Towards real-time activity recognition

Shane Reid
School of Computing, Engineering and Intelligent Systems
Ulster University
UK
Reid-S22@ulster.ac.uk

Philip Vance
School of Computing, Engineering and Intelligent Systems
Ulster University
UK
p.vance@ulster.ac.uk

Sonya Coleman
School of Computing, Engineering and Intelligent Systems
Ulster University
UK
sa.coleman@ulster.ac.uk

Dermot Kerr
School of Computing, Engineering and Intelligent Systems
Ulster University
UK
d.kerr@ulster.ac.uk

Siobhan O'Neill
School of Psychology
Ulster University
UK
sm.oneill@ulster.ac.uk

2020 IEEE 4th International Conference on Image Processing, Applications and Systems (IPAS) | 978-1-7281-7574-4/20/$31.00 ©2020 IEEE | DOI: 10.1109/IPAS50080.2020.9334948

***Abstract*—Activity recognition relates to the automatic visual detection and interpretation of human behaviour and is emerging as an active domain of computer vision. It has important applications such as identifying individuals who are at risk of suicide in public locations such as bridges or railway stations. These individuals are known to exhibit easily observable activities and behaviours such as pacing, looking up and down the railway tracks, and leaving objects on the platform. In order to detect these behaviours, an approach to individual person activity recognition is needed which can run in real time and monitor multiple individuals in parallel. We present a method for human activity recognition using skeletal keypoints and investigate how using varying sample rates and sequence lengths impacts accuracy. The results show that for any given sequence length, optimising the sample rate can result in an overall increase in classification accuracy and improvement in run-time. Results demonstrate that finding the optimal time period over which to sample frames is more important than simply decreasing the number of frames sampled. Further, we show that keypoint based activity recognition approaches outperform other state of the art approaches. Finally, we show that this approach is fast enough for real time activity recognition when up to 14 people are present in the image whilst maintaining a high degree of accuracy.**

Keywords—Activity recognition, Video processing, Real time, Keypoints

I. Introduction

Human activity recognition has become increasingly popular due to an interest in the detection of social signals. Social signal processing (SSP) covers a large number of complex computing challenges, such as the development of reliable lie detectors, clinical diagnostic tools and more [1]. An interesting problem within SSP is that of detecting suicidal individuals in the context of bridges and railway platforms; this is directly relatable to standard computer vision activity recognition tasks. A number of studies have shown that individuals who are at risk of jumping exhibit easily observable behaviours beforehand [2]–[5], such as pacing and leaving objects on the platform. A method for real-time detection of these activities could permit intervention and save lives [5].

Human activity recognition has been an open problem in computer vision for over two decades. Most attempts to detect human behaviours using computer vision are either optical flow based [6]–[9] or deep learning based [10]–[13]. While these approaches may be quite accurate, they have a number of drawbacks, in particular they are computationally expensive.

Figure 1 Train platforms can be quite crowded, therefore a fast approach is needed to monitor the activities of all individuals simultaneously

Thus, it may be difficult or even impossible for these algorithms to be implemented in real-time dynamic environments with a large number of people, such as the example given in Figure 1.

The use of interest points to represent activities from sequences of images is one approach which can maintain high accuracy whilst providing a significant reduction in computational cost [14]. These methods work by first extracting a number of fixed interest points on the human body and then tracking the locations of these points over a number of video frames. Early methods for achieving this were based on the use of general feature detectors such as SIFT or SURF. However, the use of these methods faced a number of drawbacks as there was no agreed standard for human representation [15]. To mitigate these problems there has been significant research within computer vision for more specialised "skeletal keypoint" detectors. Rather than returning a large set of key-points within an image, these methods return sets of key-points that directly relate to the specific body parts for each individual within an image. Two of the most successful skeletal key-point approaches are AlphaPose [16] and OpenPose [17] which are based on the use of deep learning neural networks to extract the skeletal keypoints.

The question of determining how many image samples (frames) should be used to accurately classify their activity in an image sequence has previously been explored in the context of optical flow [18]. It was found that calculating optical flow over two consecutive frames was often enough to achieve accuracy of approximately 88% using the KTH dataset. When optical flow was computed over 7 frames, accuracy of 90.9% was achieved. Similarly, the problem of optimal sample rate

978-1-7281-7574-4/20/$31.00 ©2020 IEEE

was explored in [19] where they used four state of the art methods for activity recognition and reduced the number of frames sampled over a fixed period of time. Reducing the sample rate in this way corresponded to a reduction in overall classification accuracy, which seems intuitive as having more data on the individual's movement over a fixed time period should increase prediction accuracy. However, using a reduced sample rate provides benefits such as reducing the computational cost.

In this paper we investigate the effect of reducing the sample rate and the number of frames in the sequence length. In this way we can determine whether it is possible to achieve an increase in classification accuracy, while maintaining or even reducing the amount of data needed to perform classification. To do this we present a general-purpose method for activity recognition using skeletal keypoints, generated with OpenPose [17]. Classification of these keypoint features is performed using a XGBoost classifier [20] which is based on the concept of tree boosting and used extensively in real world applications [20]. The remainder of this paper is organized as follows: Section 2 outlines the methodology used. Section 3 presents experimental results and provides a comparison with other state-of-art methods. In Section 4 we evaluate the speed of this approach on a multi-person dataset and Section 5 concludes the paper.

