This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAMD.2014.2315676, IEEE Transactions on Autonomous Mental Development

Adaptive Human Action Recognition With an Evolving Bag of Key Poses

Alexandros Andre Chaaraoui, *Student Member, IEEE,* and Francisco Flórez-Revuelta, *Senior Member, IEEE*

***Abstract*—Vision-based human action recognition allows to detect and understand meaningful human motion. This makes it possible to perform advanced human-computer interaction, among other applications. In dynamic environments, adaptive methods are required to support changing scenario characteristics. Specifically, in human-robot interaction, smooth interaction between humans and robots can only be performed if these are able to evolve and adapt to the changing nature of the scenarios. In this paper, an adaptive vision-based human action recognition method is proposed. By means of an evolutionary optimisation method, adaptive and incremental learning of human actions is supported. Through an evolving bag of key poses, which models the learnt actions over time, the current learning memory is developed to recognise increasingly more actions or actors. The evolutionary method selects the optimal subset of training instances, features and parameter values for each learning phase, and handles the evolution of the model. The experimentation shows that our proposal achieves to adapt to new actions or actors successfully, by rearranging the learnt model. Stable and accurate results have been obtained on four publicly available RGB and RGB-D datasets, unveiling the method's robustness and applicability.**

***Index Terms*—Evolutionary computing and genetic algorithms, Feature evaluation and selection, Human computer interaction, Vision and Scene Understanding**

I. Introduction

HUMAN action recognition has recently become of important interest due to its wide variety of applications. Improvements in vision-based recognition of short-temporal human behaviours have led to advanced visual surveillance systems [1], as well as sophisticated human-computer interaction (HCI) techniques [2], which are applied to gaming or intelligent environments, among others. Although visual interpretation of human motion, like actions or gestures, has been studied extensively [3], specific requirements of dynamic environments have only sparingly been taken into account [4], [5]. These are needed, for example in human-robot interaction scenarios. Especially at home, robots can be useful supporting several safety and health care scenarios as, for instance, monitoring or mobility assistance [6]. These care services, among others, can potentially improve the independent living of the elderly, or serve senior assisted living facilities. Reliable support can only be ensured if robots are intelligent enough to analyse and understand the scenario they perform in and the events that occur, in order to be able to interact appropriately.

The application of human action recognition (HAR) to this specific case of HCI comes along with several additional hurdles: 1) since human behaviours are subject to change depending on the specific scenario and actor, and moreover, the behaviours can vary over time, an incremental and adaptive learning process is required. The system has to adapt its knowledge dynamically to recognise new scenarios as, for instance, new actions or new actors. This process needs to happen incrementally, as the system should be able to learn continuously over time without requiring to start from scratch. Furthermore, the recognition capabilities need to evolve and adapt to the new data that needs to be discriminated; 2) for instance, a robot presents several limitations related to the sensor data that can be collected due to space and weight constraints, and also related to the computational capacity. Therefore, a simple camera setup should be employed and real-time recognition algorithms are required. We have considered these constraints choosing a low-cost feature extraction and a state-of-the-art real-time HAR recognition method.

In order to support this dynamic behaviour of the learning process, the use of an evolutionary algorithm (EA) is proposed. EA can provide good solutions to optimisation problems in a limited amount of processing time. Nevertheless, in this contribution the EA is not only employed as an optimisation technique, but also as on-line solution component, guiding the incremental learning. By means of selection of instances, features and parameters, the learning process of the classification algorithm can be optimised. Instance and feature selection intend to find the optimal subset of, respectively, training instances or features in order to eliminate redundant or noisy data and retain the most characteristic elements [7], [8]. However, parameter selection pursues to obtain the classification algorithm's optimal parameter values for the current data. Although, multi-objective EA can be employed considering the action-class recognition rates as fitness values, for the sake of simplicity, we employ the global recognition rate in order to use a single optimisation objective.

Manuscript received May 15, 2013.

This work has been partially supported by the European Commission under project "caring4U - A study on people activity in private spaces: towards a multisensor network that meets privacy requirements" (PIEF-GA-2010-274649) and by the Spanish Ministry of Science and Innovation under project "Sistema de visión para la monitorización de la actividad de la vida diaria en el hogar" (TIN2010-20510-C04-02). Alexandros Andre Chaaraoui acknowledges financial support by the Conselleria d'Educació, Formació i Ocupació of the Generalitat Valenciana (fellowship ACIF/2011/160). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript. We sincerely thank the reviewers for their constructive and insightful suggestions that have helped to improve the quality of this paper.

A.A. Chaaraoui is with the Department of Computer Technology, University of Alicante, P.O. Box 99, E-03080, Alicante, Spain (e-mail: alexandros@dtic.ua.es).

F. Flórez-Revuelta is with the Faculty of Science, Engineering and Computing, Kingston University, KT1 2EE, Kingston upon Thames, United Kingdom (e-mail: F.Florez@kingston.ac.uk).

A. Summary of the proposal

In this paper, a state-of-the-art real-time HAR method is extended in order to support a dynamic behaviour, i.e. incremental and adaptive learning. The original method is based on a bag-of-key-poses model, which learns the most representative pose representations of the action classes, Sequence recognition is performed, where action labels are assigned to the video sequences based on temporal alignment. Therefore, the learning memory of the method is constituted by the bag of key poses. This model is evolved in our dynamic proposal in order to continuously consider more and more action classes and learning data over time. By means of an evolutionary optimisation method, three optimisation targets are pursued: selection of feature subset, training instances and parameter values. The best performing configuration is learnt through evolution using the current training data. Incrementally, this learning is fed with new data which leads to the evolution of the bag-of-key-poses model. In the performed experimentation, it is analysed how the method develops and adapts itself to the increasingly difficult recognition task. In order to validate our approach on different kinds of input data, both traditional RGB cameras and RGB-D sensors (Microsoft Kinect) have been employed. It can be seen that consistently high results are obtained over the different datasets and input data types, and an outstanding performance increase is obtained with respect to the static learning, in which incremental HAR is performed without any adaptation.

