-art methods on the KTH dataset. It can be seen that the proposed method was slightly weaker than the methods proposed in [111,112], but it should be noted that these two methods are more complex than the proposed method. Table 10 and Figure 11 show the results for the Weizmann dataset, which confirm that the proposed method was more efficient than most existing methods.

Table 9. Comparison among state-of-the-art and proposed methods on the KTH dataset.

Ref	Accuracy (%)
Sargano et al. [113]	89.86
Dasari et al. [112]	87
El-Henawy et al. [114]	95
Jain et al. [115]	95
Shao et al. [116]	95
Yang et al. [117]	96
Cheng et al. [118]	97
Liu et al. [111]	94
Sharif et al. [119]	99
Elharrouss et al. [120]	99.82
Shao et al. [121]	97.50
Shi et al. [122]	96.80
Aslan L et al. [123]	96.16
Afza et al. [124]	100
Proposed	99.44

Table 10. Comparison among state-of-the-art and proposed methods on the Weizmann dataset.

Ref	Accuracy (%)
Jiang et al. [125]	95
Zhang et al. [126]	98
Kaminski et al. [127]	81
Sharif et al. [119]	95
Elharrouss et al. [120]	99.85
Simonyan and Zisserman [128]	92.80
Tran et al. [129]	95.69
Tran et al. [130]	94.03
Karpathy et al. [131]	91.11
Li et al. [132]	97.90
Proposed	96.67

The final comparison was conducted on the HMDB51 and UCF101 datasets. Due to the large amount of data in these datasets, researchers often use different assumptions to report their results. In this study, the proposed method was applied to some actions in the two datasets and compared with the most recent relevant methods. The confusion matrix of the test data for the UCF101 dataset is presented in Table 11, and the results of the proposed method are compared with the ones of related works in Table 12 and Figure 12.

Despite the significantly lower complexity of the proposed system, the achieved accuracy was competitive with most of the state-of-the-art methods in this field.

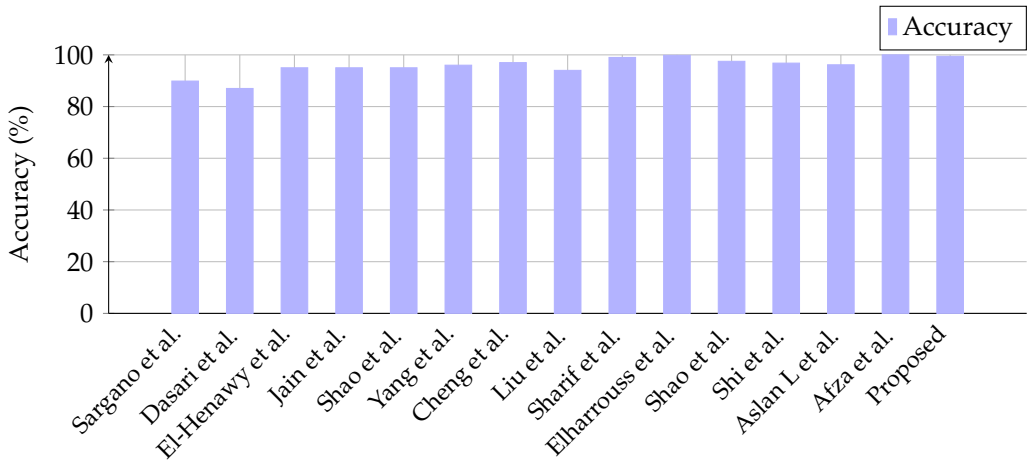

Figure 10. Accuracy of state-of-the-art and the proposed methods on the KTH dataset [111–124].

In other words, the accuracy improves significantly when the HAR methods minimise the influence of static background objects and camera movement in the preprocessing step and perform proper initial segmentations. In this study, all simulations were performed on MATLAB R2020b and PyCharm IDE Professional Edition 2020 using a personal computer with an Intel Core i7-9700 CPU (8 cores, 12M Cache, up to 4.70 GHz), 16GB of RAM and a GeForce RTX 2070 SUPER 8GB GPU. Among the different parts of the proposed method, the camera motion cancellation in the temporal feature extraction had the longest execution time. In the spatial feature extraction phase, all three parts had relatively equal execution time, but colour-based image segmentation took the longest. In total, the processing of every 10-frame set took about 1.9 s. Finally, considering deep neural network response time, action detection in a 10-s video of the UCF 101 dataset (Table 1), with a frame rate of 25 frames per second, took less than 60 s.

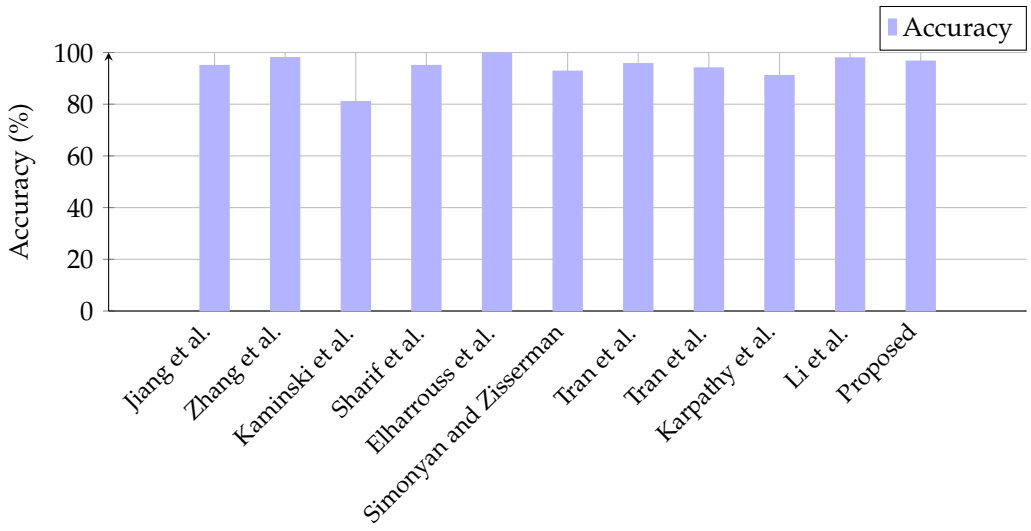

Figure 11. Comparison of accuracy among state-of-the-art and proposed methods on the Weizmann dataset [119,120,125–132].