II. Methodology

The OpenPose library [17] is used for skeletal keypoint extraction as it has a high level of accuracy and low computational cost. However, it should be noted that the contributions of this paper are not dependent on any specific skeletal keypoint estimation approach and can be implemented with any other skeletal keypoint extraction method such as AlphaPose [16], or Megvii [21]. Furthermore, as skeletal keypoint estimation is an open problem in computer vision and faster and more accurate keypoint estimation approaches emerge, the accuracy and speed of these approaches will also improve. However, the issue of frame-rate optimisation still remains in order to optimise classification speed [22]. Regardless of the method used for feature extraction, each individual keypoint may be defined as:

$$k_i = \{x_i, y_i\} \tag{1}$$

where x_i and y_i are the image coordinates of the extracted keypoint. For each frame of each video, the set of skeletal keypoints for each individual are extracted. For a given frame I, the extracted set K_I of γ skeletal keypoints per individual is defined as:

$$K_I = \{k_1, k_2 k_3 \dots k_\gamma\} \tag{2}$$

When using OpenPose 25 skeletal key points are extracted per individual as shown in Figure 2.

In order to investigate the influence of different sample rates and sequence length on the classification accuracy, we represent the temporal component of an activity by constructing an activity feature vector for the activity θ using a concatenated sequence of skeletal keypoints K_I. This is defined as:

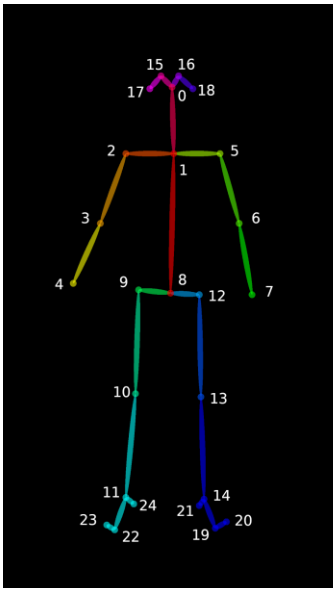

Figure 2 The 25 keypoints extracted by OpenPose

$$\theta = \{K_I, K_{I-m}, K_{I-2m}, \dots K_{I-(n-1)m}\} \tag{3}$$

where n is the sequence length, m is the integer step size, $m \propto \mathit{Sample\ Rate}$ and can be calculated as:

$$m = \frac{\mathit{Video\ Frame\ Rate}}{\mathit{Sample\ Rate}} \tag{4}$$

By varying n and m, we are able to adjust both the length of the feature vector, and the time period over which skeletal keypoints are sampled. This allows us to determine the influence these changes have on the overall classification accuracy. Furthermore, this approach ensures that the computational cost may be kept as low as possible yet still consider overall accuracy in terms of the baseline approach using all the sequence frames. The values for sequence length (number of frames) and step size (sample rate) used in this investigation are detailed in Section 3.

The set of feature vectors θ was then used to train an XGBoost classifier [20], to classify which activity had occurred. We use XGBoost as it is a scalable learning algorithm which has been used to achieve state of the art accuracy for a large number of real-life data science challenges [20][23]. Based on tree boosting, for a given data set **D** of N activity histories each with M features defined as:

$$D = \{\, x_i, y_i\} \, (card(D) = N, x_i \in R^M, y_i \in R) \tag{5}$$

A tree ensemble method φ uses K additive functions to predict the output:

$$\hat{y_i} = \varphi(x_i) = \sum_{k=1}^{K} f_k(x_i), f_k \in F \tag{6}$$

where F is the space of regression trees defined as:

$$F = \{f(x) = w_{q}(x)\}(q: R^{M} \rightarrow T, w \in R^{T}) \quad (7)$$

Here q represents the structure of each tree that maps an activity history to the corresponding leaf index and T is the number of leaves in the tree. Each function f_k corresponds to an independent tree structure q and leaf weights w. Unlike with standard decision trees, which store a category or number on each leaf, regression trees contain a real score on each leaf. We use w_i to represent the score on the i-th leaf. For a given activity history, the decision rules in the trees (given by q) are used to classify it into the leaves. The final prediction is then calculated by summing up the score in the corresponding leaves (given by w). The XGBoost algorithm also incorporates a number of other techniques to further improve classification performance, such as using feature subsampling in order to prevent overfitting [20].