The remainder of this paper is organised as follows: Section II summarises the most recent related works on human action recognition and incremental and adaptive approaches. Section III details the static human action recognition method which has been tested for incremental learning, and our dynamic proposal based on evolutionary optimisation. Furthermore, the different learning trajectories that are considered are presented. In section IV, the proposal's performance is analysed using a publicly available RGB-D dataset. Validation on traditional RGB datasets is also provided. Finally, we present conclusions and discussion in section V.

II. Related work

In this section, the research fields of the present work are briefly analysed, summarising the most relevant and recent work. First, a background on human action recognition is presented. Then, works which apply incremental and adaptive learning techniques to HAR are detailed.

A. Human Action Recognition

Recently, interest in human action recognition based on vision techniques has greatly grown due to several advances, both in data acquisition and feature extraction tasks, and in motion modelling and recognition [2]. For instance, these have made possible to develop advanced HCI applications, which are already being used nowadays. Huge advances have been made with regard to feature extraction methods from RGB colour images, which can be classified as *holistic* or *sparse* approaches. Among the former, the usage of human silhouettes allows to limit the region of interest and to reduce the characteristic data to shape and motion. Human silhouettes are usually obtained using background subtraction, but they could also be processed by means of human detection algorithms, or specialised cameras, as depth or infra-red sensors. Shape or shape-temporal features are then extracted in order to encode the most characteristic information [9], [10], which can also be made view-invariant [11]. Although accurate recognition results can be obtained using this kind of dense spatio-temporal approaches, two main difficulties remain: 1) the performance decay related to non-predictable occlusions, and 2) the lack of suitability for recognition of subtle movements, like gestures [12]. Sparse (also known as local) approaches try to overcome these difficulties by relying on a collection of smaller regions of interest, the so-called interest or key points. These are selected based on colour, gradient and shape properties (like Harris and SUSAN corners, SIFT and SURF points [13]), as well as their evolution over time, resulting in spatio-temporal interest points. In order to obtain these, traditional salient point detectors are extended to include the temporal dimension [14], [15]. Usually their frequency of appearance is modelled using bag of words. Even if these sparse spatio-temporal methods have the potential to overcome the limitations of the holistic ones, they come along with several new difficulties. The robustness of the feature detection suffers in cluttered environments leading to unstable results, and key point tracking-based approaches present a high computational cost.

RGB-D data, i.e. RGB colour information along pixel-wise depth measurement, is increasingly being used, since the Microsoft Kinect device has been released. Its low-cost and straight-forward data acquisition, allows to obtain marker-less body pose estimation in real time in form of 3D skeletal information [16]. This kind of data results proficient for the gesture and action recognition which is required in gaming and natural user interfaces [12]. Most of the state-of-the-art methods [17]–[22] try to recognise the different actions using all the available joints of the skeleton. However, some joints are more characteristic to represent the pose or movement than others [23]. In fact, the joints in the torso (shoulders, spine, torso, waist, hips, etc.) rarely exhibit strong independent motion; thus, dimensionality reduction, which improves classification performance, can be applied taking these constraints into account [23]. Feature selection can be driven by the application, as some gestures may be performed with the whole body while others only with arms or hands. Recently, some works have tried to find the appropriate set of joints [24], combinations [25] or weighting of them [26], [27] to improve the recognition. A more detailed survey can be found in [12].

B. Incremental and Adaptive Human Action Recognition

Incremental and adaptive learning techniques have been applied rather sparingly to the field of human action recognition [28]–[31], since they are more frequent in related fields as, for instance, visual tracking [32], [33]. In incremental learning, the goal is to improve the learnt model or exemplar-based data, combining the previous experience with the knowledge extracted from the new example(s), in order to both successfully recognise new samples and also improve the recognition of existing ones. It is therefore also known as *iterative* learning [34]. Adaptive learning is closely related, but it focuses on the capacity of adaptation of the learning towards the new data. Specifically, in incremental learning approaches, it is difficult to set the appropriate parameter values of the algorithm if the data is initially unknown. Therefore, the algorithm's configuration should dynamically adapt itself to the new requirements by tuning its configuration [28], for instance, by means of evolutionary algorithms. Ryoo et al. [31] proposed a method to learn novel human activities incrementally. Based on an incremental codebook, mining of visual words is performed. Local spatio-temporal features are clusterised to obtain the bag-of-words model. When a novel activity is added to the system, new visual words are generated and the existing ones are adapted or merged. Recognition is performed based on visual words histograms, which are also sequentially updated. The method achieves similar activity classification rates as other non-incremental approaches. In [29], snippet-level action recognition is targeted using a recursively trained classifier based on a single-hidden layer feed forward neural network, which is extended to present an incremental behaviour. Nonetheless, it also is adaptive, as the input weights are set randomly initially and then adjusted by means of a generalised inverse operation of the hidden layer weight matrices. In this work, the shape of an actor is approximated by adaptively changing intensity histograms to extract pyramid histograms of oriented gradient features. The performance is analysed based on the length of the snippets. It can be seen that with only two frames a recognition rate over 80% is achieved throughout employing from 10 to 50% of the training data of the Weizmann dataset [35]. Wang et al. [30] rely on wearable sensor data instead of vision. Probabilistic neural networks and an adjustable fuzzy clustering algorithm are employed so as to support incremental learning by means of addition of new information and new activities, but also removing existing ones. The possible noise present in the training dataset is explicitly considered by differentiating the importance of pattern neurons.

