

Article

Best Frame Selection to Enhance Training Step Efficiency in Video-Based Human Action Recognition

Abdorrezza Alavi Gharahbagh ¹, Vahid Hajihashemi ¹, Marta Campos Ferreira 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.G.); up201912327@fe.up.pt (V.H.); mferreira@fe.up.pt (M.C.F.)

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 recent years, with the growth of digital media and modern imaging equipment, the use of video processing algorithms and semantic film and image management has expanded. The usage of different video datasets in training artificial intelligence algorithms is also rapidly expanding in various fields. Due to the high volume of information in a video, its processing is still expensive for most hardware systems, mainly in terms of its required runtime and memory. Hence, the optimal selection of keyframes to minimize redundant information in video processing systems has become noteworthy in facilitating this problem. Eliminating some frames can simultaneously reduce the required computational load, hardware cost, memory and processing time of intelligent video-based systems. Based on the aforementioned reasons, this research proposes a method for selecting keyframes and adaptive cropping input video for human action recognition (HAR) systems. The proposed method combines edge detection, simple difference, adaptive thresholding and 1D and 2D average filter algorithms in a hierarchical method. Some HAR methods are trained with videos processed by the proposed method to assess its efficiency. The results demonstrate that the application of the proposed method increases the accuracy of the HAR system by up to 3% compared to random image selection and cropping methods. Additionally, for most cases, the proposed method reduces the training time of the used machine learning algorithm.

Keywords: machine learning; keyframes selection; adaptive cropping; video processing



Citation: Gharahbagh, A.A.; Hajihashemi, V.; Ferreira, M.C.; Machado, J.J.M.; Tavares, J.M.R.S. Best Frame Selection to Enhance Training Step Efficiency in Video-Based Human Action Recognition. *Appl. Sci.* **2022**, *12*, 1830. <https://doi.org/10.3390/app12041830>

Academic Editors: Wen-June Wang and Chung-Hsun Sun

Received: 25 December 2021

Accepted: 6 February 2022

Published: 10 February 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The use of video and digital content has expanded due to smartphones and other available imaging equipment. The fast growth of video content on social media and the internet has led to the definition of issues such as selecting essential frames of a video to use as a video marker or summarizing and reducing the required memory. In addition, the use of the most important video frames to reduce their review time in different algorithms or to better train machine learning algorithms are among the other uses of selecting the best frames of a video [1]. Due to the wide range of applications, the goal of selecting keyframes is essential. For example, in applications such as selecting keyframes to create a short film as a movie trailer, the goal is to assure an appealing trailer firm in order to prompt people to spend money, go to the cinema and watch the entire movie. In applications such as those related to machine vision systems, which use various videos for training, selecting a fixed length of a video is necessary to create a better training sample and to reduce the required training time based on relevant details of the system input [2].

Machine learning methods designed to process videos use different features. In some applications, consistency is not necessary, and the method should only extract spatial features; however, in some applications—for example, in human action recognition (HAR)—frame continuity is critical [3]. Among the research that tries to optimize a video as an

input of machine learning systems, Ref. [4] used keyframes for crowd counting, and, in [5], the goal was to summarize video contents.

One of the growing topics in video processing, which is widely used, is HAR. Among its applications, these can be cited: traffic surveillance, smart city management, hospital management and security systems. Since video processing methods usually require fixed video lengths and frame dimensions, and training videos have variable lengths, choosing keyframes to cover maximum action-related information is highly demanded. A review of video-based human performance detection algorithms revealed that most of these methods use random frame selection throughout the input video [6,7], or a mixture of all processed frames [8], which directly affects the efficiency of the training process and system accuracy, and increases the required training time. Obviously, if the most appropriate frames of a frame sequence are used in the training step, the HAR system will be better trained.

In a typical scene that includes regular actions, there are many frames with very little information due to a lack of movement that can be discarded in the training. On the other hand, HAR systems usually require many samples for training. Another challenge of most HAR systems is the mismatch between the dimensions of the input videos and the system input due to differences in the resolution of the used acquisition cameras. Hence, common video-based HAR systems use methods, such as resizing or cropping frames, to match the different acquisition camera resolutions, which may reduce the system efficiency.

Due to the above explanations, this research proposes an optimal method to select a sequence of keyframes, and then to select the region of interest (ROI) in the selected keyframes from the input video that contains the most relevant information for HAR systems. The selected keyframes can be cropped using the founded ROI, and then can be used in the training step of the HAR system. The proposed method can be used as a pre-processing method in many HAR systems in order to enhance the training efficiencies, both in terms of accuracy and speed.

The organization of the rest of this article is as follows: Section 2 provides a literature review; Section 3 gives a detailed explanation of the theoretical framework and methodology of the proposed method; Section 4 discusses simulation details and results. Finally, the conclusion is given in Section 5.

2. Literature Review

The key goal of the proposed method is the pre-processing of input videos in order to remove unnecessary information at the beginning of the HAR system, so this section is mainly focused on keyframe selection and ROI finding at the input block of HAR systems. The existing methods for keyframe selection can be classified into the following groups: methods based on extracting temporal or spatial features, methods based on deep learning and hybrid methods. In all of these categories, the efficiency of the method is mainly defined based on the application.

