The final publication is available at Springer via http://dx.doi.org/10.1007/978-3-319-16292-8_25

# Continuous Human Action Recognition in Ambient Assisted Living Scenarios

Alexandros Andre Chaaraoui[1] and Francisco Flórez-Revuelta[2]

[1] Department of Computer Technology, University of Alicante,
P.O. Box 99, Alicante E-03080, Spain,
alexandros@dtic.ua.es,
[2] Faculty of Science, Engineering and Computing, Kingston University,
Penrhyn Road, Kingston upon Thames KT1 2EE, UK,
F.Florez@kingston.ac.uk

**Abstract.** Ambient assisted living technologies and services make it possible to help elderly and impaired people and increase their personal autonomy. Specifically, vision-based approaches enable the recognition of human behaviour, which in turn allows to build valuable services upon. However, a main constraint is that these have to be able to work online and in real time. In this work, a human action recognition method based on a bag-of-key-poses model and sequence alignment is extended to support continuous human action recognition. The detection of action zones is proposed to locate the most discriminative segments of an action. For the recognition, a method based on a sliding and growing window approach is presented. Furthermore, an evaluation scheme particularly designed for ambient assisted living scenarios is introduced. Experimental results on two publicly available datasets are provided. These show that the proposed action zones lead to a significant improvement and allow real-time processing.

**Key words:** ambient assisted living, human action recognition, continuous recognition, action zones, real time

## 1 Introduction

Currently, ambient assisted living (AAL) is attracting great interest in public and administration. This is due to the dual challenge our society is facing with an increasing need of assistance for elderly and impaired people and the simultaneous difficulties in containing the budget deficit. AAL can play a key role in this matter, since it enables diverse care and safety services and can extend the independent living at home of the people. Specifically, vision-based technology is of special interest because it allows to provide valuable services from the detection of home accidents to telecare services [1]. To this extent, vision-based human behaviour analysis can be extremely useful in order to detect actions and activities of daily living which are valuable for health-status monitoring.

In human action recognition (HAR), actions like *falling, walking, sitting* and *bending* are recognised. Great advances have been made in order to improve the recognition rate, support multiple views and view-invariant recognition [2, 3] as well as real-time performance [4, 5]. However, it can be observed that HAR has been addressed by classifying short video sequences that contain single actions. Therefore, two strong assumptions have been made: 1) segmented video sequences which only contain a single action each are provided, and 2) all the video sequences necessarily match with one of the learnt action classes. Whereas these assumptions are commonly made in the state of the art and most of the datasets provide such data, these do not hold true in practical situations as in AAL scenarios, but also regarding human-computer interaction, gaming or video surveillance. In people's homes, cameras will provide a continuous video stream which can contain actions at any moment. This leads to continuous human action recognition (CHAR). In other words, an unsegmented video stream has to be analysed in order to detect actions at any point. Another restriction, which comes along with dealing with the raw video stream of the cameras, is that actually these may not record the expected actions. The person could be performing an unknown action, or nothing at all. Therefore, the proposed system needs to be robust enough in order to discard unknown actions that otherwise would result in misclassifications.

In this paper, continuous human action recognition (CHAR) is addressed in order to overcome the aforementioned assumptions. The concept of *action zones* is introduced and a novel method is proposed to detect the most discriminative segments of action sequences. For continuous recognition, a method based on a sliding and growing window technique is presented. Finally, to perform continuous evaluation considering specific constraints of AAL scenarios, a suitable evaluation scheme based on segment analysis and F1 score is proposed. Experimental results on two publicly available datasets are provided.

## 2 Related Work

Determining the relevant segments of a continuous video stream may be trivial for a human, but it certainly involves a great difficulty for an automated computer vision system. This explains why few works deal with the related additional constraints. Some works try to find the boundaries of the actions in order to apply temporal segmentation. These boundaries can be detected based on discontinuities or extremes in acceleration, velocity or curvature [6]. Once the resulting video segments are obtained, sequence-based action recognition can be applied. Such a temporal segmentation is performed in [7], where atomic movements are localised in the video stream based on so-called 'ballistic movements'. These are defined as impulsive movements, which involve a sudden propulsion of the limbs, and rely on the acceleration and deceleration of start and end of the ballistic segments. A trajectory-based motion feature (*i.e.* the popular motion-history images from [8]) is employed along velocity magnitude features based on silhouette transformation, frame differences and optical flow. Two approaches