III. Evolving Bag of Key Poses

A. Original Human Action Recognition Method

As has been previously mentioned, our method builds on a state-of-the-art classification method [36] and extends it to an incremental and adaptive behaviour by means of an evolutionary algorithm. This classification method provides the evaluation of a specific optimisation in terms of the resulting global recognition rate.

The classification method is made up of a learning stage, in which the representative feature instances are modelled as key poses, and a recognition stage, in which sequences of key poses are recognised based on sequence matching. We will go through these parts briefly, a more detailed explanation can be found in [36], [37]. Fig. 1 shows an overview of the process.

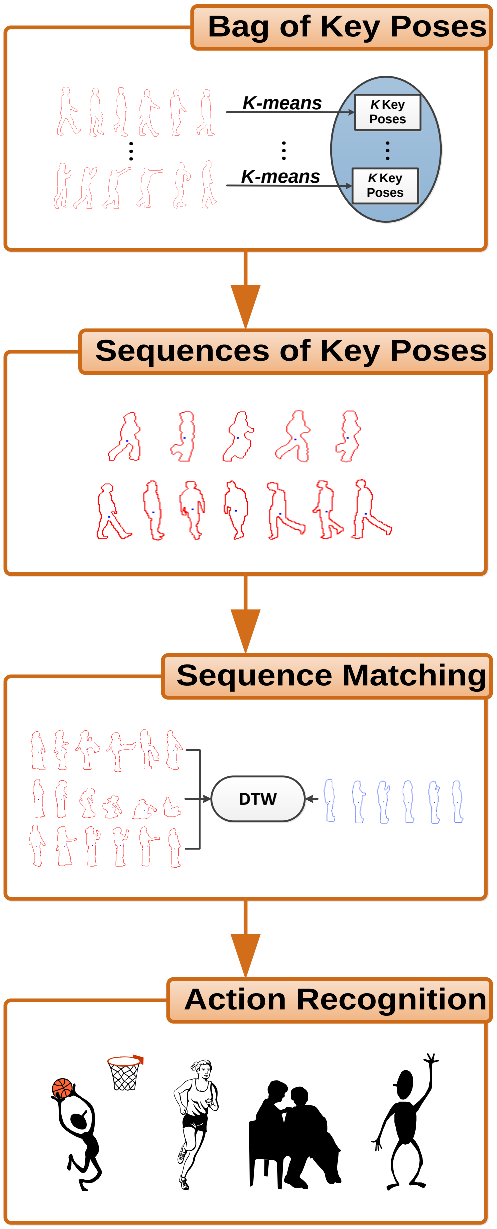


Fig. 1. Overview of the static human action recognition method that has been used: First, learning is performed by means of a bag-of-key-poses model that learns K key poses per action class. Then, the temporal relation between key poses is learnt by modelling sequences of key poses. Action recognition can be performed by matching the unknown sequence of key poses to the closest known one.

1) Learning based on bag of key poses: First, pose representations are obtained out of the available video sequences in order to obtain a feature vector for each frame. The specific feature depends on the type of data and will be detailed in section IV. The most representative feature instances of each action class are learnt by means of key poses. The usage of key poses is motivated by attempting to recognise actions similarly to humans [38]. In this way, our goal is to model an action class based on a few indicative poses and the transitions between them. Using a clustering algorithm as the common K-means algorithm, we generate the $K_1, K_2, ..., K_A$ representative instances for the A action classes by employing

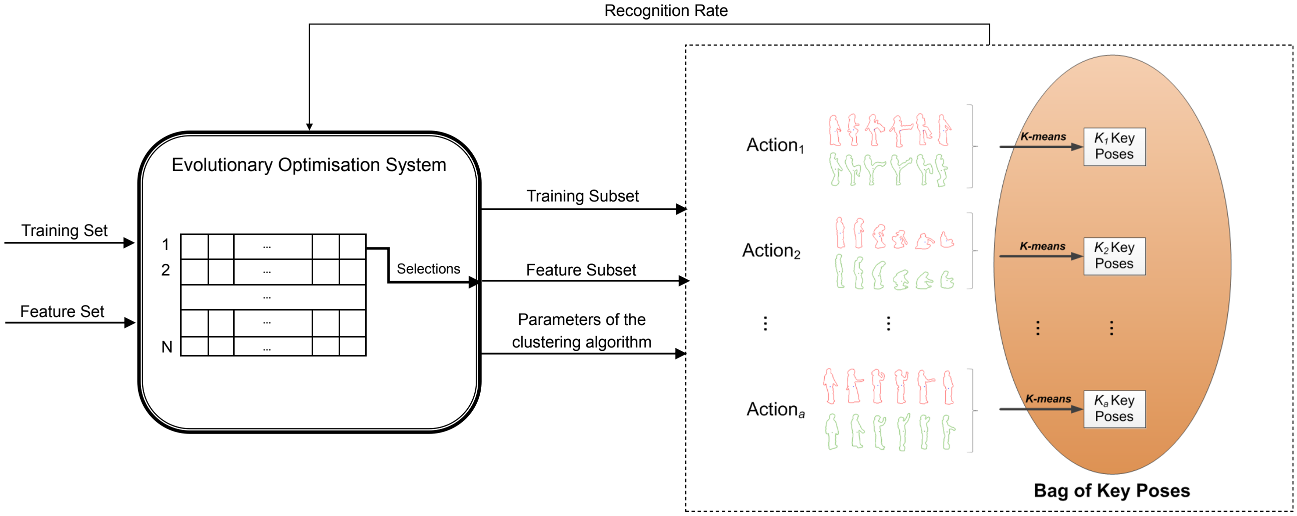


Fig. 2. Evolutionary controller: the evolutionary optimisation system obtains the best performing selection of training instances, features and parameter values by testing them using the human action recognition method. The fitness value of the evaluated individual is then established as the returned recognition rate. In this way, the selections can be improved by applying elitism.

the resulting cluster centres. These action class key poses are combined together in a single bag of key poses.