III. Experimental Results

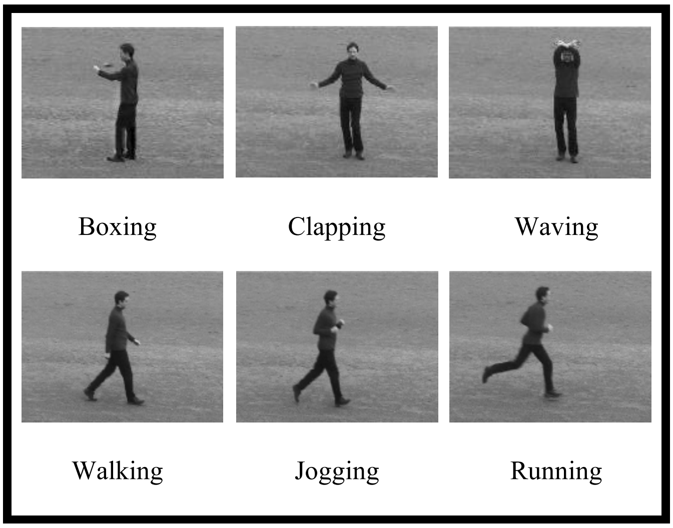

Figure 3 Example frames of the six activities from the KTH dataset

The proposed methodology was used to train and test an activity classifier using the KTH dataset [24] which contains short video clips of 6 distinct human activities: Walking, Jogging Running, Boxing, Hand waving and Hand Clapping. For each activity there are 25 sets of videos, each containing a different individual. Each video set contains four videos of each activity with a different background: outdoors, outdoors with different scale, outdoors with different clothes and indoors. This results in a total of 600 video clips with an average activity length of 4 seconds each. Videos were recorded at 25 frames per second (fps) and a resolution of 160 x 120 pixels. Figure 3 shows example frames from the dataset.

For each frame of each video in the dataset, we use OpenPose to extract a set of skeletal keypoints. These keypoints, from each frame, are then concatenated to construct an activity feature vector as outlined in Section 2. An XGBoost model is trained with the activity vector with the activity type used as the classification label. All presented results are validated using "leave-one-out" 25-fold cross validation [25] where the complete set of activity vectors relating to the videos for one individual are kept for testing and the remaining activity vectors for the other individuals are used for system training. The task is therefore to identify the activity exhibited by an unknown individual, irrespective of the background.

We use the methodology outlined in Section II to investigate how changing the sample rate and number of frames used will impact classification accuracy. To do this we conduct the above experiments using a number of different values for the sample rate and sequence length. The set of sample rates investigated, measured in fps, was as follows:

$$Sample\ Rate = \{25,\ 12.5,\ 8.\overline{3},\ 5,\ 2.5,\ 1.\overline{6}\}.$$

As the original frame rate of the video was 25fps, the corresponding values used for m in (3), calculated using (4), are as follows:

$$m = \{1, 2, 3, 5, 10, 15\}.$$

The sequence length refers to the total *number* of frames subsampled. We use different sequence lengths in order to determine the optimal values. The set of sequence lengths corresponds to the values used for n in (3):

$$Sequence\ Length = \{3, 5, 10, 15\}.$$

Classification accuracy is determined by comparing the predicted activity against the actual activity and represented using a confusion matrix. The set of activity feature vectors for a given activity A is defined as:

$$\mu(A) = \{x \in \Theta, L(x) = A\} \quad (8)$$

where $\boldsymbol{\Theta}$ is the full set of θ_I feature vectors, and $L(x)$ is the true label for activity feature vector x. The percentage of activities predicted as activity B is computed as:

$$\delta(A, B) = \frac{card(\{x \in \mu(A), P(x) = B\})}{card(\mu(A))} \quad (9)$$

where $P(x)$ is the predicted label for activity feature vector x. We also evaluate the average accuracy over all classes for each frame rate and sample size using:

$$Acc = \frac{card(\{x \in \Theta,\ P(x) = L(x)\})}{|\Theta|} \quad (10)$$

The activity time period for each set of results can be calculated using the equation:

$$TimePeriod = \frac{(n-1) \times m}{25} \quad (11)$$

The confusion matrices are presented in Table 4 where results are arranged with the sequence length on the horizontal axis and the sample rate on the vertical axis. Within each matrix the predicted activity is listed on the horizontal axis, with the actual activity listed on the vertical axis. Accuracy is calculated using equation (9).

The highest accuracies were obtained across a range of sequence lengths per activity - there was no one definitive sequence length for all activities. For example, the optimal parameters for classification of the walking activity are 15 frames with a sample rate of $1.\bar{6}$fps. The optimal parameters for the classification of the jogging activity are 15 frames with a sample rate of $8.\bar{3}$fps. Using Equation 11 we can see that these values correspond to an overall activity time period of 8.40 seconds and 1.68 seconds respectively. Similarly, the optimal activity time period for classification of running is 0.56 seconds, classification of boxing is 2.80 seconds, classification of clapping is 3.60 seconds and classification of waving is 1.80 seconds.

There are two variables in this experiment, the sample rate and the sequence length. A decrease in only the sample rate variable will result in an increase in the time period as defined in (11). Given that the classification accuracy for each activity is highest when the sequence length is optimised, correctly selecting the corresponding sample rate can result in classification accuracy improvements. This can be seen clearly in Table 4 with the waving activity. When the sequence length was fixed at 3 frames, sampling at 25fps meant that the overall activity time period was short, at only 0.08 seconds. This resulted in a classification accuracy of 87.21%. When the sample rate was reduced to $1.\bar{6}$fps, the overall activity time period increases to 1.2 seconds resulting in an accuracy of 94.98%. Optimising the sample rate shows similar performance improvements for all other activities.