The methods based on extracting temporal or spatial features use saliency features, such as edge or motion features, for keyframe selection. Zhenxing et al. [9] used the Laplacian operator to select the appropriate sequences for short periods as the input of a deep learning scheme to identify Hong Kong sign language. Zayed and Rivaz [10] performed elastomeric experiments using ultrasonic images taken from a pressurized mechanical object. The acquired images were first submitted to a multilayer perceptron (MLP) classifier, and any two consecutive frames that contain no relevant information are removed from the list of training frames. They decomposed the displacement into a linear combination of weighted principal components, which were used as an input for the MLP. Kyung and Yang [11] proposed a method for selecting keyframes in RGB-Depth (RGB-D) video tracking systems, which uses the difference of frames and features extracted from the image depth information simultaneously. Lin et al. [12] used the Kanade–Lucas–Tomasi (KLT) algorithm to select keyframes in an automated driving system. In this method, the main features, such as margins, road lines and other obstacles, are extracted from the frames, and then the difference between these features is used to discard or keep the

frames. Rajpal et al. [13] used a fuzzy method based on single-frame information, such as contrast, edges and luminescence, in order to select the best frames for watermarking. Chen et al. [14] suggested a frame selection scheme for video-based person re-identification. The spatial and temporal characteristics are used simultaneously to select keyframes. None of the above research was designed for HAR systems, and none of them can be applied directly to a HAR system, but they have features and concepts that can be used in the development of a suitable method.

The second category includes methods based on deep learning; therefore, these methods use a type of deep learning such as long short-term memory (LSTM) and the convolutional neural network (CNN) for keyframe selection. Xu et al. [15] proposed a method using an autoencoder for selecting the frame sequence of a movie to create an automatic thumbnail for the video, which is vital for, as an example, online video sharing sites that provide a short tag for each movie. Wu et al. [16] proposed a dynamic method to remove unnecessary video information for video recognition. The authors used an LSTM network for selecting frames with the most relevant information. Pretuary and Pillay [17] trained various CNN-based methods, such as ResNet, to select frames to create video thumbnails automatically. Zhao et al. [18] proposed a hybrid visual tracking method based on deep and reinforcement learning. The suggested method only used frames where the object being tracked moved away from its previous location more than a specific threshold; the other frames were detected and removed from the training process. Xiang et al. [19] used the ConvNet network with two different spatial and temporal approaches in order to select keyframes in a HAR system. The Xiang method maintains action consistency and selects frames with more spatial and temporal information as an input. Wu et al. [20] used the LSTM to select keyframes in video recognition. The suggested method increases the training speed, reduces the frame length and improves the network performance. Deep learning methods show good results in keyframe selection in video recognition systems such as HAR, but require a complex and time-consuming training step. At the same time, they need a huge number of labeled training samples, which are usually not available in HAR applications.

The last group is hybrid methods, which use a combination of features and machine learning schemes for keyframe and ROI selection. Fasogbon et al. [21] proposed an inertial measurement unit (IMU) to make a video depth map. The authors used keyframe selection to pick frames in an input video that involves minimal human movement. Usually, a smartphone needs 30 frames per second to create a depth map of the environment, but by selecting keyframes, a depth map can be made using just five frames. Kang [22] proposed a robotic imaging system for selecting the best shot among several portraits. Rahimi et al. [23] selected the minimum number of frames, i.e., keyframes, required in a high resolution (HR) imaging task. Jeyabharathi and Djey [24] extracted the video background using a cut set by selecting keyframes from a sequence of frames. They found patches with a similarity between successive forms in a video, removed frames with less information and preserved keyframes. To select the best frames in a HAR video, Wang et al. [25] counted the moving parts of the human body that form action in each frame. The frames where the number of moving parts or amount of movement in them are low were removed from the training process. Zhou et al. [26] presented a video object segmentation scheme using deep learning that can be used for human detection in HAR systems. The suggested system showed good results in different applications, but its applicability in complex scenes with several humans simultaneously is still unclear. Jagtap et al. [27] proposed two adaptive activation functions to accelerate deep learning method convergence. Jagtap et al. [28,29] showed the applicability and flexibility of the adaptive activation functions in various applications, such as video processing. The adaptive activation functions can be combined by keyframe selection in order to enhance the efficiency of deep-learning-based HAR systems. The hybrid methods are usually limited in terms of applications, and specifically in terms of the types of actions involved in HAR systems.

According to the literature review, previous methods to be used as a pre-processing step in HAR systems need to be redesigned or, at least, demand new examples for retraining. The proposed method selects a predefined number of frames, which should include a relatively complete expression of the human action, regardless of the type of action. After that, the proposed method crops the frames considering the input size of the HAR system and the maximum action information. The main advantage of the proposed method is the improvement of the learning process of any HAR system by pre-processing the input videos without the need for training.