2) Sequence recognition: With the purpose of modelling the temporal relation between key poses, sequences of key poses are obtained. Specifically, for each training sequence, an equivalent sequence of key poses is built by translating the pose representations, i.e. the RGB or RGB-D based features, to key poses. For each feature instance, the *nearest neighbour* key pose out of the bag of key poses is taken. Since this is also performed for the sequence to be recognised, human actions can be classified using sequence matching. In this regard, dynamic time warping (DTW) [39] has been chosen due to the required alignment of human motion, which is performed at different pace among people of varying condition and age. The result of the classification is defined as the label of the best matching sequence, i.e. the sequence with the lowest DTW distance.

B. Evolution of the model

This bag-of-key-poses model may be used in an environment where the complete training set is established at the beginning of the learning process [36]. In this paper, we extend the model in order to allow an adaptive and incremental learning. The bag-of-key-poses model will continuously evolve in order to adapt to the new data and, at the same time, optimisation is applied, trying to find the best training set, features and parameters to improve the global recognition rate (see Fig. 2).

The evolutionary optimisation system consists of a population of individuals representing combinations of training set, feature subset and parameters. The fitness value of these individuals is given by the global recognition rate that is obtained using their configuration with the presented HAR method. The optimisation process follows the evolutionary algorithm 1.

Algorithm 1 Evolutionary algorithm

```
Initialise the populations with $N$ randomly generated individuals
Rank the population by fitness using the recognition rate obtained with the configuration provided by each individual

repeat
  ——— Adapt to the inclusion of new data ———
  Increase the length of the training set vector by the number of new training instances $i_{new}$
  if a new action class is learnt then
    Increase by one the length of the parameter vector
    Learn the $K_{new}$ key poses for the new action class
  end if
  if a new actor is learnt then
    Readapt the bag of key poses considering the new instances
  end if

  for number of new individuals to be created do
    ——— Generate a new individual ———
    Create one new individual $ind$ by crossover
    Mutate $ind$
    ——— Calculate fitness ———
    Calculate $fitness(ind)$
  end for
  ——— Generate next generation's population ———
  Rank the population by fitness
  Select next generation's population with elitism
until a termination condition is achieved or forever
```

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAMD.2014.2315676, IEEE Transactions on Autonomous Mental Development

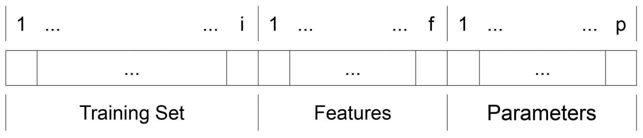


Fig. 3. Structure of the individual: i binary values are defined for the selection of instances, f binary values for the selection of features and p integer values are used to establish the parameter values.

Next, information about this evolutionary process is explained in detail.

1) Individuals' representation: Individuals are encoded as the concatenation of three different vectors representing the training set and feature selections, and the parameters of the clustering algorithm.

The characteristics of each of the vectors are (see also Fig. 3):

- Training set: This vector is made up of i elements corresponding to the number of instances (i.e. video sequences) available for training. This number is variable as new sequences increase the size of the vector. It is encoded as a binary vector, where each gene represents the selection of that instance during the learning.
- Features: This vector contains f elements, one for each feature element of the feature vector that has been obtained for pose representation. This number is constant. Each gene of this binary vector indicates whether or not the corresponding feature element should be used during the classification process.
- Parameters: This vector defines p parameter values, one for each of the A action classes. Therefore, it is increased by one element if a new action must be learnt. Its values correspond to the number of key poses that are learnt for each action class. Each gene takes an integer value in the range of allowed number of key poses.

Each individual in the population stores the bag of key poses obtained with its configuration in order to use it when new instances are included in the learning. This allows the online updating of the bag of key poses when new instances are added to the training set.

2) Crossover: The usual method in evolutionary computation is to perform a single crossover operation to the individual. However, as the individual considered in this work has three different parts, one crossover is applied to each one of the vectors (training set, features and parameters) as if these were independent individuals. This is similar to coevolutionary algorithms. Nevertheless, the use of a coevolutionary algorithm with different populations for each of the vectors would difficult the management of the key poses associated with each of the individuals, because these would be shared among individuals of different populations.

Therefore, a 1-point crossover operator is applied to each part of the individual. The parents are selected by ranking among the individuals of the population. However, if there is some knowledge about the specific problem, e.g. about the structure of the features, other more specific crossover operators can be applied (see Section IV-A).

3) Mutation: Similarly to the crossover operation, a mutation is performed over each part of the individual with different probabilities. Instance and feature vectors use standard mutation, i.e. each gene changes its value according to probabilities mut_I and mut_F. In the parameter population, each gene is mutated with a probability mut_P. This mutation can be done in two different ways (with equal chance): modifying slightly its value applying Gaussian noise, or setting it to a random value in an interval.