The results in Table 4 corroborate those found by [18] and [19] who also showed that reducing the number of frames sampled over a given time frame results in a reduction in classification accuracy, and clarifies that sampling over the optimal time frame maximizes classification accuracy for a given sequence length.

Table 1 Average accuracy for each sequence length and sample rate as defined using (10)

		Sequence Lengths / Frames			
		3	5	10	15
Sample Rate/FPS	25	83.10%	85.07%	87.15%	88.38%
	12.5	85.10%	86.85%	88.91%	89.73%
	$8.\bar{3}$	86.29%	87.88%	89.65%	90.19%
	5	87.38%	89.10%	89.97%	90.24%
	2.5	88.76%	89.60%	90.06%	90.14%
	$1.\bar{6}$	88.96%	89.40%	89.67%	89.88%

Table 1 shows the mean accuracies for all activities computed using leave-one-out cross validation using Equation (10), where it can be seen that the overall accuracy increases as the sample rate is decreased, up to an optimal value. The highest classification accuracy of 90.24% was obtained when the overall sequence length was 15 frames, with a sample rate of 5 fps, corresponding to an overall activity time period of 2.8 seconds.

By optimising the sample rate and the sequence length we also optimise the resulting feature vector. Table 1 shows that a sequence length of 15 frames, sampled at a rate of 25fps achieves an accuracy of 88.38% compared with a sequence length of 3 frames with a sample at rate of $1.\bar{6}$ fps which achieves an accuracy of 88.96%.

Table 2 Comparison with other approaches

Accuracy of the KTH dataset with a sample rate of 5fps and sequence length of 15 frames	
Simple keypoint method	91%
Action Snippets [18]	81%
Bag of Visual Words [26]	77%
Dense Trajectories [27]	79%
Motion Interchange pattern [28]	43%

We also directly compare our approach with those presented in [19] using Dense trajectories [27], action snippets [18], Bag of visual words [26] and Motion interchange pattern [28].

In this experiment the frame rate was kept at a constant 5fps and the sequence length was 15 frames for all five approaches. All results use the same experimental setup where 16 subjects are used for training and 9 subjects are used testing (subjects: 2, 3, 5, 6, 7, 8, 9 and 10). The results from this comparison are presented in Table 2 and demonstrate that our skeletal keypoint based approach performs significantly better in terms of classification accuracy using a reduced sample rate of 5fps when compared with the other methods. Even with a reduced amount of data, our approach results in a 10% improvement over other techniques and a total accuracy of 91%.

IV. Multi-person Runtime Evaluation

In order to demonstrate the robustness of this approach, we investigate scalability when classifying the activities of a large number of individuals simultaneously. We present performance accuracy and runtime evaluations using both the KTH dataset, as described in Section 3, and the PNNL parking lot dataset [29] which consists of individuals moving across an empty car park. The first video consists of 14 individuals and is 1000 frames long, and the second consists of 13 individuals and is 1,500 frames long. Both videos have a resolution of 1920x1080, and a frame rate of 29 fps.

We evaluate the runtime and classification accuracy of our simple keypoint approach using a sequence length of 15 frames and a sample rate of 5 fps, as these parameters achieved the best accuracy on the KTH dataset as shown in Section 3. The classifier was first trained on the KTH dataset, and the unseen PNNL dataset was used for testing. If the classifier detected any of the three human locomotion classes (Walking, Jogging or

Running) then this was deemed to be a correct result. Runtimes were calculated on a PC running Ubuntu 18.04 an Intel XeonE5-1620, and Nvidia Titan XP with 16GB RAM. Table 3 presents both the overall classification accuracies and the runtime results from these experiments.

Table 3 Runtime Evaluation

Classification Runtime Evaluation			
Video	No People	Runtime	Accuracy
KTH Dataset	1	24.69 fps	90.24%
Car Park 1	14	7.31 fps	96.27%
Car Park 2	13	8.27 fps	88.69%

These results demonstrate that for single person activity recognition (KTH), this method runs at almost 25fps, which is sufficient for real time applications. The results also demonstrate that for multi-person activity recognition, this method runs at 7-8 fps, with a high degree of accuracy, for up to 14 individuals. Given the video sample rate used was 5fps, the classifier processing rate of over 7 fps demonstrates that the approach is fast enough for real-time multi-person activity recognition. This is significant as, to the authors' knowledge, this is the first approach to perform multi-person activity recognition with this many people in real time [30]. Therefore, for challenging social signal processing problems, such as those discussed in Section 1, these methods could be used to monitor the activities of numerous people in real time.

V. Conclusion

We have investigated how changing sample rate and sample size affects the classification accuracy of a skeletal keypoint method for human activity recognition. Results have shown that reducing the sample rate so that samples are taken over the optimal time period results in improved performance over simply using all available data. Furthermore, we compared this keypoint based method with other state of the art activity recognition approaches at a reduced sample rate and demonstrated that the skeletal keypoint based method is more accurate at a lower frame rate than other existing approaches.