are tested for the temporal segmentation. The first proposal handles alignment of the optical flow direction by means of dynamic programming. Whereas in the second, assuming that boundaries are characterised by zero velocity, movement begin-end detection is performed with a boosting based classifier. The first option performed better, since it does not classify specific temporal moments, but aligns a globally optimal segmentation taking into account movement direction. In [9], start and end key frames of actions are identified. Segmentation is performed based on the posterior probability of model matching considering recognition rounds. Depending on the accumulated probability, rounds are ended if a threshold is reached. Adjacent rounds classified as the same action classes are connected into a single segment. Lu *et al.* deal with temporal segmentation of successive actions in [10]. During the learning, only a few characteristic frames are selected based on change, which leads to an outstanding temporal performance of the recognition. Likelihood of action segments is computed considering pair-wise representations of characteristic frames. Although good results are obtained, no further instructions are provided on how an actual continuous video stream would be handled.

A very popular technique in video and audio processing is the sliding window approach. Sliding windows allow to analyse different overlapping segments of the stream in order to isolate a region of interest and then perform classification comparing the window to a set of training templates. If a variable size is also considered, both window position and size dynamically change so that all necessary locations and scales are taken into account. Some works have applied the sliding window technique to CHAR [8, 11, 12]. In [13], a sliding window is employed to accumulate and smooth the frame-wise predictions of a frame-based low-latency recognition. Low-latency CHAR is also considered in [14], where so-called *action points* are proposed as natural temporal anchors. These are especially useful for gaming. Two approaches are proposed. The first relies on a continuous observation hidden Markov model (HMM) with firing states that detect action points. And the second employs a direct classification based on random forests classifiers and sliding window. In conclusion, by means of sliding window techniques, the temporal segmentation is simplified, since no specific boundaries have to be detected. However, due to its computational cost it may only be used if the applied segment analysis can be performed very efficiently.

## 3 Human Action Recognition Method

As it has been previously mentioned, this work builds upon prior contributions in which HAR has been successfully performed for action sequences that have been segmented beforehand. Since in this work these contributions are extended to support continuous recognition, this section provides a brief summary of the related previous publications.

For pose representation a silhouette-based approach has been chosen due to its rich spatial information and low computational requirements. More specifically, a feature representation based on the distance between the contour points

and the centroid of the human silhouette is employed. Furthermore, spatial alignment and a significant dimensionality reduction are performed to obtain a low-dimensional and noise-reduced feature (see [15] for greater detail).

Based on the method published in [16], the most representative feature representations involved in each action class (the so-called key poses) are obtained based on a clustering algorithm, and a bag-of-key-poses model is generated. In order to complement this spatial information related to the human posture, temporal cues are considered by means of modelling the evolution of the human silhouettes along the action sequences. To extract this kind of information, sequences of key poses are learned. These, in turn, are employed for action recognition, where temporal alignment of sequences is performed for matching using dynamic time warping (DTW). Also multi-view recognition is taken into account [5]. Concretely, intelligent feature fusion of single-view feature representations is performed with a feature concatenation operator in addition to a weighted feature fusion scheme that is based on *a priori* knowledge about the usefulness of each camera.

## 4 Learning of Action Zones

It can be observed that the method presented in section 3 is clearly based on segmented recognition since it performs spatio-temporal matching of action sequences. Nevertheless, its accurate recognition and outstanding temporal performance led us to extend it for continuous scenarios. The first issue that has to be addressed is the existence of misclassifications. Action sequences may contain irrelevant segments which are common among actions and therefore ambiguous for classification. For this reason, we propose to extract *action zones*.

**Definition 4.1** *Action zones correspond to the most discriminative segments with respect to the other action classes in the course of an action.*

Based on definition 4.1, for instance, the *fall* action contains an action zone corresponding to the segment from where the body is partially bent, until it is completely collapsed. In other words, the part where the person is standing still is ignored as well as the part where the person is lying on the floor, since these are not discriminative with respect to other actions. In this way, the most relevant segments can be identified in order to ease the differentiation between actions. Furthermore, action zones are shorter than the original sequences. For this reason, the matching time will be significantly reduced. Since the underlying HAR method also presents a very low computational cost, a sliding window approach may be employed without prohibitively reducing the temporal performance.