C. Inclusion of New Data

In this section, the different learning trajectories that have been considered are detailed. If we try to classify the development of the human brain from a machine learning perspective, two clearly different learning trajectories can be distinguished: 1) the learning of new data which belongs to previously unknown classes, or 2) the learning of new samples which belong to known classes. For this reason, two learning trajectories have been designed, learning of new actions classes and learning of class samples of new actors (see Fig. 4 for a graphical explanation on how this affects the evolving bag-of-key-poses model). Note that, in the first case, the inclusion of more and more action classes reduces the inter-class distance, whereas in the second case, the inclusion of new actors, which perform actions differently, produces an increase of the intra-class distance.

1) Learning of New Actions: Learning a new action involves the generation of the specific K_{new} key poses for that action in the bag of key poses. The following steps are performed:

(a) The length of the training set vector of each individual is updated by increasing it by the number of training instances of the new action class i_{new}. The new binary genes are set randomly following the same distribution than when the population was initialised.

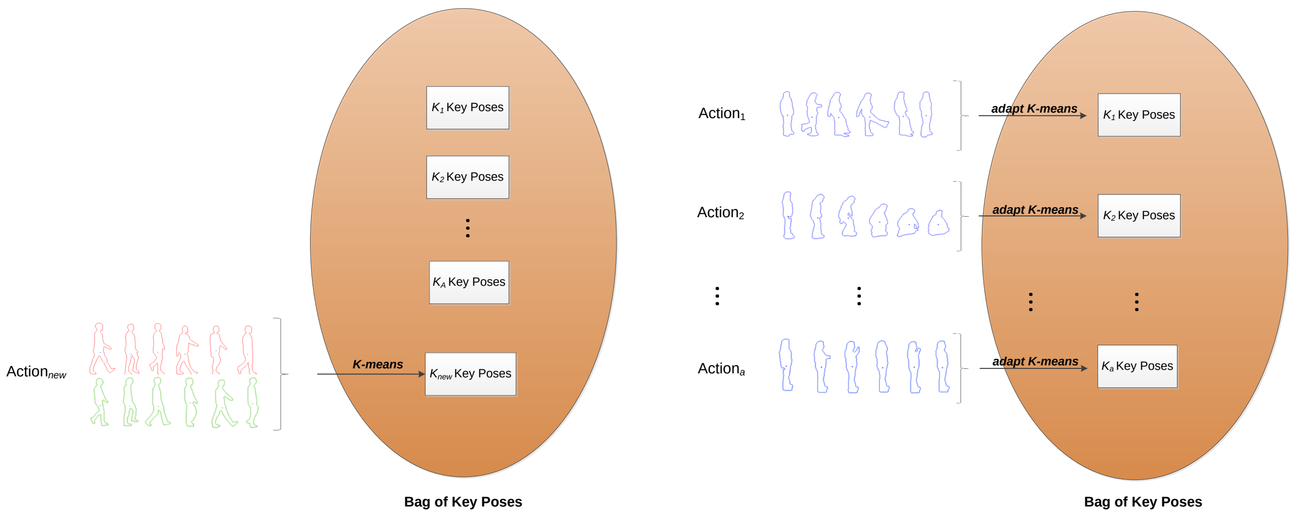
(b) The length of the parameter vector is increased by one gene associated to the number of key poses of the new action $Action_{new}$. This gene is set to a random value in the interval of possible K.

(c) The K_{new} key poses for the new action are learnt, executing the K-means clustering algorithm for each individual of the population and recalculating its fitness. The key poses for the actions that have already been learnt do not need to be obtained as they are stored along with each individual.

2) Learning of New Actors: If training instances of a new actor are learnt, the incremental learning is applied as follows:

(a) As for the learning of new actions, the training set vector is increased by the number of training instances of the new actor i_{new}, and the values are initialised randomly.

(b) Learning training instances performed by a new actor does not involve to apply a new K-means clustering. However, the new training instances must be considered in the K-means associated with each action. This is achieved by initialising the K-means algorithm with the previously obtained key poses instead of using random



(a) New actions

(b) New actor

Fig. 4. Incremental learning of new actions (a) and new actors (b). It can be seen that for a new action class, the bag of key poses is updated with the corresponding K_{new} key poses. But for the inclusion of new actors, all the key poses obtained for the A action classes need to be updated by adapting their key poses to the performances of the new actor.

vectors. This process will produce a readaptation of the K-means clustering algorithm, and an updating of the bag of key poses that, consequently, will take into account the new data.

Once the current population is updated with the new data, new individuals can be created and the evolution continues until a termination condition is achieved (for instance, that no more new data has to be learnt).

IV. Experimentation

A. Adaptive Human Action Recognition with RGB-D cameras

The proposed method has been evaluated with the MSR Action3D dataset [40]. This dataset contains 20 different actions performed by 10 subjects with up to three repetitions. This makes a total of 567 sequences. However 10 sequences are not used in that paper because the skeletons were either missing or wrong, as explained by the authors[1]. Specifically, we apply a *leave-one-actor-out* cross validation. In this test, the sequences from all but one actor are used for training, and the remaining sequences are used in the evaluation. This is repeated for all the available actors. The average score is used as the global recognition rate. This test focuses especially on the robustness to actor variance, which is related to the peculiarities of each subject's appearance and behaviour.

Li et al. [40] divided the dataset in three subsets of eight gestures each. These are shown in Table I. Most of the papers working with this dataset have also used these subsets, due to the high computational cost of dealing with the whole dataset. The AS1 and AS2 subsets are intended to group actions with similar movement, whereas AS3 is intended to group complex actions together.