Table 11. Confusion matrix for the proposed HAR system on the UCF101 dataset.

Activity	Basketball	Biking	Billiards	Diving	Haircut	HorseRace	Skiing	Walking with Dog	Accuracy (%)
Basketball	36	0	0	3	0	0	0	1	90
Biking	0	34	0	0	0	2	3	1	85
Billiards	1	0	41	0	0	0	0	3	91.11
Diving	3	0	0	40	0	2	0	0	88.89
Haircut	1	0	2	0	36	0	0	1	90
HorseRace	0	1	0	0	0	35	1	0	94.59
Skiing	0	3	0	0	0	0	37	0	92.5
Walking with Dog	1	0	1	0	0	0	0	35	94.59
Accuracy (%)	85.71	89.47	93.18	93.02	100	89.74	90.24	85.37	

Table 12. Comparison among state-of-the-art and proposed methods on the HMDB51 and UCF101 datasets.

Method	HMDB51 (%)	UCF101 (%)
Zhang et al. [133]	78.82	97.27
Carreira and Zisserman [134]	80.7	98
Wang et al. [135]	68.5	94
He et al. [136]	——	93.5
Jiang et al. [137]	72.2	96.2
Li et al. [132]	73.3	96.9
Proposed	71.13	90.74

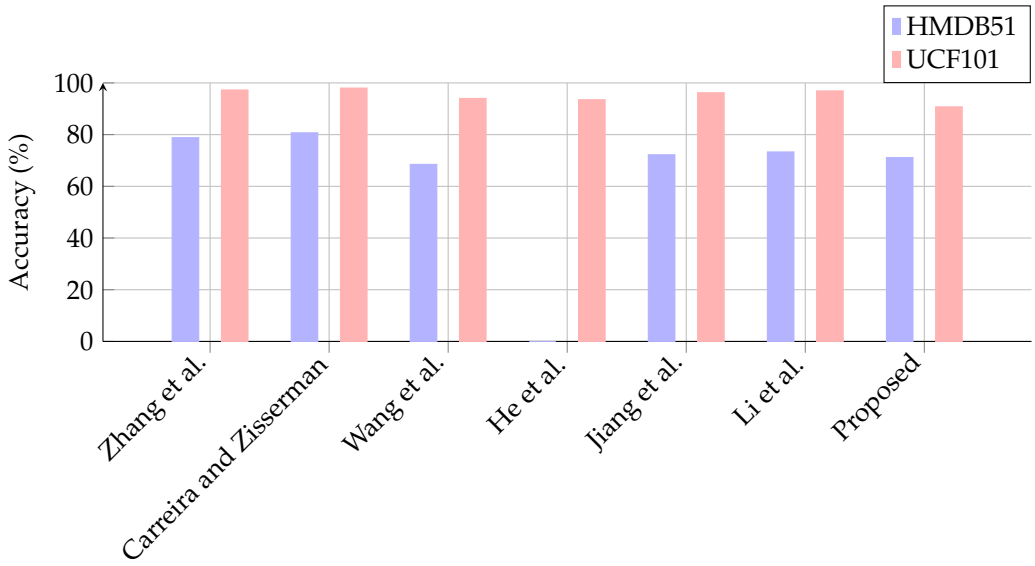

Figure 12. Accuracy on HMDB51 and UCF101 datasets for various video analysis methods vs proposed method [132–137].

6. Conclusions

In this study, an accurate method for a HAR system was proposed. Generally, a deep learning HAR system consists of two steps: preprocessing, which may include feature extraction, and deep learning-based classification. Camera movement and shot changes in the input video and the unknown location of the person under study in the scene are some of the most important factors that can cause errors in HAR systems. The proposed method improves the efficiency of existing HAR systems by adding preprocessing steps, including camera movement

cancellation and shot detection, and specifying the human's location as accurately as possible. To reuse the features extracted in the preprocessing step, the proposed method was designed so that extracted features can be used with a deep learning classifier. Therefore, it can be added to existing HAR systems with minimal computational overhead.

The proposed method was added to a deep learning-based HAR system, and the effectiveness of each step on system accuracy was validated. The results indicated that the accuracy achieved by the proposed method was comparable to that of state-of-the-art methods across all tested scenarios. Because the proposed unsupervised approach separates the important regions of each frame based on motion and then texture- and colour-based features, it can be added as a preprocessing block to HAR methods and enhance their overall performance.

One of the main areas of future work is to investigate the sensitivity of existing HAR methods to camera movement and shot in the input videos. Another possibility would be to investigate the impact of camera movement and shot changes on different optical flow extraction methods. The findings will help to develop more robust HAR methods that are less sensitive to camera movement and shot in videos.

Author Contributions: Conceptualization, funding acquisition and supervision by J.M.R.S.T.; investigation, data collection, formal analysis and writing—original draft preparation by A.A. and V.H.; writing—review and editing by J.J.M.M. and J.M.R.S.T. All authors have read and agreed to the published version of the manuscript.