3. System Overview

This section describes the architecture of the proposed method for shortening the video length and modifying video dimensions to optimize the training process of HAR systems. The flowchart of the proposed method is shown in Figure 1. As can be seen, the proposed method consists of two separate main steps that are carried out hierarchically. In the first step, the length of the input video is modified based on the acceptable video length for the deep learning network: the video shortening step. Secondly, the shortened video is processed in terms of information within the frames and cropped to match the size of the system input: the adaptive frame cropping step.

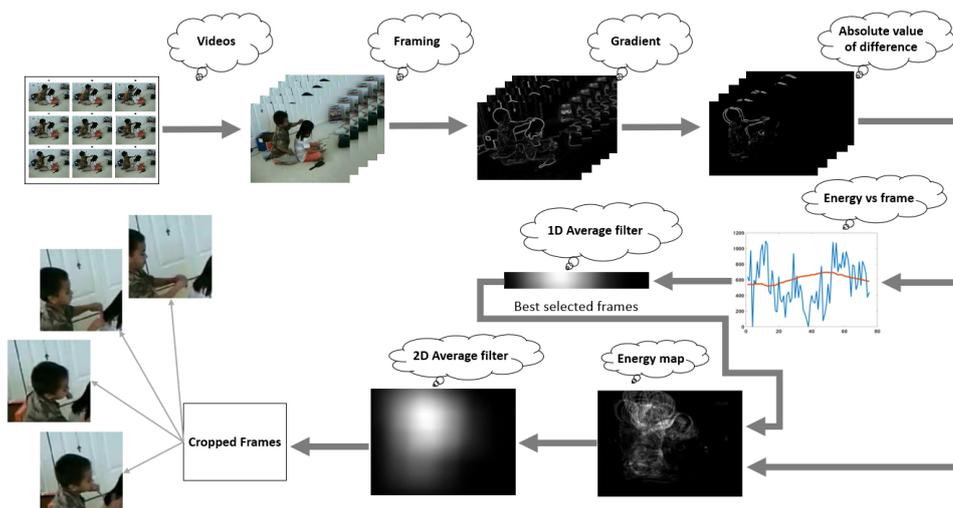


Figure 1. Flowchart of the proposed method.

3.1. Video Shortening

According to previous research in this field, various methods have been used to find keyframes and to select a suitable sequence according to the application. The proposed method does not necessarily need to find the best sequence, but only the frames that convey the relevant information about human actions. The computational speed of the proposed method is an essential feature because many HAR systems work online. Based on these specifications, the proposed method uses a gradient operator to extract the images edges, i.e., the relevant places of each image frame, and then the difference between the edges of consecutive frames, i.e., the movement that is modeled by the difference between the edges of two consecutive frames, is taken into account in order to calculate an approximate estimate of the action information. The gradient difference of the frames can indicate the amount of movement in main locations. Due to the low calculations of the gradient operator relative to the usual motion detection operators, such as optical flow, this approach shows a lower runtime. The pseudo-code of the proposed method is given by Algorithm 1.

Algorithm 1: Gradient Best Frame Selection (Shortening Video)

Input : Input Video V , length of shortened video N_{frames}
 1- Separate Video to frames F_i (i is the number of frame),
 2- Apply Gradient to all frames,
 3- Compute absolute difference between frame gradients $\Delta G_i = |G_{F_i^V} - G_{F_{i-1}^V}|$
 All pixel values lower than P_{Tresh} are assumed to be 0,
 4- Compute energy belong to each difference as $E_{\Delta G_i} = \sum_{g \in \Delta G_i} p_g$ (p is the pixel value)
 Discard frames with energy lower than E_M ,
 5- Calculate sum of remained energies using a sliding window with length N_{frames}
 as $S_{E_i} = \sum_{k=i}^{i+N_{frames}-1} E_{\Delta G_k}$,
 6- Select the maximum S and its index i_S as the start of shortened video,
Output: Merge frames from i_S to $i_S + N_{frames} - 1$ and create the Shortened Video

Firstly, the input video is separated into its frames (F_i). In the next step, the gradient operator extracts the edges of all of the frames G_{F^V} . The gradient operator can use different masks. In the Sobel and Prewitt masks, the gradient of each pixel is a weighted sum of a 3-by-3 neighborhood; the Roberts mask uses a 2-by-2 neighborhood; and the “central difference” and “intermediate difference” masks are column vectors. Figure 2 shows the different gradient masks in the vertical direction; in the horizontal direction, the masks are transposed.

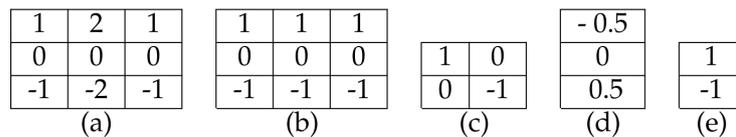


Figure 2. Different Gradient masks: (a) Sobel, (b) Prewitt, (c) Roberts, (d) Central difference, and (e) Intermediate difference.

The results of different gradient masks are presented in the “Result and Discussion” section. The absolute value of the gradient difference of the consecutive frames is used as a fast, relatively low and simple operator in order to calculate the motion information (ΔG_{F^V}).