Finally, we evaluated the runtime of this approach on a multi-person dataset and demonstrated that reducing the sample rate in this way enables real time activity recognition for up to fourteen people. This is especially important in contexts such as railway platforms, where there may be many individuals who need to be monitored simultaneously.

Future work will involve using these techniques to investigate SSP tasks such as suicide detection. Furthermore, we suggest investigating methods which build on skeletal keypoint features such as the Euclidean distance and direction of keypoint changes between frames, in order to generate a more accurate classification, especially for activities which appear similar such as walking and running.

V References

[1] J. J. K. Burgoon, N. Magnenat-Thalmann, M. Pantic, and A. Vinciarelli, *Social signal processing*, First Edit. Cambridge University Press, 2017.

[2] B. L. Mishara and C. Bardon, "Systematic review of research

Table 4: Set of confusion matrix for KTH dataset sampled at varying frame rates and sample sizes

			Sequence Length																							
			3						5						10						15					
			Walking	Jogging	Running	Boxing	Clapping	Waving	Walking	Jogging	Running	Boxing	Clapping	Waving	Walking	Jogging	Running	Boxing	Clapping	Waving	Walking	Jogging	Running	Boxing	Clapping	Waving
Sample rate	25	Walking	78.01%	7.94%	11.18%	0.79%	0.13%	0.17%	79.28%	7.35%	10.13%	0.70%	0.12%	0.15%	80.91%	6.67%	8.68%	0.45%	0.17%	0.13%	82.30%	6.39%	7.52%	0.44%	0.20%	0.11%
		Jogging	10.40%	62.36%	21.27%	1.53%	0.21%	0.12%	9.70%	63.75%	21.23%	1.30%	0.21%	0.13%	9.22%	66.01%	21.39%	0.80%	0.23%	0.08%	8.86%	67.01%	20.88%	0.55%	0.25%	0.08%
		Running	9.81%	21.59%	63.48%	0.60%	0.15%	0.09%	9.26%	21.52%	65.02%	0.48%	0.15%	0.09%	8.35%	21.50%	66.24%	0.26%	0.14%	0.06%	7.49%	20.73%	69.85%	0.12%	0.14%	0.04%
		Boxing	1.25%	7.79%	3.42%	95.69%	2.32%	0.63%	1.23%	7.14%	2.92%	96.12%	2.39%	0.60%	0.98%	5.56%	3.02%	97.31%	2.10%	0.58%	0.89%	5.60%	1.44%	97.76%	1.78%	0.36%
		Clapping	0.02%	0.20%	0.35%	0.68%	82.64%	11.78%	0.03%	0.19%	0.33%	0.74%	86.26%	9.09%	0.02%	0.20%	0.38%	0.60%	89.62%	6.98%	0.03%	0.23%	0.11%	0.65%	90.93%	5.78%
		Waving	0.50%	0.11%	0.30%	0.71%	14.55%	87.21%	0.51%	0.05%	0.37%	0.66%	10.87%	89.93%	0.52%	0.06%	0.29%	0.58%	7.74%	92.16%	0.44%	0.04%	0.19%	0.49%	6.70%	93.63%
	12.5	Walking	79.33%	7.44%	10.12%	0.71%	0.11%	0.17%	80.61%	6.74%	9.26%	0.47%	0.16%	0.15%	82.91%	6.13%	6.89%	0.33%	0.23%	0.09%	84.14%	6.03%	7.24%	0.25%	0.21%	0.04%
		Jogging	9.72%	63.47%	22.02%	1.28%	0.20%	0.12%	9.11%	65.51%	21.84%	0.88%	0.24%	0.08%	8.74%	67.84%	21.57%	0.38%	0.21%	0.06%	8.34%	68.61%	22.40%	0.20%	0.17%	0.03%
		Running	9.36%	21.63%	64.42%	0.49%	0.15%	0.09%	8.72%	21.62%	65.13%	0.28%	0.15%	0.06%	7.00%	20.49%	70.28%	0.10%	0.13%	0.04%	6.44%	20.26%	69.52%	0.07%	0.09%	0.03%
		Boxing	1.08%	7.17%	2.81%	96.18%	2.31%	0.61%	1.03%	5.85%	3.15%	97.08%	2.11%	0.56%	0.91%	5.25%	0.96%	98.05%	1.43%	0.31%	0.75%	4.94%	0.56%	98.35%	1.20%	0.22%
		Handclapping	0.02%	0.22%	0.31%	0.70%	86.37%	8.97%	0.02%	0.20%	0.35%	0.67%	89.29%	7.12%	0.02%	0.24%	0.13%	0.70%	91.56%	5.32%	0.03%	0.12%	0.07%	0.74%	92.51%	4.23%
		Handwaving	0.49%	0.07%	0.32%	0.64%	10.86%	90.04%	0.51%	0.08%	0.28%	0.61%	8.05%	92.03%	0.41%	0.06%	0.16%	0.43%	6.45%	94.17%	0.30%	0.05%	0.20%	0.38%	5.81%	95.46%
	8.3̇	Walking	80.07%	7.12%	9.60%	0.58%	0.12%	0.14%	81.46%	6.43%	8.01%	0.43%	0.18%	0.13%	84.05%	5.98%	7.14%	0.28%	0.20%	0.05%	84.96%	6.28%	8.10%	0.22%	0.16%	0.03%
		Jogging	9.24%	64.72%	22.26%	1.04%	0.23%	0.09%	9.09%	66.37%	21.63%	0.63%	0.25%	0.07%	8.29%	68.49%	22.90%	0.25%	0.18%	0.04%	8.01%	69.27%	23.35%	0.22%	0.14%	0.02%