Funding: This article is partially a result of the project "Sensitive Industry", co-funded by the European Regional Development Fund (ERDF) through the Operational Programme for Competitiveness and Internationalization (COMPETE 2020) under the PORTUGAL 2020 Partnership Agreement. The second author would like to thank "Fundação para a Ciência e Tecnologia (FCT)", in Portugal, for his PhD grant with reference 2021.08660.BD.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Used data is identified in the article and is publicly available.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Caetano, C.; dos Santos, J.A.; Schwartz, W.R. Optical Flow Co-occurrence Matrices: A novel spatiotemporal feature descriptor. In Proceedings of the 2016 23rd International Conference on Pattern Recognition (ICPR), Cancun, Mexico, 4–8 December 2016; pp. 1947–1952. [CrossRef]
2. Gupta, A.; Balan, M.S. Action recognition from optical flow visualizations. In Proceedings of the 2nd International Conference on Computer Vision & Image Processing, Roorkee, India, 1 March 2018; pp. 397–408. [CrossRef]
3. Kumar, S.S.; John, M. Human activity recognition using optical flow based feature set. In Proceedings of the 2016 IEEE International Carnahan Conference on Security Technology (ICCST), Orlando, FL, USA, 24–27 October 2016; pp. 1–5. [CrossRef]
4. Rashwan, H.A.; Garcia, M.A.; Abdulwahab, S.; Puig, D. Action representation and recognition through temporal co-occurrence of flow fields and convolutional neural networks. *Multimed. Tools Appl.* **2020**, *79*, 34141–34158. [CrossRef]
5. Rashwan, H.A.; García, M.Á.; Chambon, S.; Puig, D. Gait representation and recognition from temporal co-occurrence of flow fields. *Mach. Vis. Appl.* **2019**, *30*, 139–152. [CrossRef]
6. Xu, G.L.; Zhou, H.; Yuan, L.Y.; Huang, Y.Y. Using Improved Dense Trajectory Feature to Realize Action Recognition. *J. Comput.* **2021**, *32*, 94–108. [CrossRef]
7. Liu, C.; Ying, J.; Yang, H.; Hu, X.; Liu, J. Improved human action recognition approach based on two-stream convolutional neural network model. *Vis. Comput.* **2021**, *37*, 1327–1341. [CrossRef]
8. Kumar, B.S.; Raju, S.V.; Reddy, H.V. Human action recognition using a novel deep learning approach. *Proc. Iop Conf. Ser. Mater. Sci. Eng.* **2021**, *1042*, 012031. [CrossRef]
9. Abdelbaky, A.; Aly, S. Two-stream spatiotemporal feature fusion for human action recognition. *Vis. Comput.* **2021**, *37*, 1821–1835. [CrossRef]
10. Guha, R.; Khan, A.H.; Singh, P.K.; Sarkar, R.; Bhattacharjee, D. CGA: A new feature selection model for visual human action recognition. *Neural Comput. Appl.* **2021**, *33*, 5267–5286. [CrossRef]
11. Dash, S.C.B.; Mishra, S.R.; Srujan Raju, K.; Narasimha Prasad, L. Human action recognition using a hybrid deep learning heuristic. *Soft Comput.* **2021**, *25*, 13079–13092. [CrossRef]