Initially, the same learning is performed as detailed in section 3. Since segmented sequences are still needed for the learning process, these can easily be obtained relying on the frame-wise ground truth and discarding the segments where no action is performed. Action zones may be located at different parts of the actions depending on the type of action and how the action ground truth has

been labelled. However, based on the provided definition, action zones can be detected automatically by analysing the transition of key poses. For this purpose, we first compute the discrimination value of each key pose $w_{kp}$. All available pose representations are matched with their nearest neighbour among the bag of key poses and the ratio of within-class matches is obtained ($w_{kp} = \frac{matches_{kp}}{assignments_{kp}}$). Therefore, this value indicates how valuable a key pose is for distinguishing action classes. In this way, based on the transition of key poses and their discriminative value, our action zones, *i.e.* the most discriminative segments, can be detected.

Specifically, for each training sequence of action class $a$ and a specific temporal instant $t$, the following steps are taken for the corresponding frame:

1. The feature representation $\bar{V}(t)$ of the current frame is matched with the key poses of the bag-of-key-poses model. For each action class $a$, the nearest neighbour key pose $kp_a(t)$ is obtained.
2. For the $A$ action classes, the raw class evidence values $H_{raw_1}(t), H_{raw_2}(t), ..., H_{raw_A}(t)$ are computed based on the ratio between the discrimination value $w_{kp_a(t)}$ and the distance $dist_{kp_a(t)}$, where $dist_{kp_a(t)}$ denotes the Euclidean distance between the pose representation and the matched key pose $kp_a(t)$. Hence, the discrimination value will be taken into account depending on how well the key pose defines the current pose.

$$H_{raw_a}(t) = \frac{w_{kp_a(t)}}{dist_{kp_a(t)}}, \quad \forall a \in [1...A] \ . \tag{1}$$

3. Normalisation is applied with respect to he highest value observed:

$$H_{norm_a}(t) = \frac{H_{raw_a}(t)}{H_{raw_{max}}(t)}, \quad \forall a \in [1...A] \ . \tag{2}$$

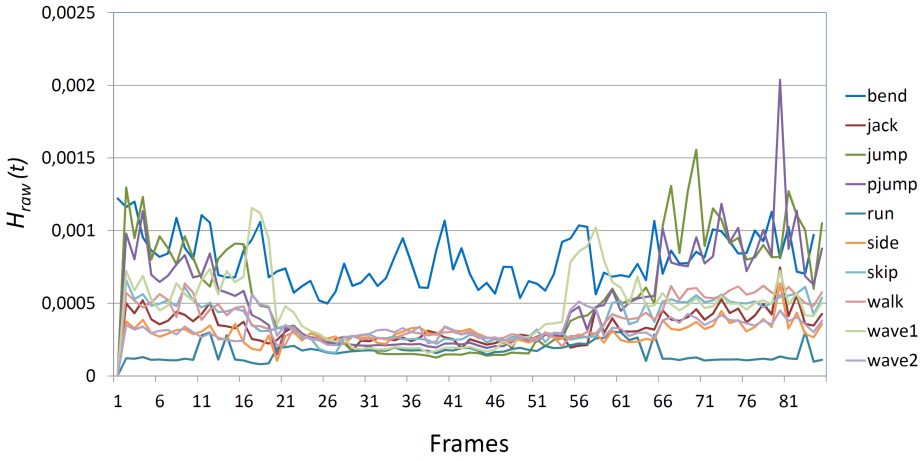
4. Gaussian smoothing is performed centred in the current frame, considering only the frames from a temporal instant $u \leq t$. In this way, we do not take into account future frames, as this would require to delay the recognition for a constant time interval. Convolution is applied to the history $H_{norm}(u)$ values with a Gaussian filter kernel in order to generate $H_{smooth}(t)$. Discrete kernel values are processed based on approximating the continuous values (see [17]):

$$G(u) = \frac{1}{\sigma\sqrt{2\pi}} \, e^{\frac{-(u-\mu)^2}{2\sigma^2}}, \quad / \quad u \leq t \ . \tag{3}$$

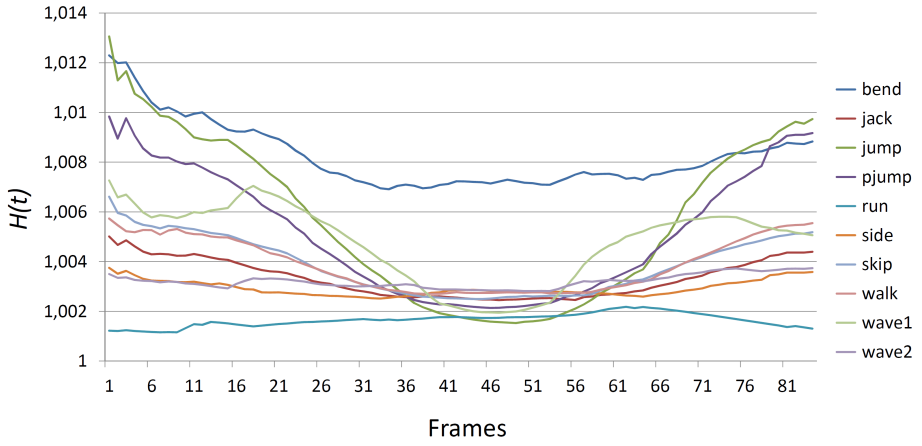
5. Attenuating the resulting value, the final class evidence $H(t)$ is obtained:

$$H_a(t) = e^{10H_{smooth_a}(t)}, \quad \forall a \in [1...A] \ . \tag{4}$$

Figure 1 shows the $H(t)$ evidence values that have been obtained over the course of a *bend* action. In comparison to the raw values, here outliers have been filtered and the differences between classes have become more pronounced. As it can be seen, the evidence of the *bend* class is significantly higher than the others in the central part of the sequence. This is due to the fact that the person is initially standing still. He or she then bends down and, finally,




(a) Raw values



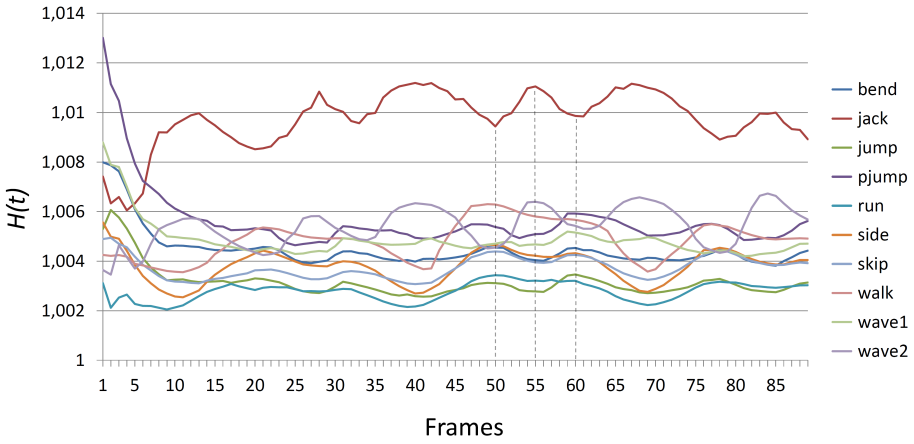

(b) Processed values

**Fig. 1.** Evidence values of each action class before and after processing are shown for a *bend* sequence of the Weizmann dataset.

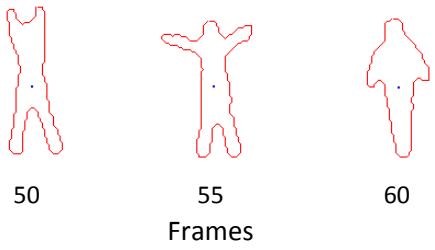
returns to the initial position. The segment that corresponds to the poses in which the person is bent down is the most discriminative one. The poses of this segment match with the most discriminative key poses of the *bend* action class, whereas the ratio between discrimination value and distance is lower for the other classes. For this reason, action zones can be detected by defining the thresholds $HT_1(t), HT_2(t), ..., HT_A(t)$ that have to be reached by the class evidence values of these segments. Specifically, an action zone will be collected from the frame on where:

$$H_{action}(t) > H_{median}(t) + HT_{action} \ , \tag{5}$$

where *action* corresponds to the action class of the current sequence and $H_{median}(t)$ indicates the median value out of $H_1(t), H_2(t), ..., H_A(t)$. An action zone will end if this condition ceases to be met. The median value is employed because the expected peak of $H_{action}(t)$ would influence the average. Moreover, this approach also works if action segments present a high evidence value for more than one action class, which may happen for very similar actions. A second example is shown in figure 2, where the class evidence values that have been obtained for the cyclic *jumping jack* action are detailed. Several short action zones could be found choosing the appropriate threshold $HT_a$. It can also be seen that the peaks correspond to the discriminative segments in which the limbs are outstretched.




(a) Evidence values




(b) Corresponding silhouettes

**Fig. 2.** Evidence values $H(t)$ of each action class and the corresponding silhouettes of one of the peaks of evidence are shown for a *jumping jack* sequence of the Weizmann dataset.