Figure 3 shows nine different frames of an input video, their gradient and the absolute value of the difference between consecutive frame gradients after normalization. The normalization is carried out based on the maximum value of all differences; in addition, if the gradient difference in a pixel is less than the specified threshold, it is discarded.

Selecting a proper value for P_{tresh} is important. If the gradient of the two consecutive frames difference in a pixel after normalization is less than P_{tresh} , it is assumed that no motion has occurred at that point between the two frames. The effect of P_{tresh} on the accuracy of the HAR system that used the proposed pre-processing method in the input is given in the “Result and Discussion” section. The total energy of each difference is calculated in line 5 of Algorithm 1, and frames with energy lower than E_M are discarded. In the case of Figure 1, the proposed algorithm eliminated frames 2, 5 and 6. In step 6 (the last one) of Algorithm 1, the energy of the remaining frames is added together in a sliding window with length N_{frames} . The window where the summation is maximum is the sequence with maximum motion information. This is because the zero-energy frames have been removed, and the selected window will contain more edge motion information than any other parts of the video. This edge motion, with an acceptable approximation, can include action information. Based on the simulation results, almost all of the videos that were separated from the original human action video in this way had relatively complete information about the involved action. After this step, the frames belonging to the selected window are submitted to the second part of the algorithm.

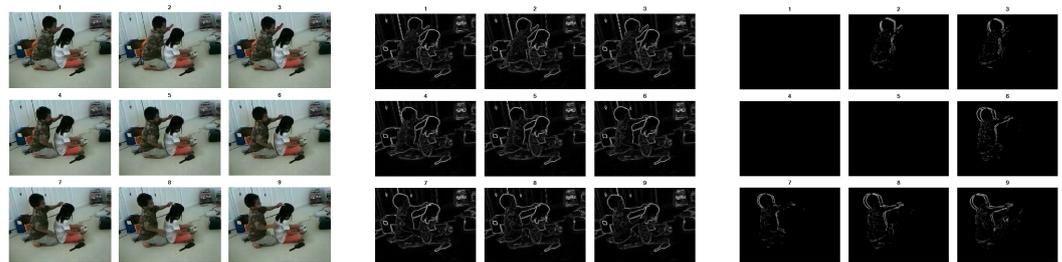


Figure 3. (Left): nine consecutive frames of a video; (center): the gradient of each frame; (right): the absolute value of the difference between the gradient of each frame and the next frame.

3.2. Adaptive Frame Cropping

In the second part of the proposed method, the frames selected in the previous step must be resized or cropped to the input size of the HAR system. In some methods, the input video is resized using conventional image resizing methods. However, in these methods, the performance of the HAR system may be reduced due to the small size of the human image in the resultant frames. The proposed method was designed to estimate the movement region, i.e., the region related to a human action, in the input video with a low computational burden.

Selecting the cropping region randomly is also a weak and too simple approach. On the other hand, the best approach is to identify the human location in the frame and to select the cropped region using human location and action-related features. However, this approach is not desirable due to a high computational burden. In the related step of the proposed method, which is described in Algorithm 2, all of the calculated differences between the normalized frames are transferred from the previous step to this step, which decreases the needed computations, and are used to build an energy map of the shortened video. In this map, the value of each pixel represents the sum of the pixel motion information in the entire frames. After this step, an average filter is applied to the built energy map. The window size of the average filter is defined as equal to the input size of the used HAR system, and the final image is obtained.

Algorithm 2: Adaptive Frame cropping

Input : Shortened Video V , The desired size of final video D_s

1- Separate Video to frames F_i (i is the number of frame),

2- Apply Gradient to all frames G_{F_i} ,

3- Compute absolute difference between frame gradients $\Delta G_i = |G_{F_i} - G_{F_{i-1}}|$,

4- Add all ΔG s together and made an energy map for video, $E_{map} = \sum_{i=1}^{(N_{frames})} \Delta G_i$,

5- Apply Average (or mean) filter with size D_s to E_{map} ,

6- Select the maximum pixel value index i_s of filtered image as the center of cropping area,

7- Crop frames using i_s as center and D_s as crop size,

Output: Merge cropped frames and create the Shortened cropped Video

Finally, the pixel that has the highest value in the filtered image is selected as the center of the cropping region. It is easy to argue that this window contains more relevant motion information, which was found based on the gradient difference between frames, than any other window than can be defined in the video.

Figure 4 shows the overall energy map of a sample shortened video and the result of applying a average filter with dimensions equal to [111,111] on it. Figure 5 shows the original frames, the result of applying the proposed method and the result of a bad random selection. Hence, it is possible to conclude from Figure 5 that a randomly cropped video, in some cases, does not contain valuable information for training HAR systems due to the wrongly cropped regions. Contrary to random selection, the result of the proposed method contains complete information about the involved action.