12. Khan, M.A.; Zhang, Y.D.; Khan, S.A.; Attique, M.; Rehman, A.; Seo, S. A resource conscious human action recognition framework using 26-layered deep convolutional neural network. *Multimed. Tools Appl.* **2021**, *80*, 35827–35849. [CrossRef]
13. Jaouedi, N.; Boujnah, N.; Bouhlel, M.S. A new hybrid deep learning model for human action recognition. *J. King Saud Univ.-Comput. Inf. Sci.* **2020**, *32*, 447–453. [CrossRef]
14. Zheng, Y.; Yao, H.; Sun, X.; Zhao, S.; Porikli, F. Distinctive action sketch for human action recognition. *Signal Process.* **2018**, *144*, 323–332. [CrossRef]
15. Ramya, P.; Rajeswari, R. Human action recognition using distance transform and entropy based features. *Multimed. Tools Appl.* **2021**, *80*, 8147–8173. [CrossRef]
16. Haddad, M.; Ghassab, V.K.; Najar, F.; Bouguila, N. A statistical framework for few-shot action recognition. *Multimed. Tools Appl.* **2021**, *80*, 24303–24318. [CrossRef]
17. Snoun, A.; Jlidi, N.; Bouchrika, T.; Jemai, O.; Zaied, M. Towards a deep human activity recognition approach based on video to image transformation with skeleton data. *Multimed. Tools Appl.* **2021**, *80*, 29675–29698. [CrossRef]
18. Abdelbaky, A.; Aly, S. Human action recognition using three orthogonal planes with unsupervised deep convolutional neural network. *Multimed. Tools Appl.* **2021**, *80*, 20019–20043. [CrossRef]
19. Xia, L.; Ma, W. Human action recognition using high-order feature of optical flows. *J. Supercomput.* **2021**, *77*, 14230–14251. [CrossRef]
20. Martínez Carrillo, F.; Gouiffès, M.; Garzón Villamizar, G.; Manzanera, A. A compact and recursive Riemannian motion descriptor for untrimmed activity recognition. *J. Real-Time Image Process.* **2021**, *18*, 1867–1880. [CrossRef]
21. Guo, Y.; Wang, X. Applying TS-DBN model into sports behavior recognition with deep learning approach. *J. Supercomput.* **2021**, *77*, 12192–12208. [CrossRef]
22. Aghaei, A.; Nazari, A.; Moghaddam, M.E. Sparse deep LSTMs with convolutional attention for human action recognition. *SN Comput. Sci.* **2021**, *2*, 151. [CrossRef]
23. Zebhi, S.; AlModarresi, S.M.T.; Abootalebi, V. Human activity recognition using pre-trained network with informative templates. *Int. J. Mach. Learn. Cybern.* **2021**, *12*, 3449–3461. [CrossRef]
24. Wang, Y.; Shen, X.; Chen, H.; Sun, J. Action Recognition in Videos with Spatio-Temporal Fusion 3D Convolutional Neural Networks. *Pattern Recognit. Image Anal.* **2021**, *31*, 580–587. [CrossRef]
25. Khan, S.; Khan, M.A.; Alhaisoni, M.; Tariq, U.; Yong, H.S.; Armghan, A.; Alenezi, F. Human action recognition: A paradigm of best deep learning features selection and serial based extended fusion. *Sensors* **2021**, *21*, 7941. [CrossRef] [PubMed]
26. Xu, Q.; Zheng, W.; Song, Y.; Zhang, C.; Yuan, X.; Li, Y. Scene image and human skeleton-based dual-stream human action recognition. *Pattern Recognit. Lett.* **2021**, *148*, 136–145. [CrossRef]
27. Wu, C.; Li, Y.; Zhang, Y.; Liu, B. Double constrained bag of words for human action recognition. *Signal Process. Image Commun.* **2021**, *98*, 116399. [CrossRef]
28. El-Assal, M.; Tirilly, P.; Bilasco, I.M. A Study On the Effects of Pre-processing On Spatio-temporal Action Recognition Using Spiking Neural Networks Trained with STDP. In Proceedings of the 2021 International Conference on Content-Based Multimedia Indexing (CBMI), Lille, France, 28–30 June 2021; pp. 1–6. [CrossRef]
29. Boualia, S.N.; Amara, N.E.B. 3D CNN for Human Action Recognition. In Proceedings of the 2021 18th International Multi-Conference on Systems, Signals & Devices (SSD), Monastir, Tunisia, 22–25 March 2021; pp. 276–282. [CrossRef]
30. Mishra, O.; Kavimandan, P.; Kapoor, R. Modal Frequencies Based Human Action Recognition Using Silhouettes And Simplicial Elements. *Int. J. Eng.* **2022**, *35*, 45–52. [CrossRef]
31. Ha, J.; Shin, J.; Park, H.; Paik, J. Action recognition network using stacked short-term deep features and bidirectional moving average. *Appl. Sci.* **2021**, *11*, 5563. [CrossRef]
32. Gharahbagh, A.A.; Hajihashemi, V.; Ferreira, M.C.; Machado, J.J.; Tavares, J.M.R. Best Frame Selection to Enhance Training Step Efficiency in Video-Based Human Action Recognition. *Appl. Sci.* **2022**, *12*, 1830. [CrossRef]
33. Hajihashemi, V.; Pakizeh, E. Human activity recognition in videos based on a Two Levels K-means and Hierarchical Codebooks. *Int. J. Mechatron. Electr. Comput. Technol* **2016**, *6*, 3152–3159.
34. Deshpnande, A.; Warhade, K.K. An Improved Model for Human Activity Recognition by Integrated feature Approach and Optimized SVM. In Proceedings of the 2021 International Conference on Emerging Smart Computing and Informatics (ESCI), Pune, India, 5–7 March 2021; pp. 571–576. [CrossRef]
35. Ma, J.; Tao, X.; Ma, J.; Hong, X.; Gong, Y. Class incremental learning for video action classification. In Proceedings of the 2021 IEEE International Conference on Image Processing (ICIP), Anchorage, AK, USA, 19–22 September 2021; pp. 504–508. [CrossRef]
36. Shekokar, R.; Kale, S. Deep Learning for Human Action Recognition. In Proceedings of the 2021 6th International Conference for Convergence in Technology (I2CT), Maharashtra, India, 2–4 April 2021; pp. 1–5. [CrossRef]
37. Sawanglok, T.; Songmuang, P. Data Preparation for Reducing Computational Time with Transpose Stack Matrix for Action Recognition. In Proceedings of the 2021 13th International Conference on Knowledge and Smart Technology (KST), Bangsaen, Thailand, 21–24 January 2021; pp. 141–146. [CrossRef]
38. Shi, S.; Jung, C. Deep Metric Learning for Human Action Recognition with SlowFast Networks. In Proceedings of the 2021 International Conference on Visual Communications and Image Processing (VCIP), Munich, Germany, 5–8 December 2021; pp. 1–5. [CrossRef]