		Running	9.16%	21.54%	65.01%	0.39%	0.14%	0.09%	8.03%	20.96%	68.42%	0.18%	0.14%	0.04%	6.40%	20.60%	69.18%	0.07%	0.09%	0.04%	6.56%	21.01%	67.92%	0.06%	0.08%	0.03%
		Boxing	1.04%	6.32%	2.50%	96.70%	2.18%	0.55%	0.96%	5.91%	1.43%	97.61%	1.74%	0.45%	0.87%	4.73%	0.55%	98.35%	1.14%	0.23%	0.11%	3.24%	0.44%	98.57%	1.04%	0.21%
		Clapping	0.02%	0.22%	0.34%	0.67%	88.46%	7.68%	0.02%	0.23%	0.23%	0.66%	90.54%	6.14%	0.03%	0.16%	0.09%	0.66%	92.52%	4.36%	0.04%	0.08%	0.08%	0.62%	92.99%	3.62%
		Waving	0.48%	0.10%	0.29%	0.63%	8.86%	91.46%	0.43%	0.10%	0.28%	0.50%	7.15%	93.17%	0.36%	0.04%	0.13%	0.39%	5.87%	95.29%	0.31%	0.12%	0.12%	0.32%	5.59%	96.09%
	5	Walking	80.75%	6.64%	9.00%	0.47%	0.17%	0.13%	82.91%	6.16%	7.50%	0.31%	0.23%	0.08%	83.59%	6.44%	8.55%	0.20%	0.15%	0.02%	83.97%	6.74%	7.88%	0.14%	0.11%	0.03%
		Jogging	9.28%	65.42%	22.66%	0.74%	0.23%	0.10%	8.95%	68.12%	21.77%	0.33%	0.23%	0.05%	8.13%	69.15%	23.48%	0.25%	0.14%	0.02%	8.46%	69.13%	22.28%	0.18%	0.15%	0.03%
		Running	8.52%	21.73%	65.62%	0.26%	0.14%	0.07%	6.94%	20.66%	69.60%	0.10%	0.12%	0.05%	6.54%	21.42%	67.31%	0.05%	0.09%	0.04%	6.31%	21.01%	69.26%	0.04%	0.08%	0.03%
		Boxing	0.99%	5.87%	2.09%	97.38%	1.73%	0.47%	0.79%	4.77%	0.90%	98.16%	1.32%	0.29%	1.35%	2.80%	0.45%	98.54%	1.04%	0.21%	0.92%	2.85%	0.37%	98.81%	1.04%	0.28%
		Clapping	0.02%	0.25%	0.30%	0.59%	90.07%	6.21%	0.04%	0.24%	0.06%	0.70%	91.75%	4.92%	0.05%	0.08%	0.10%	0.59%	93.09%	3.45%	0.05%	0.14%	0.11%	0.58%	93.36%	3.43%
		Waving	0.43%	0.08%	0.33%	0.55%	7.66%	93.02%	0.36%	0.05%	0.17%	0.39%	6.36%	94.60%	0.34%	0.11%	0.11%	0.36%	5.49%	96.26%	0.29%	0.12%	0.10%	0.26%	5.26%	96.21%
	2.5	Walking	82.23%	6.34%	8.09%	0.29%	0.22%	0.07%	83.03%	6.68%	9.13%	0.22%	0.14%	0.04%	83.59%	6.56%	8.35%	0.12%	0.11%	0.02%	84.67%	7.14%	8.64%	0.14%	0.11%	0.04%
		Jogging	9.30%	67.64%	22.49%	0.41%	0.22%	0.05%	8.29%	68.68%	23.87%	0.29%	0.14%	0.03%	8.88%	68.63%	22.77%	0.19%	0.14%	0.03%	8.21%	68.76%	23.66%	0.21%	0.14%	0.03%
		Running	7.06%	21.28%	68.41%	0.15%	0.11%	0.07%	6.71%	21.52%	66.19%	0.07%	0.08%	0.06%	6.24%	21.68%	68.17%	0.05%	0.08%	0.05%	5.83%	21.03%	66.92%	0.05%	0.08%	0.05%
		Boxing	0.99%	4.44%	0.81%	98.02%	1.33%	0.32%	1.46%	2.91%	0.62%	98.47%	1.00%	0.16%	0.86%	2.89%	0.52%	98.70%	1.00%	0.27%	0.84%	2.86%	0.58%	98.71%	0.99%	0.26%
		Clapping	0.05%	0.22%	0.06%	0.76%	91.28%	4.92%	0.08%	0.08%	0.10%	0.60%	92.81%	3.87%	0.08%	0.12%	0.09%	0.67%	93.38%	3.44%	0.09%	0.10%	0.09%	0.61%	93.34%	3.43%
		Waving	0.37%	0.08%	0.14%	0.38%	6.84%	94.58%	0.43%	0.13%	0.09%	0.34%	5.83%	95.85%	0.35%	0.12%	0.10%	0.26%	5.29%	96.19%	0.36%	0.10%	0.12%	0.29%	5.34%	96.20%
	1.6	Walking	82.99%	6.47%	9.00%	0.25%	0.16%	0.05%	82.71%	6.98%	8.87%	0.15%	0.08%	0.02%	84.52%	7.37%	9.13%	0.15%	0.09%	0.02%	85.42%	7.25%	7.99%	0.15%	0.11%	0.02%
		Jogging	8.55%	67.96%	24.19%	0.39%	0.15%	0.04%	8.83%	67.85%	23.93%	0.27%	0.16%	0.02%	8.54%	67.36%	23.48%	0.26%	0.16%	0.02%	7.97%	67.96%	24.64%	0.25%	0.15%	0.04%
		Running	6.68%	21.77%	65.98%	0.11%	0.09%	0.05%	6.78%	21.93%	66.54%	0.10%	0.09%	0.04%	5.78%	21.39%	66.62%	0.10%	0.08%	0.05%	5.40%	21.08%	66.53%	0.09%	0.09%	0.05%
		Boxing	1.35%	3.60%	0.55%	98.17%	1.13%	0.27%	1.30%	2.95%	0.44%	98.46%	1.06%	0.28%	0.65%	3.61%	0.61%	98.54%	1.17%	0.20%	0.79%	3.44%	0.63%	98.64%	1.09%	0.26%
		Clapping	0.05%	0.14%	0.12%	0.65%	91.56%	4.61%	0.05%	0.15%	0.10%	0.64%	92.80%	4.18%	0.10%	0.14%	0.05%	0.61%	92.95%	4.14%	0.08%	0.15%	0.06%	0.58%	92.91%	4.04%
		Waving	0.39%	0.07%	0.16%	0.43%	6.91%	94.98%	0.33%	0.14%	0.13%	0.38%	5.81%	95.45%	0.41%	0.12%	0.11%	0.35%	5.56%	95.57%	0.33%	0.13%	0.15%	0.29%	5.66%	95.59%