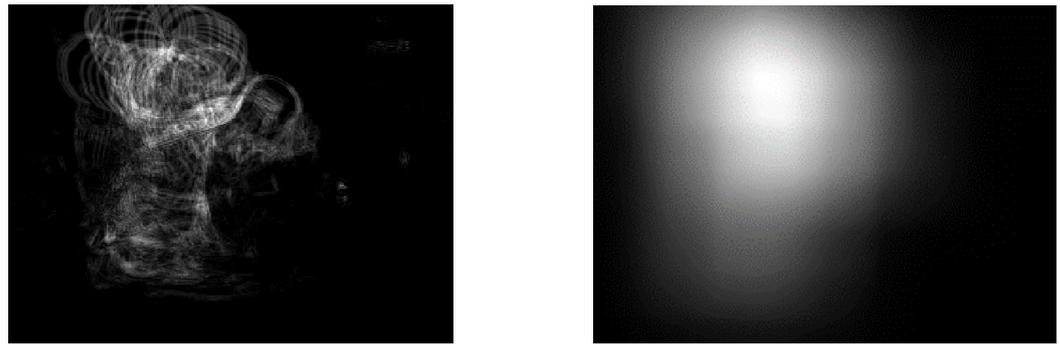


Figure 4. (Left): video energy map; (right): the resulting image after applying the average filter.



Figure 5. (Left): the frames of the shortened video; (center): result obtained by the proposed method; (right): the result of a bad random ROI selection.

4. Result and Discussion

To evaluate the efficiency of the proposed method, it was applied to the input of some current HAR systems. The accuracy obtained by the studied HAR systems before and after the proposed method application indicates its efficiency over other existing methods. Four different HAR methods that used random approaches to select the training input [30–33] were chosen to evaluate the performance of the proposed method. The two UCF101 [34] and HMDB51 [35] public HAR datasets based on [30–33] were used in the evaluation. In addition to accuracy, the execution time of the algorithms is also compared.

4.1. Dataset

The two selected datasets are the UCF101 and HMDB51 datasets. The UCF101 dataset contains 13,320 videos of 101 actions, and the HMDB51 dataset contains 6766 videos of 51 human actions. The length of the videos varies, and all videos involve just one action.

4.2. Implementation Details

All HAR methods [30–33] were trained using random frames and video cropping with static or blind cropping regions. All of these methods were implemented on a system with the following hardware specifications:

- CPU: i7 9700;
- RAM: 16 GB;
- Video Graphics Card: Nvidia RTX 2070 super;

and using the MATLAB 2020 software. A total of 70% of the videos in each dataset were used for training, and the remaining 30% were used as test data. The training and test data were the same in all comparisons. In all implementations, the number of input frames was set to 20, and the video dimensions were set to 111 by 111. The remaining part is the data augmentation used in previous methods, which could not be used in the proposed method due to the optimal selection of frames and cropping regions.

To generate different data without losing optimality, a total of four different time intervals with or without overlap with a difference of at least five frames were selected

based on the computed energy (Part 5 of Algorithm 1, and 200 different cropping regions were selected from the maximum values obtained in Section 3 of Algorithm 2. Hence, 800 quasi-random candidates could be created for each video as data augmentation.

In the first step of the implementation, a constant P_{thresh} equal to 0.1 was assumed, and different gradient masks were tested in the proposed method Part 2 of Algorithm 2. The obtained results are given in Tables 1 and 2. The results show that the Sobel mask is the best choice as the gradient operator for adaptive frame cropping in both datasets. In some cases, especially in the training phase of HMDB51, the other masks showed better accuracy, but in the overall evaluation, due to the test set results, the Sobel mask was found as the best choice.

Table 1. Accuracy of the Train and Test data in the HMDB51 dataset versus Gradient types (D—difference).

Pre-Processing Type		HMDB51					
		None	Gradient Types of Keyframes Selection				
			Sobel	Perwitt	Roberts	Central D	Intermed D
Two-Stream I3D [30]	Train	72.1	76.1	71.5	72.2	70.2	72.9
	Test	65.2	68.5	68.1	67.5	68.4	68.6
Motion Guided Network [31]	Train	72.3	77.4	78.1	75.9	68.8	70
	Test	68	70.3	70	69	68.1	68.4
Spatiotemporal network [32]	Train	67.3	69.8	69.1	70.2	73.4	70
	Test	66.4	67.4	67.4	66.6	66.3	66.1
Correlation net [33]	Train	73	74.7	75.9	76.7	77.8	73.1
	Test	68.1	70.7	70.1	68.4	69.1	68.5

Table 2. Accuracy of the Train and Test data in the UCF101 dataset versus Gradient types (D—difference).