39. Gao, Z.; Gu, Q.; Han, Z. Human Behavior Recognition Method based on Two-layer LSTM Network with Attention Mechanism. *J. Phys. Conf. Ser.* **2021**, *2093*, 012006. [CrossRef]
40. Wang, J.; Xia, L.; Ma, W. Human action recognition based on motion feature and manifold learning. *IEEE Access* **2021**, *9*, 89287–89299. [CrossRef]
41. Nasir, I.M.; Raza, M.; Shah, J.H.; Khan, M.A.; Rehman, A. Human action recognition using machine learning in uncontrolled environment. In Proceedings of the 2021 1st International Conference on Artificial Intelligence and Data Analytics (CAIDA), Riyadh, Saudi Arabia, 6–7 April 2021; pp. 182–187. [CrossRef]
42. Sowmyayani, S.; Rani, P. STHARNet: Spatio-temporal human action recognition network in content based video retrieval. *Multimed. Tools Appl.* **2022**, *82*, 38051–38066. [CrossRef]
43. Singh, K.; Dhiman, C.; Vishwakarma, D.K.; Makhija, H.; Walia, G.S. A sparse coded composite descriptor for human activity recognition. *Expert Syst.* **2022**, *39*, e12805. [CrossRef]
44. Mithsara, W. Comparative Analysis of AI-powered Approaches for Skeleton-based Child and Adult Action Recognition in Multi-person Environment. In Proceedings of the 2022 International Conference on Computer Science and Software Engineering (CSASE), Duhok, Iraq, 15–17 March 2022; pp. 24–29. [CrossRef]
45. Nair, S.A.L.; Megalingam, R.K. Fusion of Bag of Visual Words with Neural Network for Human Action Recognition. In Proceedings of the 2022 12th International Conference on Cloud Computing, Data Science & Engineering (Confluence), Noida, India, 27–28 January 2022; pp. 14–19. [CrossRef]
46. Megalingam, R.K.; Nair S., A.L. Human Action Recognition: A Review. In Proceedings of the 2021 10th International Conference on System Modeling & Advancement in Research Trends (SMART), Moradabad, India, 10–11 December 2021; pp. 249–252. [CrossRef]
47. Bayoudh, K.; Hamdaoui, F.; Mtibaa, A. An Attention-based Hybrid 2D/3D CNN-LSTM for Human Action Recognition. In Proceedings of the 2022 2nd International Conference on Computing and Information Technology (ICCIT), Tabuk, Saudi Arabia, 25–27 January 2022; pp. 97–103. [CrossRef]
48. Liang, Y.; Huang, H.; Li, J.; Dong, X.; Chen, M.; Yang, S.; Chen, H. Action recognition based on discrete cosine transform by optical pixel-wise encoding. *APL Photonics* **2022**, *7*, 116101. [CrossRef]
49. Khater, S.; Hadhoud, M.; Fayek, M.B. A novel human activity recognition architecture: Using residual inception ConvLSTM layer. *J. Eng. Appl. Sci.* **2022**, *69*, 45. [CrossRef]
50. Momin, M.S.; Sufian, A.; Barman, D.; Dutta, P.; Dong, M.; Leo, M. In-home older adults' activity pattern monitoring using depth sensors: A review. *Sensors* **2022**, *22*, 9067. [CrossRef] [PubMed]
51. Sun, Z.; Ke, Q.; Rahmani, H.; Bennamoun, M.; Wang, G.; Liu, J. Human action recognition from various data modalities: A review. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *45*, 3200–3225. [CrossRef]
52. Wu, Z.; Du, H. Research on Human Action Feature Detection and Recognition Algorithm Based on Deep Learning. *Mob. Inf. Syst.* **2022**, *2022*, 4652946. [CrossRef]
53. Ahn, D.; Kim, S.; Hong, H.; Ko, B.C. STAR-Transformer: A spatio-temporal cross attention transformer for human action recognition. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 3–7 January 2023; pp. 3330–3339.
54. Vaitesswar, U.; Yeo, C.K. Multi-Range Mixed Graph Convolution Network for Skeleton-Based Action Recognition. In Proceedings of the 2023 5th Asia Pacific Information Technology Conference, Ho Chi Minh, Vietnam, 9–11 February 2023; pp. 49–54. [CrossRef]
55. Lee, J.; Lee, M.; Lee, D.; Lee, S. Hierarchically decomposed graph convolutional networks for skeleton-based action recognition. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, 2–6 October 2023; pp. 10444–10453.
56. Wu, J.; Wang, L.; Chong, G.; Feng, H. 2S-AGCN Human Behavior Recognition Based on New Partition Strategy. In Proceedings of the 2022 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), Chiang Mai, Thailand, 7–10 November 2022; pp. 157–163. [CrossRef]
57. Radulescu, B.A.; Radulescu, V. Modeling 3D convolution architecture for actions recognition. In Proceedings of the Information Storage and Processing Systems. American Society of Mechanical Engineers, Online, 2–3 June 2021; Volume 84799, p. V001T01A001. [CrossRef]
58. Yan, Z.; Yongfeng, Q.; Xiaoxu, P. Dangerous Action Recognition for Spatial-Temporal Graph Convolutional Networks. In Proceedings of the 2022 IEEE 12th International Conference on Electronics Information and Emergency Communication (ICEIEC), Beijing, China, 15–17 July 2022; pp. 216–219. [CrossRef]
59. Liao, T.; Zhao, J.; Liu, Y.; Ivanov, K.; Xiong, J.; Yan, Y. Deep transfer learning with graph neural network for sensor-based human activity recognition. In Proceedings of the 2022 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), Las Vegas, NV, USA, 6–8 December 2022; pp. 2445–2452. [CrossRef]
60. Bi, C.; Yuan, Y.; Zhang, J.; Shi, Y.; Xiang, Y.; Wang, Y.; Zhang, R. Dynamic mode decomposition based video shot detection. *IEEE Access* **2018**, *6*, 21397–21407. [CrossRef]
61. Lu, Z.M.; Shi, Y. Fast video shot boundary detection based on SVD and pattern matching. *IEEE Trans. Image Process.* **2013**, *22*, 5136–5145. [CrossRef]
62. Mishra, R. Video shot boundary detection using hybrid dual tree complex wavelet transform with Walsh Hadamard transform. *Multimed. Tools Appl.* **2021**, *80*, 28109–28135. [CrossRef]