on railway and urban transit system suicides," *J. Affect. Disord.*, vol. 193, pp. 215–226, 2016, doi: 10.1016/j.jad.2015.12.042.

[3] K. Lukaschek, J. Baumert, and K. H. Ladwig, "Behaviour patterns preceding a railway suicide: Explorative study of German Federal Police officers' experiences," *BMC Public Health*, vol. 11, pp. 0–5, 2011, doi: 10.1186/1471-2458-11-620.

[4] B. Ryan, "Developing a framework of behaviours before suicides at railway locations," *Ergonomics*, vol. 61, no. 5, pp. 605–626, 2018, doi: 10.1080/00140139.2017.1401124.

[5] B. L. Mishara, C. Bardon, and S. Dupont, "Can CCTV identify people in public transit stations who are at risk of attempting suicide? An analysis of CCTV video recordings of attempters and a comparative investigation," *BMC Public Health*, vol. 16, no. 1, pp. 1–10, 2016, doi: 10.1186/s12889-016-3888-x.

[6] A. Ilidrissi and J. K. Tan, "A deep unified framework for suspicious action recognition," *Artif. Life Robot.*, vol. 24, no. 2, pp. 219–224, 2019, doi: 10.1007/s10015-018-0518-y.

[7] M. F. Aslan, A. Durdu, and K. Sabanci, "Human action recognition with bag of visual words using different machine learning methods and hyperparameter optimization," *Neural Comput. Appl.*, vol. 9, pp. 1–13, 2019, doi: 10.1007/s00521-019-04365-9.

[8] M. M. Moussa, E. Hamayed, M. B. Fayek, and H. A. El Nemr, "An enhanced method for human action recognition," *J. Adv. Res.*, vol. 6, no. 2, pp. 163–169, 2015, doi: 10.1016/j.jare.2013.11.007.

[9] X. Ji, Q. Wu, Z. Ju, and Y. Wang, "Study of Human Action Recognition Based on Improved Spatio-temporal Features," vol. 11, no. October, pp. 500–509, 2014, doi: 10.1007/s11633-014-0831-4.

[10] A. G. D'Sa and B. G. Prasad, "An IoT Based Framework For Activity Recognition Using Deep Learning Technique," *ArXiv Prepr.*, 2019, [Online]. Available: http://arxiv.org/abs/1906.07247.