Pre-Processing Type		UCF101					
		None	Gradient Types of Keyframes Selection				
			Sobel	Perwitt	Roberts	Central D	Intermed D
Two-Stream I3D [30]	Train	93.5	95.6	95.7	94.3	95.6	94.3
	Test	92.5	93.1	93.1	92.9	92.9	92.9
Motion Guided Network [31]	Train	96.4	97.2	95.2	96.9	96.7	96.7
	Test	94.1	95.1	94.1	94.5	94.1	93.9
Spatiotemporal network [32]	Train	94.7	97.4	97.4	95.9	96	94.7
	Test	93.8	94.7	94.7	94.0	94.2	94
Correlation net [33]	Train	96	98	95.3	95	94.7	95.5
	Test	92.8	95.2	93.9	93.8	93.6	93.1

In the second step of the implementation, the Sobel mask was chosen as the gradient operator, and the P_{thresh} effect in the proposed method was analyzed (Part 3 of Algorithm 1. The obtained results are presented in Tables 3 and 4, which show that the P_{thresh} value directly affects the accuracy, and that the value of 0.1 was the best candidate. In some cases, especially in the training phase of HMDB51, the other values showed a better accuracy, but in the overall evaluation in both datasets, the value of 0.1 was the best choice. The results obtained with the selected parameters are given in Table 5. In addition, the relative change in the training time was calculated according to:

$$Relative_{Runtime} = \frac{HAR \text{ with Proposed Method Training time}}{HAR \text{ without Proposed Method Training time}} \quad (1)$$

Accordingly, the results of Table 6 show the advantage of the proposed method in terms of $Relative_{Runtime}$.

Table 3. Accuracy of the Train and Test data in HMDB51 dataset in Different P_{thresh} values.

		HMDB51					
Method	Pre-Processing Type	None	Different P_{thresh} Values				
			0.06	0.08	0.1	0.12	0.14
Two-Stream I3D [30]	Train	72.1	75.3	72.8	76.1	72.6	68.2
	Test	65.2	68	68.3	68.5	67.6	67.3
Motion Guided Network [31]	Train	72.3	73.7	71.2	77.4	78	75.2
	Test	68	69	69.3	70.3	69.9	69.8
Spatiotemporal network [32]	Train	67.3	68.2	68.6	69.8	73.3	68.1
	Test	66.4	66.5	66.8	67.4	66.6	65.9
Correlation net [33]	Train	73	75.7	78	74.7	73.8	71.2
	Test	68.1	70.1	70.7	70.7	69.9	69.6

Table 4. Accuracy of the Train and Test data in UCF101 dataset in Different P_{thresh} values.

		UCF101					
Method	Pre-Processing Type	None	Different P_{thresh} Values				
			0.06	0.08	0.1	0.12	0.14
Two-Stream I3D [30]	Train	93.5	93.3	93.9	95.6	93.2	93.6
	Test	92.5	91.2	92.3	93.1	93	92.2
Motion Guided Network [31]	Train	96.4	93.8	94.9	97.2	97.4	95.4
	Test	94.1	92.6	93.6	95.1	95.1	94.4
Spatiotemporal network [32]	Train	94.7	94.3	94.4	97.4	94.7	95.3
	Test	93.8	92.5	93.6	94.7	94.1	93.3
Correlation net [33]	Train	96	97	97	98	95.5	95
	Test	92.8	94.9	95.2	95.2	95	94.6

Table 5. Total accuracy obtained by the HAR systems under study with the tuned proposed method.

Method	Dataset	HMDB51		UCF101	
		Train	Test	Train	Test
Two-Stream I3D [30]	Train	72.1	76.1	93.5	95.6
	Test	65.2	68.5	92.5	93.1
Motion Guided Network [31]	Train	72.3	77.4	96.4	97.2
	Test	68	70.3	94.1	95.1
Spatiotemporal network [32]	Train	67.3	69.8	94.7	97.4
	Test	66.4	67.4	93.8	94.7
Correlation net [33]	Train	73	74.7	96	98
	Test	68.1	70.7	92.8	95.2

The best improvement occurred in the [31] method, where a 7.05% improvement in the training set of the HMDB51 database was achieved. In the UCF101 dataset, the best improvement in the training sets was related to the spatiotemporal network [32], where the system accuracy was improved by 2.85%. According to the results, the improvement in the HMDB51 dataset was more remarkable. Based on the results of Table 6, using the proposed method for pre-processing at the beginning of the HAR systems led, in most cases, to a reduction in the training time, which was due to the elimination of irrelevant inputs in the network training process that sped up the convergence of the training process.

Therefore, it is possible to conclude that the proposed method can be added to the beginning of any HAR system that uses random frame selection in order to improve the accuracy of the final system.

Table 6. *RelativeRuntime* after adding the proposed method to the HAR systems under study.

Method	HMDB51	UCF101
Two-Stream I3D [30]	0.88	0.94
Motion Guided Network [31]	0.94	0.92
Spatiotemporal network [32]	0.92	0.93
Correlation net [33]	0.94	1.02

5. Conclusions

This research proposed a method for selecting the keyframes and suitable regions in a video to increase the speed and accuracy of HAR systems. The proposed method achieves its goals by removing unnecessary data from the video and creating a HAR-compatible input. The proposed hierarchical method identifies the moving areas in a video using the gradient operator, edge extraction and the difference of the gradients between frames, and extracts the frame sequence with more relevant motion information.