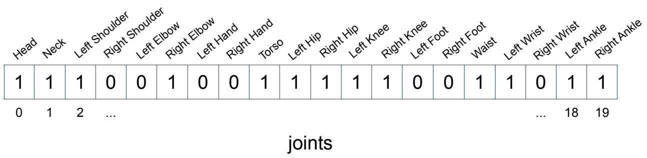


Fig. 5. 20-joint model from the MSR Action3D dataset. A sample selection of joints is shown to illustrate how the f binary values of the individual are employed for feature selection.

1) Feature set: For pose representation, the skeletal 3D pose information has been employed. Since the MSR Action3D dataset uses the 20-joint model described in Fig. 5, we used a feature vector of 20 genes ($f = 20$), where each gene represents whether or not a specific joint is used in the recognition algorithm. For the classification, each joint is described by its 3D coordinates, applying a normalisation process where the skeleton is normalised to scale and rotation (see [24] for more details).

2) Ad-hoc crossover operator for the feature vector: As it has been stated before, normally a 1-point crossover operator would be considered. However, since the joints have a known topology relation among them, an ad-hoc crossover operator has been designed. It works similar to the typical crossover in genetic programming where a node in one parent is randomly selected and the branch below it is substituted by the same branch from a different parent (see Fig. 6).

3) Mutation: Mutation is performed over the three vectors with different probabilities. Instead of having a static mutation probability, the system selects each time a random value in an interval: $mut_I \in [0, 0.1]$, $mut_F \in [0, 0.2]$ and $mut_P \in$

[1] MSR Action Recognition Datasets and Codes, http://research.microsoft.com/en-us/um/people/zliu/actionrecorsrc/default.htm (last access: 02/01/2014)

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAMD.2014.2315676, IEEE Transactions on Autonomous Mental Development

TABLE I
ACTIONS OF EACH OF THE *MSR-Action3D* SUBSETS

AS1		AS2		AS3	
Label	Action name	Label	Action name	Label	Action name
a02	Horizontal arm wave	a01	High arm wave	a06	High throw
a03	Hammer	a04	Hand catch	a14	Forward kick
a05	Forward punch	a07	Draw cross	a15	Side-kick
a06	High throw	a08	Draw tick	a16	Jogging
a10	Hand clap	a09	Draw circle	a17	Tennis swing
a13	Bend	a11	Two-hand wave	a18	Tennis serve
a18	Tennis serve	a12	Side-boxing	a19	Golf swing
a20	Pick-up and throw	a14	Forward kick	a20	Pick-up and throw

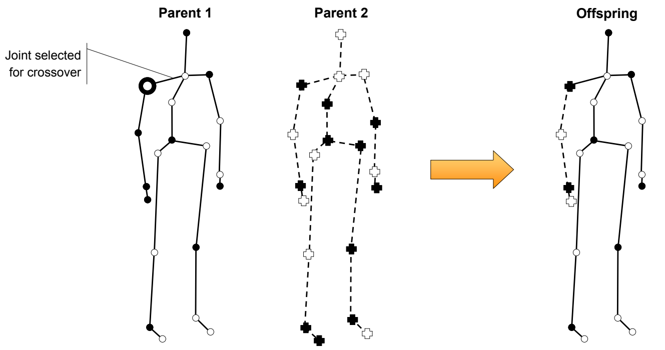


Fig. 6. Crossover between skeletons: The randomly selected joint serves as crossover point. All the joints corresponding to the branch below it are substituted with the values from a different parent. The current selection of joints is represented by the colour of the joints: white joints are selected, black ones are discarded.

$[0, 0.2]$. In this way, we try to avoid early convergence of the evolution to a local minimum. Besides, in order to limit the search space, the genes in the parameter vector take a value between 4 and 75, which represents the K value, i.e. the number of key poses for each action class. These values have been chosen experimentally.

4) Results: In Fig. 7, an extensive analysis of the results which have been obtained on the AS2 dataset is shown. We have chosen this particular dataset for an exhaustive analysis of our adaptive approach because it is considered to be the most difficult among the three previously mentioned sets. The graph shows the behaviour of the whole system, comparing the dynamic and static approaches when actions are sequentially incorporated every 50 generations. With the purpose to ease the usage of the results as references and establish a learning order, we employ the given alphanumeric order of the actions. In order to be able to perform classification, we start with the two first action classes *a01* and *a04*. All the tests have been performed with a population of $N = 10$ individuals.

The static line represents the best solution considering that the system is using all the training instances, all the joints and the default value of the parameters (these have been set to 16 for all the action classes). Therefore, in the static execution, incremental learning is applied without any adaptation. This line varies during a stage because of the non-deterministic behaviour of the K-means clustering, which is related to its random initialisation. Obviously, the system always uses the configuration that has obtained the best recognition results until that moment, considering an equivalent number of 50 iterations per stage. For this reason, the recognition rate may increase during a learning stage. The dynamic lines show three different runs of our evolving proposal of adaptive and incremental learning with different random initialisations.

The graph shows that when a new action is included the recognition rate usually suffers a dramatic change. Sometimes the recognition rate decreases. This is because the new action is difficult to recognise, or because there are other similar actions already learnt by the system and misclassifications are performed. The global recognition rate may also improve if the new action is easy to recognise by the system, since the average rate is improved in this case.

The recognition rate is globally better in any of the dynamic runs. The inclusion of new actions affects less than in the static version and, if it is affected, it rapidly evolves, selecting the appropriate training set, joints and parameters, thus, improving the recognition rate.