## 5 Continuous Recognition

In this proposal, continuous human action recognition is performed by detecting and classifying action zones. For the continuous recognition of the incoming multi-view data, a sliding window technique is employed. More specifically, a sliding and growing window is used to process the continuous stream at different overlapping locations and scales. At this point, a *null class* has to be considered in order to discard unknown actions and avoid false positives. This class corresponds to all the behaviours that may be observed and have not been modelled during the learning.

Algorithm 1 details the process: The sliding and growing window grows $\delta$ frames in each iteration and slides $\gamma$ frames if the window has reached its maximal length $length_{max}$. If at least $length_{min}$ frames have been collected, the segment of the video stream (or video streams if available) $S$ that corresponds to the window is compared to the known action zones. The best match is obtained by matching the segments of key poses using DTW. Then, a threshold value $DT_a$ is taken into account in order to trigger the recognition. This value $DT_a$ indicates the highest allowed distance in a per-frame basis. In this way, only segments which match well enough with an action zone are classified. Eventually, the unrecognised frames will be discarded and considered to belong to the *null class*.

## 6 Experimentation

### 6.1 Parametrisation

Special consideration has been given to the parameters $HT_1, HT_2, ..., HT_A$ and $DT_1, DT2, ..., DT_A$. The first ones define the threshold that has to be surpassed by the class evidence $H_{action}(t)$ in comparison to the $H_{median}(t)$ value. Different values are admitted for each action class, since the class evidence behaves differently for each type of action. In the case of the second set of parameters, each action class is considered to require a specific similarity between sequence segments and action zones in order to confirm the match as a recognition and avoid false positives for 'poor matches'. This leads us to two sets of $A$ parameters that are difficult to establish empirically, as exhausting tests are unaffordable.

Among the possible search heuristics, evolutionary algorithms stand out since they are proficient for scenarios where the shape of the solution space is unknown and this hinders the election of optimal algorithms. They can also deal with a large number of parameters in a moderate run time. Moreover, relying on a coevolutionary-based approach the intrinsic relationship between our two parameter sets can be considered. For this reason, a technique based on the cooperative coevolutionary algorithm from [18] has been employed for parameter selection. By means of this method, the best performing combination of $HT$ and $DT$ values can be found.

**Algorithm 1** Continuous recognition: sliding and growing window

Let $\delta$ denote the number of frames the window grows in each step.
Let $\gamma$ denote the number of frames the windows moves when slid.
Let $S$ denote the video stream.

$start = 0$
$length = 0$

**repeat**
  ——————— Sliding and growing window ———————-
  $length = length + \delta$

  **if** $length > length_{max}$ **then**
    **Discard** $\gamma$ frames considered to belong to the *null class*
    $start = start + \gamma$
    $length = length - \gamma$
  **end if**
  ——————— Compare to action zones ———————
  **if** $length \geq length_{min}$ **then**
    $dist_{min} = max\_value$
    **for each** $action\_class \in training\_set$ **do**
      **for each** $action\_zone \in action\_class$ **do**
        $dist = d_{DTW}(action\_zone, S[start : start + length])$
        **if** $dist < dist_{min}$ **then**
          $dist_{min} = dist$
          $a = action\_class$
        **end if**
      **end for**
    **end for**
    ——————— Recognise or continue ———————
    **if** $dist_{min} \leq length \times DT_a$ **then**
      **Recognise** segment $S[start : start + length]$
      as action class $a$
      $start = start + length$
      $length = 0$
    **end if**
  **end if**
**until** end of stream or forever

### 6.2 Continuous evaluation

For action recognition based on segmented sequences, the evaluation scheme is straightforward. Since the ground truth label of each sequence is known, the ratio of correctly classified sequences in the test is commonly used as accuracy score. Nevertheless, for continuous evaluation, several new constraints appear. Depending on the application scenario, one might be interested in the number of repetitions of each action. This happens in gaming (*e.g.* three *punches*),

whereas in video surveillance the fact that the action happened is more relevant (*e.g. punching*). In AAL, it is especially important not to miss any actions, because this could result in safety issues (*e.g. falling down*). A delay of a few seconds may be acceptable if this improves the recognition avoiding false negatives. As a result, the applied evaluation scheme varies between authors.