63. Rashmi, B.; Nagendraswamy, H. Video shot boundary detection using block based cumulative approach. *Multimed. Tools Appl.* **2021**, *80*, 641–664. [CrossRef]
64. Hu, W.C.; Chen, C.H.; Chen, T.Y.; Huang, D.Y.; Wu, Z.C. Moving object detection and tracking from video captured by moving camera. *J. Vis. Commun. Image Represent.* **2015**, *30*, 164–180. [CrossRef]
65. Moore, B.E.; Gao, C.; Nadakuditi, R.R. Panoramic robust pca for foreground–background separation on noisy, free-motion camera video. *IEEE Trans. Comput. Imaging* **2019**, *5*, 195–211. [CrossRef]
66. Zhang, W.; Sun, X.; Yu, Q. Moving Object Detection under a Moving Camera via Background Orientation Reconstruction. *Sensors* **2020**, *20*, 3103. [CrossRef] [PubMed]
67. Ahammed, M.J.; Venkata Rao, V.; Bajidvali, S. Human Gait Detection Using Silhouette Image Recognition. *Turk. J. Comput. Math. Educ. (TURCOMAT)* **2021**, *12*, 1320–1326. [CrossRef]
68. Lam, T.H.; Lee, R.S. A new representation for human gait recognition: Motion silhouettes image (MSI). In *Advances in Biometrics*; Springer: Berlin/Heidelberg, Germany, 2005; pp. 612–618. [CrossRef]
69. Jawed, B.; Khalifa, O.O.; Bhuiyan, S.S.N. Human gait recognition system. In Proceedings of the 2018 7th International Conference on Computer and Communication Engineering (ICCCE), Kuala Lumpur, Malaysia, 19–20 September 2018; pp. 89–92. [CrossRef]
70. Maity, S.; Chakrabarti, A.; Bhattacharjee, D. Robust human action recognition using AREI features and trajectory analysis from silhouette image sequence. *IETE J. Res.* **2019**, *65*, 236–249. [CrossRef]
71. Vishwakarma, D.K.; Dhiman, C. A unified model for human activity recognition using spatial distribution of gradients and difference of Gaussian kernel. *Vis. Comput.* **2019**, *35*, 1595–1613. [CrossRef]
72. Yang, D.; Peng, B.; Al-Huda, Z.; Malik, A.; Zhai, D. An overview of edge and object contour detection. *Neurocomputing* **2022**, *488*, 470–493. [CrossRef]
73. Kass, M.; Witkin, A.; Terzopoulos, D. Snakes: Active contour models. *Int. J. Comput. Vis.* **1988**, *1*, 321–331. [CrossRef]
74. Cohen, L.D. On active contour models and balloons. *CVGIP: Image Underst.* **1991**, *53*, 211–218. [CrossRef]
75. Xu, C.; Prince, J.L. Snakes, shapes, and gradient vector flow. *IEEE Trans. Image Process.* **1998**, *7*, 359–369. [CrossRef] [PubMed]
76. Li, B.; Acton, S.T. Active contour external force using vector field convolution for image segmentation. *IEEE Trans. Image Process.* **2007**, *16*, 2096–2106. [CrossRef]
77. Mumford, D.; Shah, J. Boundary detection by minimizing functionals. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA, 9–13 June 1985; Volume 17, pp. 137–154.
78. Caselles, V.; Kimmel, R.; Sapiro, G. Geodesic active contours. *Int. J. Comput. Vis.* **1997**, *22*, 61–79. [CrossRef]
79. Chan, T.F.; Vese, L.A. Active contours without edges. *IEEE Trans. Image Process.* **2001**, *10*, 266–277. [CrossRef] [PubMed]
80. Li, C.; Kao, C.Y.; Gore, J.C.; Ding, Z. Minimization of region-scalable fitting energy for image segmentation. *IEEE Trans. Image Process.* **2008**, *17*, 1940–1949. [CrossRef]
81. Ghosh, A.; Bandyopadhyay, S. Image co-segmentation using dual active contours. *Appl. Soft Comput.* **2018**, *66*, 413–427. [CrossRef]
82. Han, J.; Quan, R.; Zhang, D.; Nie, F. Robust object co-segmentation using background prior. *IEEE Trans. Image Process.* **2017**, *27*, 1639–1651. [CrossRef] [PubMed]
83. Merdassi, H.; Barhoumi, W.; Zagrouba, E. A comprehensive overview of relevant methods of image cosegmentation. *Expert Syst. Appl.* **2020**, *140*, 112901. [CrossRef]
84. Anitha, J.; Pandian, S.I.A.; Agnes, S.A. An efficient multilevel color image thresholding based on modified whale optimization algorithm. *Expert Syst. Appl.* **2021**, *178*, 115003. [CrossRef]
85. Jing, Y.; Kong, T.; Wang, W.; Wang, L.; Li, L.; Tan, T. Locate then segment: A strong pipeline for referring image segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 9858–9867.
86. Badrinarayanan, V.; Kendall, A.; Cipolla, R. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2481–2495. [CrossRef]
87. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *40*, 834–848. [CrossRef] [PubMed]
88. Lin, G.; Milan, A.; Shen, C.; Reid, I. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1925–1934.
89. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
90. Wang, X.; Kong, T.; Shen, C.; Jiang, Y.; Li, L. Solo: Segmenting objects by locations. In Proceedings of the European Conference on Computer Vision. Springer, Glasgow, UK, 23–28 August 2020; pp. 649–665. [CrossRef]
91. Kabilan, R.; Devaraj, G.P.; Muthuraman, U.; Muthukumaran, N.; Gabriel, J.Z.; Swetha, R. Efficient color image segmentation using fastmap algorithm. In Proceedings of the 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV), Tirunelveli, India, 4–6 February 2021; pp. 1134–1141. [CrossRef]
92. Abualigah, L.; Almotairi, K.H.; Elaziz, M.A. Multilevel thresholding image segmentation using meta-heuristic optimization algorithms: Comparative analysis, open challenges and new trends. *Appl. Intell.* **2022**, *53*, 11654–11704. [CrossRef]