The best compatible edge detection method for the proposed method was found using simulations. The threshold value for keyframe selection was found by analyzing its effect on the system accuracy. After this step, a region that includes the most relevant motion information, based on the built motion energy map of the selected frames, is found. The selected frames are then cropped using the founded region, and the final video is used as an input to the HAR system. A high speed, being applicable to all actions and an appropriate approximation in selecting the action area are the main advantages of the proposed method. Finally, the proposed method was combined with several new HAR methods, and it was verified that, by adding its pre-processing to the HAR input, the system accuracy was improved, and its training time decreased.

An interesting research area in HAR systems is the extension of the pre-processing method to remove unnecessary parts from input videos. The unnecessary parts can be defined as unnecessary frames or unnecessary objects in the video scene. Future work can focus on human semantic analysis [36] or human parsing methods [37] to increase the efficiency of the pre-processing method of HAR systems, mainly in their training process.

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.G. and V.H.; writing—review and editing by M.C.F., 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 a result of the project Safe Cities—“Inovação para Construir Cidades Seguras”, with reference POCI-01-0247-FEDER-041435, cofunded 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: Not applicable.

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

References

1. Yang, Y.; Cai, Z.; Yu, Y.; Wu, T.; Lin, L. Human action recognition based on skeleton and convolutional neural network. In Proceedings of the 2019 Photonics & Electromagnetics Research Symposium-Fall (PIERS-Fall), IEEE, Xiamen, China, 17–20 December 2019; pp. 1109–1112.
2. Ji, Y.; Zhan, Y.; Yang, Y.; Xu, X.; Shen, F.; Shen, H.T. A Context knowledge map guided coarse-to-fine action recognition. *IEEE Trans. Image Process.* **2019**, *29*, 2742–2752. [[CrossRef](#)] [[PubMed](#)]
3. Sim, J.; Kasahara, J.Y.L.; Chikushi, S.; Nagatani, K.; Chiba, T.; Chayama, K.; Yamashita, A.; Asama, H. Effects of Video Filters for Learning an Action Recognition Model for Construction Machinery from Simulated Training Data. In Proceedings of the 2021 IEEE/SICE International Symposium on System Integration (SII), Iwaki, Japan, 11–14 January 2021; pp. 12–16.
4. Zhou, Q.; Zhang, J.; Che, L.; Shan, H.; Wang, J.Z. Crowd counting with limited labeling through submodular frame selection. *IEEE Trans. Intell. Transp. Syst.* **2018**, *20*, 1728–1738. [[CrossRef](#)]
5. Ren, J.; Shen, X.; Lin, Z.; Mech, R. Best frame selection in a short video. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Snowmass Village, CO, USA, 2–5 March 2020; pp. 3212–3221.
6. Song, X.; Lan, C.; Zeng, W.; Xing, J.; Sun, X.; Yang, J. Temporal–spatial mapping for action recognition. *IEEE Trans. Circuits Syst. Video Technol.* **2019**, *30*, 748–759. [[CrossRef](#)]
7. 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.
8. Deshpande, 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), IEEE, Pune, India, 5–7 March 2021; pp. 571–576.
9. Zhou, Z.; Lui, K.S.; Tam, V.W.; Lam, E.Y. Applying (3+ 2+ 1) D Residual Neural Network with Frame Selection for Hong Kong Sign Language Recognition. In Proceedings of the 2020 25th International Conference on Pattern Recognition (ICPR), IEEE, Milan, Italy, 10–15 January 2021; pp. 4296–4302.
10. Zayed, A.; Rivaz, H. Fast Strain Estimation and Frame Selection in Ultrasound Elastography using Machine Learning. *IEEE Trans. Ultrason. Ferroelectr. Freq. Control* **2020**, *68*, 406–415. [[CrossRef](#)]
11. Han, K.M.; Kim, Y.J. KeySLAM: Robust RGB-D Camera Tracking Using Adaptive VO and Optimal Key-Frame Selection. *IEEE Robot. Autom. Lett.* **2020**, *5*, 6940–6947. [[CrossRef](#)]
12. Lin, X.; Wang, F.; Guo, L.; Zhang, W. An automatic key-frame selection method for monocular visual odometry of ground vehicle. *IEEE Access* **2019**, *7*, 70742–70754. [[CrossRef](#)]
13. Rajpal, A.; Mishra, A.; Bala, R. A Novel fuzzy frame selection based watermarking scheme for MPEG-4 videos using Bi-directional extreme learning machine. *Appl. Soft Comput.* **2019**, *74*, 603–620. [[CrossRef](#)]
14. Chen, Y.; Huang, T.; Niu, Y.; Ke, X.; Lin, Y. Pose-guided spatial alignment and key frame selection for one-shot video-based person re-identification. *IEEE Access* **2019**, *7*, 78991–79004. [[CrossRef](#)]
15. Xu, Y.; Bai, F.; Shi, Y.; Chen, Q.; Gao, L.; Tian, K.; Zhou, S.; Sun, H. GIF Thumbnails: Attract More Clicks to Your Videos. In Proceedings of the AAAI Conference on Artificial Intelligence, USA, Virtual Conference, 2–9 February 2021; Volume 35, pp. 3074–3082.
16. Wu, Z.; Li, H.; Xiong, C.; Jiang, Y.G.; Davis, L.S. A dynamic frame selection framework for fast video recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**. [[CrossRef](#)]
17. Pretorius, K.; Pillay, N. A Comparative Study of Classifiers for Thumbnail Selection. In Proceedings of the 2020 International Joint Conference on Neural Networks (IJCNN), IEEE, Glasgow, UK, 19–24 July 2020; pp. 1–7.
18. Zhao, K.; Lu, Y.; Zhang, Z.; Wang, W. Adaptive visual tracking based on key frame selection and reinforcement learning. In Proceedings of the 2020 International Workshop on Electronic Communication and Artificial Intelligence (IWECAI), Qingdao, China, 12–14 June 2020; pp. 160–163.
19. Yan, X.; Gilani, S.Z.; Feng, M.; Zhang, L.; Qin, H.; Mian, A. Self-supervised learning to detect key frames in videos. *Sensors* **2020**, *20*, 6941. [[CrossRef](#)] [[PubMed](#)]
20. Wu, Z.; Xiong, C.; Ma, C.Y.; Socher, R.; Davis, L.S. Adaframe: Adaptive frame selection for fast video recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 1278–1287.
21. Fasogbon, P.; Heikkilä, L.; Aksu, E. Frame selection to accelerate Depth from Small Motion on smartphones. In Proceedings of the IECON 2019-45th Annual Conference of the IEEE Industrial Electronics Society, Lisbon, Portugal, 14–17 October 2019; Volume 1, pp. 113–118.
22. Kang, H.; Zhang, J.; Li, H.; Lin, Z.; Rhodes, T.; Benes, B. LeRoP: A learning-based modular robot photography framework. *arXiv* **2019**, arXiv:1911.12470.
23. Rahimi, A.; Moallem, P.; Shahtalebi, K.; Momeni, M. Preserving quality in minimum frame selection within multi-frame super-resolution. *Digit. Signal Process.* **2018**, *72*, 19–43. [[CrossRef](#)]
24. Jayabharathi, D.; Dejeu. Cut set-based dynamic key frame selection and adaptive layer-based background modeling for background subtraction. *J. Vis. Commun. Image Represent.* **2018**, *55*, 434–446. [[CrossRef](#)]
25. Wang, H.; Yuan, C.; Shen, J.; Yang, W.; Ling, H. Action unit detection and key frame selection for human activity prediction. *Neurocomputing* **2018**, *318*, 109–119. [[CrossRef](#)]