A common option is to apply frame-by-frame evaluation as in [10], but the reliability of this approach is arguable. This is due to the lack of correlation between frames and actions. It could happen that only a few last frames of an action are not recognised correctly. This would result in a high frame-by-frame recognition rate (*e.g.* 90%), although only one correct class label and one or more incorrect predictions have been returned by the system. This means that 50% or more of the returned labels were erroneous. For this reason, other levels of evaluation have been proposed, such as event analysis, where only the activity occurrence and order is considered, or the hybrid segment analysis [19]. In this last approach, a segment is defined as "an interval of maximal duration in which both the ground truth and the predicted activities are constant". In this way, despite the fact that segments may have different durations, alignment is given since each ground truth or prediction change leads to a new unit of evaluation.

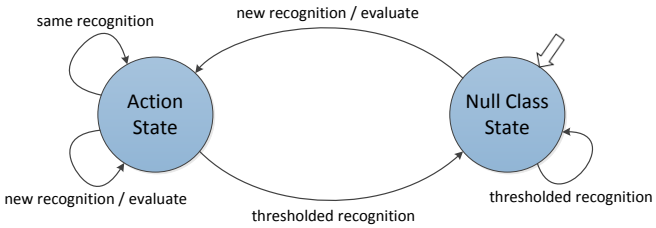



**Fig. 3.** This finite-state machine details the logic behaviour of the applied segment analysis.

This last level of analysis has been employed in this work, as it provides a clear way to align the recognitions with the ground truth and avoids the disadvantages of the frame-based analysis. Figure 3 shows how the *null class* has been considered in the segment analysis. As it can be seen, only new recognitions (*i.e.* different from the last predicted action class) are taken into account for evaluation. The thresholded recognitions are retained and their segments are considered to belong to the *null class*. In addition, recognitions are accepted for a delay of $\tau$ frames after the ground truth indicated the end of the action. Note that this is only allowed if no prediction was given until that moment, *i.e.* the *null class* state was active since the action started and until the delayed recognition has occurred. Otherwise, the action would have already been classified (correctly or wrongly).

In view of the multi-class classification that is performed and that now a *null class* has to be contemplated, results are measured in terms of true positives (TP), false positives (FP), true negatives (TN) and false negatives (FN). These values are accumulated along a cross validation test. A leave-one-actor-out cross validation (LOAO) is proposed in which each sequence includes several continuously performed actions of an actor. In order to consider both precision and recall rates, the $F_1$-measure is used as follows:

$$F_1 = 2 \times \frac{precision \times recall}{precision + recall} \tag{6}$$

$$precision = \frac{TP}{TP + FP} \tag{7}$$

$$recall = \frac{TP}{TP + FN} \tag{8}$$

### 6.3 Results

Our approach has been validated on the multi-view INRIA XMAS (IXMAS) [20] dataset and the single-view Weizmann [21] dataset. The former provides continuous multi-view sequences of different actions performed by the same actor, whereas the latter provides segmented single-view sequences. In order to support continuous recognition, the sequences of the same actor are concatenated into a single continuous sequence. Consequently, unnatural transitions are created due to the gaps of information. Nevertheless, tests have been performed on this dataset for illustrative purposes so that a comparison with other approaches can be made.

With regard to the introduced parameters, the following values have been used during the experimentation (these have been chosen based on experimentation):

1. The threshold parameters have been established by the coevolutionary parameter selection algorithm as follows: $HT \in [0.05, 1.5]$ and $DT \in [0.002, 0.02]$. In figure 4, the class evidence values of a sample sequence can be seen, where the action zone that has been obtained using these $HT$ class evidence thresholds is highlighted.
2. The Gaussian smoothing applied to the $H(t)$ class evidence considers $\sigma = 10.486$ frames. Since approximate discrete values are applied for the convolution, a total of 22 history frames are taken into account and the rest is considered zero.
3. Regarding the sliding and growing window, in each iteration the window grows 5 frames ($\delta = 5$), and when the maximal length $length_{max}$ is reached, the window slides 10 frames ($\gamma = 10$).
4. A delayed recognition is accepted within a period of 60 frames, corresponding to approximately 2 seconds ($\tau = 60$). This time interval has been considered acceptable for this AAL application.