93. Sathya, P.; Kalyani, R.; Sakthivel, V. Color image segmentation using Kapur, Otsu and minimum cross entropy functions based on exchange market algorithm. *Expert Syst. Appl.* **2021**, *172*, 114636. [CrossRef]
94. Plyer, A.; Le Besnerais, G.; Champagnat, F. Massively parallel Lucas Kanade optical flow for real-time video processing applications. *J. Real-Time Image Process.* **2016**, *11*, 713–730. [CrossRef]
95. Sundberg, P.; Brox, T.; Maire, M.; Arbeláez, P.; Malik, J. Occlusion boundary detection and figure/ground assignment from optical flow. In Proceedings of the CVPR 2011, Colorado Springs, CO, USA, 20–25 June 2011; pp. 2233–2240. [CrossRef]
96. Galasso, F.; Nagaraja, N.S.; Cardenas, T.J.; Brox, T.; Schiele, B. A unified video segmentation benchmark: Annotation, metrics and analysis. In Proceedings of the IEEE International Conference on Computer Vision, Sydney, Australia, 1–8 December 2013; pp. 3527–3534.
97. Zhao, Q.; Yan, B.; Yang, J.; Shi, Y. Evolutionary Robust Clustering Over Time for Temporal Data. *IEEE Trans. Cybern.* **2022**, *53*, 4334–4346. [CrossRef]
98. Han, D.; Xiao, Y.; Zhan, P.; Li, T.; Fan, M. A Semi-Supervised Video Object Segmentation Method Based on ConvNext and Unet. In Proceedings of the 2022 41st Chinese Control Conference (CCC), Hefei, China, 25–27 July 2022; pp. 7425–7431. [CrossRef]
99. Hu, Y.T.; Huang, J.B.; Schwing, A.G. Unsupervised video object segmentation using motion saliency-guided spatio-temporal propagation. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 2–14 September 2018; pp. 786–802.
100. Schuldt, C.; Laptev, I.; Caputo, B. Recognizing human actions: A local SVM approach. In Proceedings of the 17th International Conference on Pattern Recognition, Cambridge, UK, 23–26 August 2004; Volume 3, pp. 32–36. [CrossRef]
101. Laptev, I. Local Spatio-Temporal Image Features for Motion Interpretation. Ph.D. Thesis, KTH Numerisk Analys Och Datalogi, Stockholm, Sweden, 2004.
102. Laptev, I.; Lindeberg, T. Local descriptors for spatio-temporal recognition. In Proceedings of the International Workshop on Spatial Coherence for Visual Motion Analysis, Prague, Czech Republic, 15 May 2004; pp. 91–103. [CrossRef]
103. Laptev, I.; Lindeberg, T. Velocity adaptation of space-time interest points. In Proceedings of the 17th International Conference on Pattern Recognition, Cambridge, UK, 23–26 August 2004; Volume 1, pp. 52–56. [CrossRef]
104. Laptev, I. On space-time interest points. *Int. J. Comput. Vis.* **2005**, *64*, 107–123. [CrossRef]
105. Blank, M.; Gorelick, L.; Shechtman, E.; Irani, M.; Basri, R. Actions as space-time shapes. In Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV'05), Beijing, China, 17–21 October 2005; Volume 2, pp. 1395–1402. [CrossRef]
106. Nadeem, A.; Jalal, A.; Kim, K. Human actions tracking and recognition based on body parts detection via Artificial neural network. In Proceedings of the 2020 3rd International Conference on Advancements in Computational Sciences (ICACS), Lahore, Pakistan, 17–19 February 2020; pp. 1–6. [CrossRef]
107. Nigam, S.; Khare, A. Integration of moment invariants and uniform local binary patterns for human activity recognition in video sequences. *Multimed. Tools Appl.* **2016**, *75*, 17303–17332. [CrossRef]
108. Basavaiah, J.; Patil, C.; Patil, C. Robust feature extraction and classification based automated human action recognition system for multiple datasets. *Int. J. Intell. Eng. Syst.* **2020**, *13*, 13–24. [CrossRef]
109. Kuehne, H.; Jhuang, H.; Garrote, E.; Poggio, T.; Serre, T. HMDB: A large video database for human motion recognition. In Proceedings of the 2011 International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 2556–2563. [CrossRef]
110. Soomro, K.; Zamir, A.R.; Shah, M. A dataset of 101 human action classes from videos in the wild. *arXiv* **2012**, arXiv:1212.0402.
111. Liu, H.; Ju, Z.; Ji, X.; Chan, C.S.; Khoury, M. Study of human action recognition based on improved spatio-temporal features. In *Human Motion Sensing and Recognition*; Springer: Berlin/Heidelberg, Germany, 2017; pp. 233–250. [CrossRef]
112. Dasari, R.; Chen, C.W. Mpeg cdvs feature trajectories for action recognition in videos. In Proceedings of the 2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR), Miami, FL, USA, 10–12 April 2018; pp. 301–304. [CrossRef]
113. Sargano, A.B.; Wang, X.; Angelov, P.; Habib, Z. Human action recognition using transfer learning with deep representations. In Proceedings of the 2017 International Joint Conference on Neural Networks (IJCNN), Anchorage, AL, USA, 14–19 May 2017; pp. 463–469. [CrossRef]
114. El-Henawy, I.; Ahmed, K.; Mahmoud, H. Action recognition using fast HOG3D of integral videos and Smith–Waterman partial matching. *IET Image Process.* **2018**, *12*, 896–908. [CrossRef]
115. Jain, S.B.; Sreeraj, M. Multi-posture human detection based on hybrid HOG-BO feature. In Proceedings of the 2015 Fifth international conference on advances in computing and communications (ICACC), Kochi, India, 2–4 September 2015; pp. 37–40. [CrossRef]
116. Shao, L.; Zhen, X.; Tao, D.; Li, X. Spatio-temporal Laplacian pyramid coding for action recognition. *IEEE Trans. Cybern.* **2013**, *44*, 817–827. [CrossRef] [PubMed]
117. Yang, J.; Ma, Z.; Xie, M. Action recognition based on multi-scale oriented neighborhood features. *Int. J. Signal Process. Image Process. Pattern Recognit.* **2015**, *8*, 241–254. [CrossRef]
118. Cheng, S.; Yang, J.; Ma, Z.; Xie, M. Action recognition based on spatio-temporal log-Euclidean covariance matrix. *Int. J. Signal Process. Image Process. Pattern Recognit.* **2016**, *9*, 95–106. [CrossRef]
119. Sharif, M.; Khan, M.A.; Akram, T.; Javed, M.Y.; Saba, T.; Rehman, A. A framework of human detection and action recognition based on uniform segmentation and combination of Euclidean distance and joint entropy-based features selection. *EURASIP J. Image Video Process.* **2017**, *2017*, 89. [CrossRef]