26. Zhou, T.; Li, J.; Wang, S.; Tao, R.; Shen, J. Matnet: Motion-attentive transition network for zero-shot video object segmentation. *IEEE Trans. Image Process.* **2020**, *29*, 8326–8338. [[CrossRef](#)] [[PubMed](#)]
27. Jagtap, A.D.; Kawaguchi, K.; Em Karniadakis, G. Locally adaptive activation functions with slope recovery for deep and physics-informed neural networks. *Proc. R. Soc. A* **2020**, *476*, 20200334. [[CrossRef](#)] [[PubMed](#)]
28. Jagtap, A.D.; Shin, Y.; Kawaguchi, K.; Karniadakis, G.E. Deep Kronecker neural networks: A general framework for neural networks with adaptive activation functions. *Neurocomputing* **2022**, *468*, 165–180. [[CrossRef](#)]
29. Jagtap, A.D.; Kawaguchi, K.; Karniadakis, G.E. Adaptive activation functions accelerate convergence in deep and physics-informed neural networks. *J. Comput. Phys.* **2020**, *404*, 109136. [[CrossRef](#)]
30. 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.
31. Zheng, Z.; An, G.; Ruan, Q. Motion Guided Feature-Augmented Network for Action Recognition. In Proceedings of the 2020 15th IEEE International Conference on Signal Processing (ICSP), Beijing, China, 6–9 December 2020; Volume 1, pp. 391–394.
32. Chen, E.; Bai, X.; Gao, L.; Tinega, H.C.; Ding, Y. A spatiotemporal heterogeneous two-stream network for action recognition. *IEEE Access* **2019**, *7*, 57267–57275. [[CrossRef](#)]
33. Yudistira, N.; Kurita, T. Correlation net: Spatiotemporal multimodal deep learning for action recognition. *Signal Process. Image Commun.* **2020**, *82*, 115731. [[CrossRef](#)]
34. Soomro, K.; Zamir, A.R.; Shah, M. UCF101: A dataset of 101 human actions classes from videos in the wild. *arXiv* **2012**, arXiv:1212.0402.
35. 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, IEEE, Barcelona, Spain, 6–13 November 2011; pp. 2556–2563.
36. Zhou, T.; Wang, W.; Liu, S.; Yang, Y.; Van Gool, L. Differentiable Multi-Granularity Human Representation Learning for Instance-Aware Human Semantic Parsing. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 19–25 June 2021; pp. 1622–1631.
37. Zhou, T.; Qi, S.; Wang, W.; Shen, J.; Zhu, S.C. Cascaded parsing of human-object interaction recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *Early Access*. [[CrossRef](#)] [[PubMed](#)]