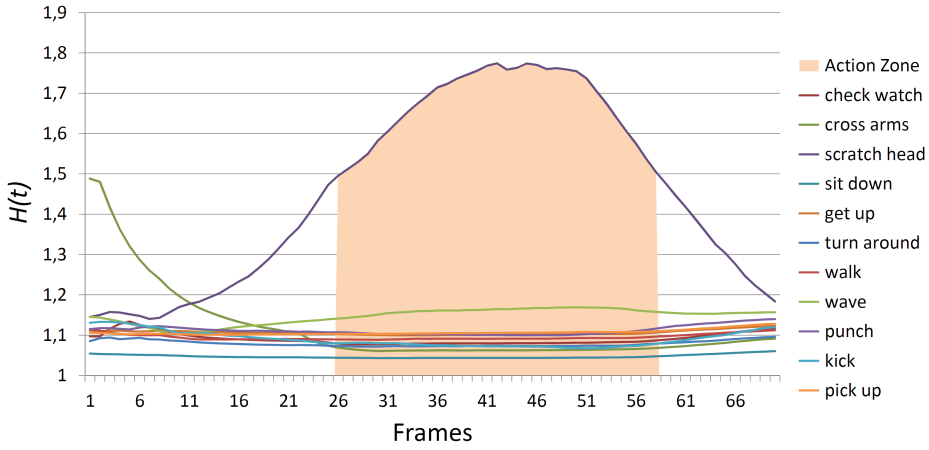



**Fig. 4.** Evidence values $H(t)$ of each action class and the detected action zone are shown for a *scratch head* sequence of the IXMAS dataset.

**Table 1.** Obtained results applying CHAR and segment analysis evaluation over a LOAO cross validation test. Results are detailed using the proposed approaches based on action zones (1) or segmented sequences (2).

| **Dataset** | **Approach** | $length_{min}$ | $length_{max}$ | $F_1$ |
|---|---|---|---|---|
| *IXMAS* | 1 | 3 | 30 | 0.705 |
| *IXMAS* | 2 | 10 | 120 | 0.504 |
| *Weizmann* | 1 | 3 | 20 | 0.928 |
| *Weizmann* | 2 | 10 | 120 | 0.693 |

Table 1 shows the scores that have been achieved by our approach over the ideal value $F_1$-measure of 1.0. The IXMAS dataset presents several known difficulties as view invariance and noise which explain the score difference. Furthermore, the segments labelled as *null class* in which 'other actions' are performed can easily lead to an increase of false positives. In order to show the benefit gained from the action zones approach (approach 1), tests have also been performed using the entire segmented sequences instead (approach 2). In this way, larger segments are considered by the sliding and growing window and these are compared to the original action sequences provided by the ground truth. It can be seen that the proposed continuous recognition based on action zones provides a substantial performance increase and leads to higher scores in general.

Comparison with other state-of-the-art works is difficult in CHAR, due to different evaluation schemes. In [10], frame analysis is employed and 81.0% accuracy is reported on the IXMAS dataset. In the case of the Weizmann dataset, for example in [9], CHAR is performed and a score of 97.8% is reached. Segment analysis is employed in this case, although the rate of correctly classified segments is computed based on a 60% overlap with the ground truth.

The temporal performance has also been evaluated for this continuous approach. While the sliding and growing window technique is computationally demanding, this is offset by the proposed action zones. The short lengths of both action zones and temporal windows make the comparisons between them very efficient. Using a PC with an Intel Core 2 Duo CPU at 3.0 GHz and Windows x64, a rate of 196 frames per second (FPS) has been measured on the Weizmann dataset including all necessary processing stages.

## 7 Discussion and Conclusion

In this work, a method for segmented human action recognition has been extended to support continuous human action recognition. Improvements have been made at the learning and recognition stages. The concept of action zones has been introduced to define and automatically learn the most discriminative segments of action performances. Relying on these action zones, recognition can be carried out by finding the equivalent segments that clearly define the action that is being performed. For this purpose, a sliding and growing window approach has been employed. Finally, segment analysis is used introducing special considerations for the specific AAL application of our work. Tests have been performed relying on the whole segmented sequences or only on the action zones, and significant differences can be seen. By means of action zones, higher accuracy scores are obtained. Real-time suitability of this continuous approach has also been verified. This is indispensable for most of the possible applications, and a necessary premise for online recognition.

In future works, further evaluation should be applied to ease the comparison to other approaches. It could be useful to implement other state-of-the-art techniques and test them in the same conditions as our proposal. Furthermore, a consensus should be reached about the appropriate evaluation schemes. It has also been observed that regarding CHAR, there is a lack of suitable benchmarks including foreground segmentations or depth data. Therefore, new datasets should be created along the corresponding evaluation schemes.