120. Elharrouss, O.; Almaadeed, N.; Al-Maadeed, S.; Bouridane, A.; Beghdadi, A. A combined multiple action recognition and summarization for surveillance video sequences. *Appl. Intell.* **2021**, *51*, 690–712. [CrossRef]
121. Shao, L.; Liu, L.; Yu, M. Kernelized multiview projection for robust action recognition. *Int. J. Comput. Vis.* **2016**, *118*, 115–129. [CrossRef]
122. Shi, Y.; Tian, Y.; Wang, Y.; Huang, T. Sequential deep trajectory descriptor for action recognition with three-stream CNN. *IEEE Trans. Multimed.* **2017**, *19*, 1510–1520. [CrossRef]
123. Aslan, M.F.; Durdu, A.; Sabanci, K. Human action recognition with bag of visual words using different machine learning methods and hyperparameter optimization. *Neural Comput. Appl.* **2020**, *32*, 8585–8597. [CrossRef]
124. Afza, F.; Khan, M.A.; Sharif, M.; Kadry, S.; Manogaran, G.; Saba, T.; Ashraf, I.; Damaševičius, R. A framework of human action recognition using length control features fusion and weighted entropy-variances based feature selection. *Image Vis. Comput.* **2021**, *106*, 104090. [CrossRef]
125. Jiang, J.; He, X.; Gao, M.; Wang, X.; Wu, X. Human action recognition via compressive-sensing-based dimensionality reduction. *Optik* **2015**, *126*, 882–887. [CrossRef]
126. Zhang, S.; Zhang, W.; Li, Y. Human action recognition based on multifeature fusion. In Proceedings of the Chinese Intelligent Systems Conference, Xiamen, China, 22–23 October 2016; pp. 183–192. [CrossRef]
127. Kamiński, Ł.; Maćkowiak, S.; Domański, M. Human activity recognition using standard descriptors of MPEG CDVS. In Proceedings of the 2017 IEEE International Conference on Multimedia & Expo Workshops (ICMEW), Hong Kong, China, 10–14 July 2017; pp. 121–126. [CrossRef]
128. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**. [CrossRef]
129. Tran, D.; Wang, H.; Torresani, L.; Feiszli, M. Video classification with channel-separated convolutional networks. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 5552–5561.
130. Tran, D.; Wang, H.; Torresani, L.; Ray, J.; LeCun, Y.; Paluri, M. A closer look at spatiotemporal convolutions for action recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 6450–6459.
131. Karpathy, A.; Toderici, G.; Shetty, S.; Leung, T.; Sukthankar, R.; Fei-Fei, L. Large-scale video classification with convolutional neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 1725–1732.
132. Li, Y.; Ji, B.; Shi, X.; Zhang, J.; Kang, B.; Wang, L. Tea: Temporal excitation and aggregation for action recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 909–918.
133. Zhang, X.Y.; Huang, Y.P.; Mi, Y.; Pei, Y.T.; Zou, Q.; Wang, S. Video sketch: A middle-level representation for action recognition. *Appl. Intell.* **2021**, *51*, 2589–2608. [CrossRef]
134. Carreira, J.; Zisserman, A. Quo vadis, action recognition? a new model and the kinetics dataset. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 6299–6308.
135. Wang, L.; Xiong, Y.; Wang, Z.; Qiao, Y.; Lin, D.; Tang, X.; Van Gool, L. Temporal segment networks: Towards good practices for deep action recognition. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 20–36. [CrossRef]
136. He, D.; Zhou, Z.; Gan, C.; Li, F.; Liu, X.; Li, Y.; Wang, L.; Wen, S. Stnet: Local and global spatial-temporal modeling for action recognition. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; Volume 33, pp. 8401–8408. [CrossRef]
137. Jiang, B.; Wang, M.; Gan, W.; Wu, W.; Yan, J. Stm: Spatiotemporal and motion encoding for action recognition. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 2000–2009.