In Fig. 7 and Tables II, III and IV further analysis of results is provided for the *Dynamic B* evolution. The bottom part of Fig. 7 shows the evolution of the joints vector throughout the learning following the definition presented in Fig. 5. It can be observed that some joints (as the RIGHT HAND and the RIGHT WRIST corresponding to positions 7 and 17) are selected throughout the whole learning. Obviously, this is related to the actions included in AS2, which are mostly performed with the right hand.

The top part shows the corresponding skeleton at the end of every stage. The joints that are finally selected are those from the arms, as all the actions in this dataset involve mostly these body parts and, as has been already mentioned, mainly the right arm. The consideration of other body parts during the evolution may be due to slight motions of those joints that allow to differ actions, or because the evolution has not been long enough to discard them.

Table II shows the global recognition rate and the specific rate for each action class at the end of each stage (every 50 generations). It allows to observe how the addition of action classes affects the recognition of the already learnt classes. It can be seen that the newly introduced class is always recognised with a very high success rate, which means that the learning successfully adapts to the new data during the learning stage. The addition also influences the recognition rates of the other classes. These can decrease, when the inter-class similarity is increased and therefore new confusions appear. But they may also increase, when the adaptation

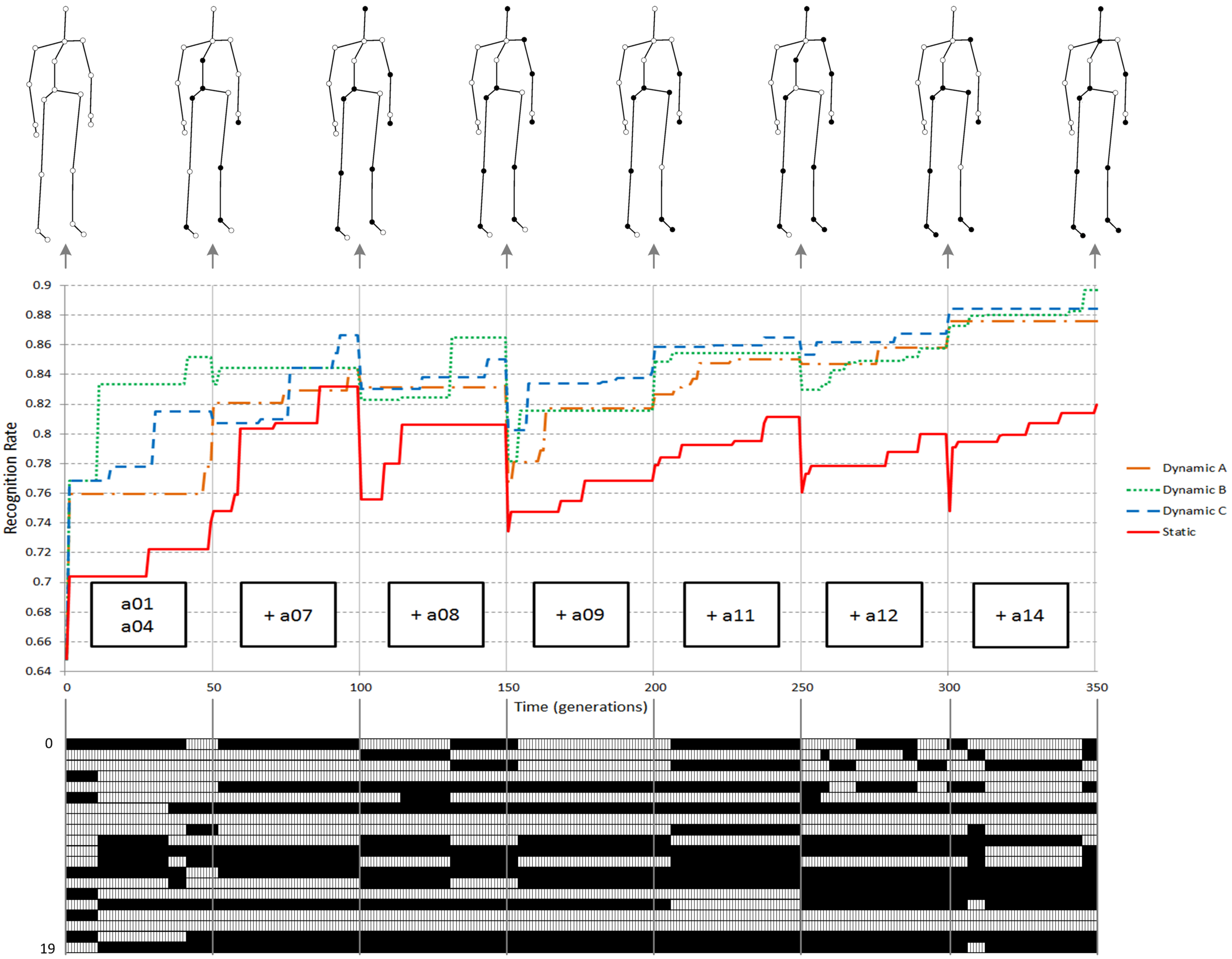
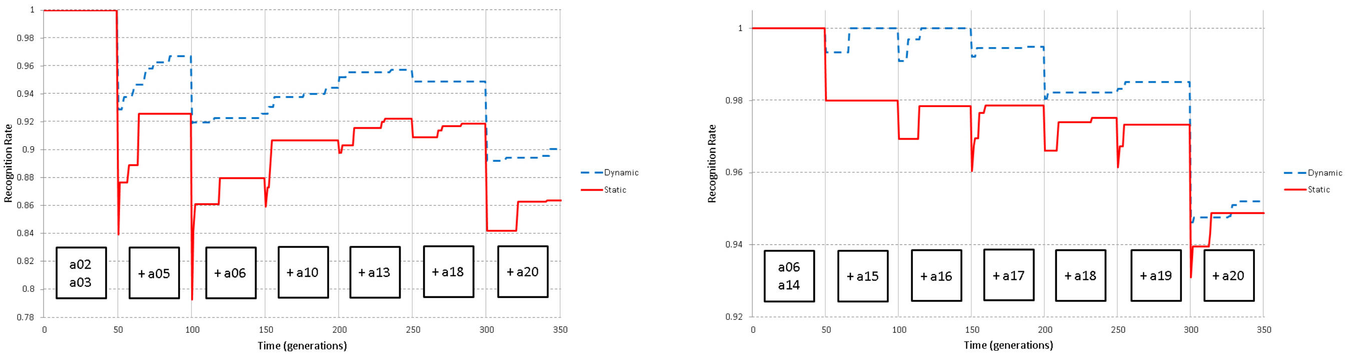