## References

1. Cardinaux, F., Bhowmik, D., Abhayaratne, C., Hawley, M.S.: Video based technology for ambient assisted living: A review of the literature. J. Ambient Intell. Smart Environ. **3**(3) (August 2011) 253–269
2. Poppe, R.: A survey on vision-based human action recognition. Image and Vision Computing **28**(6) (2010) 976 – 990
3. Aggarwal, J., Ryoo, M.: Human activity analysis: A review. ACM Computing Surveys **43**(3) (2011) 16:1 – 16:43
4. Shotton, J., Fitzgibbon, A.W., Cook, M., Sharp, T., Finocchio, M., Moore, R., Kipman, A., Blake, A.: Real-time human pose recognition in parts from single depth images. In: IEEE Conference on Computer Vision and Pattern Recognition, 2011. CVPR 2011. (2011) 1297 – 1304

5. Chaaraoui, A.A., Padilla-López, J.R., Ferrández-Pastor, F.J., Nieto-Hidalgo, M., Flórez-Revuelta, F.: A vision-based system for intelligent monitoring: Human behaviour analysis and privacy by context. Sensors **14**(5) (2014) 8895–8925
6. Kellokumpu, V.P.: Vision-based human motion description and recognition. PhD thesis, University of Oulu, Faculty of Technology, Department of Computer Science and Engineering (2011)
7. Vitaladevuni, S., Kellokumpu, V., Davis, L.: Action recognition using ballistic dynamics. In: IEEE Conference on Computer Vision and Pattern Recognition, 2008. CVPR 2008. (2008) 1 – 8
8. Bobick, A., Davis, J.: The recognition of human movement using temporal templates. IEEE Transactions on Pattern Analysis and Machine Intelligence **23**(3) (2001) 257 – 267
9. Guo, P., Miao, Z., Shen, Y., Xu, W., Zhang, D.: Continuous human action recognition in real time. Multimedia Tools and Applications (2012) 1 – 18
10. Lu, G., Kudo, M., Toyama, J.: Temporal segmentation and assignment of successive actions in a long-term video. Pattern Recognition Letters **34**(15) (2013) 1936 – 1944 Smart Approaches for Human Action Recognition.
11. Hu, Y., Cao, L., Lv, F., Yan, S., Gong, Y., Huang, T.: Action detection in complex scenes with spatial and temporal ambiguities. In: IEEE 12th International Conference on Computer Vision, 2009. ICCV 2009. (2009) 128 – 135
12. Kavi, R., Kulathumani, V.: Real-time recognition of action sequences using a distributed video sensor network. Journal of Sensor and Actuator Networks **2**(3) (2013) 486 – 508
13. Bloom, V., Argyriou, V., Makris, D.: Dynamic feature selection for online action recognition. In Salah, A., Hung, H., Aran, O., Gunes, H., eds.: Human Behavior Understanding. Volume 8212 of Lecture Notes in Computer Science. Springer International Publishing (2013) 64 – 76
14. Nowozin, S., Shotton, J.: Action points: A representation for low-latency online human action recognition. Technical report, Microsoft Research Cambridge (2012) Technical Report MSR- TR-2012-68.
15. Chaaraoui, A.A., Flórez-Revuelta, F.: Human action recognition optimization based on evolutionary feature subset selection. In: Proceeding of the fifteenth annual conference on Genetic and evolutionary computation conference. GECCO '13, New York, NY, USA, ACM (2013) 1229–1236
16. Chaaraoui, A.A., Climent-Pérez, P., Flórez-Revuelta, F.: Silhouette-based human action recognition using sequences of keyposes. Pattern Recognition Letters **34**(15) (2013) 1799–1807 Smart Approaches for Human Action Recognition.
17. Russ, J.C.: The image processing handbook. CRC Press (2006)
18. Chaaraoui, A.A., Flórez-Revuelta, F.: Optimizing human action recognition based on a cooperative coevolutionary algorithm. Engineering Applications of Artificial Intelligence (2013) Advances in Evolutionary Optimization Based Image Processing DOI 10.1016/j.engappai.2013.10.003.
19. Ward, J.A., Lukowicz, P., Gellersen, H.W.: Performance metrics for activity recognition. ACM Transactions on Intelligent Systems and Technology **2**(1) (2011) 6:1 – 6:23
20. Weinland, D., Ronfard, R., Boyer, E.: Free viewpoint action recognition using motion history volumes. Computer Vision and Image Understanding **104**(2-3) (2006) 249–257
21. Gorelick, L., Blank, M., Shechtman, E., Irani, M., Basri, R.: Actions as space-time shapes. IEEE Transactions on Pattern Analysis and Machine Intelligence **29**(12) (2007) 2247–2253