[11] D. G. Lee and S. W. Lee, "Prediction of partially observed human activity based on pre-trained deep representation," *Pattern Recognit.*, vol. 85, pp. 198–206, 2019, doi: 10.1016/j.patcog.2018.08.006.

[12] P. T. Sheeba and S. Murugan, "Fuzzy dragon deep belief neural network for activity recognition using hierarchical skeleton features," *Evol. Intell.*, no. 0123456789, 2019, doi: 10.1007/s12065-019-00245-2.

[13] M. Subedar, R. Krishnan, P. L. Meyer, O. Tickoo, and J. Huang, "Uncertainty-aware Audiovisual Activity Recognition using Deep Bayesian Variational Inference," in *ICCV*, 2019, pp. 6301–6310.

[14] F. Camarena, L. Chang, and M. Gonzalez-Mendoza, "Improving the dense trajectories approach towards efficient recognition of simple human activities," *2019 7th Int. Work. Biometrics Forensics, IWBF 2019*, pp. 1–6, 2019, doi: 10.1109/IWBF.2019.8739244.

[15] J. Sun, Y. Mu, S. Yan, and L. F. Cheong, "Activity recognition using dense long-duration trajectories," *2010 IEEE Int. Conf. Multimed. Expo, ICME 2010*, pp. 322–327, 2010, doi: 10.1109/ICME.2010.5583046.

[16] Y. Xiu, H. Wang, and C. Lu, "Pose Flow : Efficient Online Pose Tracking," in *British Machine Vision Conference*, 2018, pp. 1–12.

[17] Z. Cao, T. Simon, S. E. Wei, and Y. Sheikh, "Realtime multi-person 2D pose estimation using part affinity fields," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 7291–7299, doi: 10.1109/CVPR.2017.143.

[18] K. Schindler and L. Van Gool, "Action Snippets: How many frames does human action recognition require?," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008, pp. 1–8, doi: 10.1109/CVPR.2008.4587730.

[19] F. Harjanto, Z. Wang, S. Lu, A. C. Tsoi, and D. D. Feng, "Investigating the impact of frame rate towards robust human action recognition," *Signal Processing*, vol. 124, pp. 220–232, 2016, doi: 10.1016/j.sigpro.2015.08.006.

[20] C. G. Tianqi Chen, "XGBoost: A scalable Tree Boosting System," in *International Conference on Knowledge Discovery and Data Mining (KDD)*, 2016, pp. 785–794.

[21] Y. Cai *et al.*, "Joint COCO and Mapillary Workshop at ICCV 2019: COCO Keypoint Challenge Track Technical Report: Res-Steps-Net for Multi-Person Pose Estimation," pp. 3–8, 2019.

[22] T. J. Vennila and V. Balamurugan, "A Stochastic Framework for Keyframe Extraction," *Int. Conf. Emerg. Trends Inf. Technol. Eng. ic-ETITE 2020*, 2020, doi: 10.1109/ic-ETITE47903.2020.294.

[23] V. Ayumi, "Pose-based human action recognition with Extreme Gradient Boosting," in *Proceedings - 14th IEEE Student Conference on Research and Development: Advancing Technology for Humanity, SCOReD 2016*, 2016, p. pp 1-5, doi: 10.1109/SCORED.2016.7810099.

[24] C. Schüldt, B. Caputo, C. Sch, and L. Barbara, "Recognizing human actions : A local SVM approach Recognizing Human Actions," *Pattern Recognition, 2004. ICPR 2004. Proc. 17th Int. Conf.*, vol. 3, no. September 2004, pp. 3–7, 2004, doi: 10.1109/ICPR.2004.1334462.

[25] A. Gao, Z., Chen, M. Y., Hauptmann, A. G., & Cai, "Comparing evaluation protocols on the KTH dataset," in *International Workshop on Human Behavior Understanding*, 2010, pp. 88–100.

[26] I. Laptev, M. Marszałek, C. Schmid, and B. Rozenfeld, "Learning realistic human actions from movies," *26th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR*, pp. 0–7, 2008, doi: 10.1109/CVPR.2008.4587756.

[27] H. Wang, A. Kläser, C. Schmid, and C. L. Liu, "Action recognition by dense trajectories," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 3169–3176, 2011, doi: 10.1109/CVPR.2011.5995407.

[28] O. Kliper-gross, Y. Gurovich, T. Hassner, and L. Wolf, "Motion Interchange Patterns for action recognition in unconstrained Videos," in *European confrence on Computer vision (ECCV)*, 2012, pp. 256–269.

[29] G. Shu, A. Dehghan, O. Oreifej, E. Hand, and M. Shah, "Part-based multiple-person tracking with partial occlusion handling," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 1815–1821, 2012, doi: 10.1109/CVPR.2012.6247879.

[30] Q. Li, R. Gravina, Y. Li, S. H. Alsamhi, F. Sun, and G. Fortino, "Multi-user activity recognition: Challenges and opportunities," *Inf. Fusion*, vol. 63, no. June, pp. 121–135, 2020, doi: 10.1016/j.inffus.2020.06.004.