Fig. 7. Analysis of the results of the dynamic learning for the AS2 dataset: At the middle, the recognition rates for the static learning and three different runs of the dynamic learning (A, B and C) are shown. At the top, the selections of joints that are obtained after each stage are detailed. At the bottom, the selection of joints of the whole evolution can be seen following the definition presented in Fig. 5. Note that the feature selections correspond to the run *Dynamic B*, and selected joints or feature elements are shaded in white and discarded ones in black.



(a) AS1

(b) AS3

Fig. 8. Learning for the AS1 (a) and AS3 (b) datasets. A static and the average of three dynamic runs are shown.

process has been able to optimise the recognition of the actions, despite of the new data that has been introduced. Furthermore, the continuous optimisation can also negatively influence the recognition of a specific action class, because the global rate is used as optimisation target.

Table III shows the number of instances of each action class that have been chosen for training. In this case, around 50% of the instances are discarded, since including them decreases the recognition rate, mainly because they incorporate noise to the training. Needless to say that, although these instances are not considered in learning, all of them, chosen and discarded, are used in the testing phase. One fact is that the number of instances considered per action class usually decreases until it converges to the most discriminative and useful set of instances.

Table IV shows the values for the parameter vector, i.e. the number of key poses considered to represent each action. This vector seems to fluctuate more than the training set or feature vectors. This could have two main reasons:

(a) the key poses obtained with the K-means clustering algorithm when a new action is included are similar to other previously learnt ones, what causes a change of size; or

(b) the parameter vector is less important than the training set or feature vectors in order to improve the recognition rate. So, changes in this vector are not very relevant to the behaviour of the system.

Fig. 8 shows the evolution for the other two subsets, AS1 and AS3. In this case, the static learning is compared to the average of three dynamic runs. Results are quite similar to those obtained with AS2, i.e. recognition rate is better almost at every moment for the dynamic run. AS3 is the easiest subset. The effect is that there is not too much possibility to improve the results over the static version. It is worth mentioning that many sequences for *a20*, present in this dataset, are corrupted. The skeletons are not well obtained from the depth image, since there are many body occlusions. This produces a low recognition rate for this specific action.

The final results for our adaptive proposal shown in Fig. 7 and 8 are close to the results that would be obtained if the whole training data, with all the actions, had been used since the beginning of the learning (Table V).

We have also analysed the effect of the amount of time that is employed to learn a new action. Fig. 9 shows how more time per stage, i.e. more generations, allows a longer evolution and, consequently, a greater optimisation of the recognition. It can be observed that in the long term, the system adapts better if more time is available, and significant differences can be seen from 10 to 100 generations per stage, although approximately 50 generations are enough to outperform the static incremental learning.

Finally, we have also tested the system with the complete MSR Action3D dataset. It is very challenging to consider all the 20 actions, since many of them are quite similar. For instance, confusion between actions *a01* to *a09* is high. Consequently, the results are affected (see Fig. 10), and the recognition rate decreases when these actions are included. However, as in the previous cases, our dynamic approach recovers fast after each action is added, and the recognition rate is better than with the static option.

Fig. 11 shows the evolution of the system (average of three different runs) when new actors are added instead of new actions. In this test, always all the actions are considered and new actors are sequentially introduced to the learning. We have added the actors in the given order, starting with actors $s01$ and $s02$ and adding a new one every 50 generations. The results show that when few actors are considered, there are too few sequences for training and this produces low recognition rates. However, as more instances are included with new actors, the system is able to evolve to similar final results to those that have been obtained when new actions are learnt (see Fig. 7). It can also be observed that both in the static and dynamic runs, some subjects do not provide valuable data to improve the classification, which is probably related to noise and outlier values regarding their action performances. The dynamic approach proves to handle also this case better, since less performance decrease is obtained (for instance, when subject *s05* is learnt).

B. Adaptive Human Action Recognition with RGB cameras

The proposed approach has also been validated on regular RGB cameras. For this purpose, a silhouette-based feature is employed taking into account multiple views. Three state-of-the-art publicly available benchmarks have been tested. As it will be seen, the adaptive and incremental learning algorithm shows steadily promising results despite the singularities of different action classes, actors and scenario-related conditions.

1) Silhouette-based feature: As it has already been seen in section II-A, human silhouettes have been used successfully in order to recognise whole-body movements based on shape. So as to reduce the silhouette's dimensionality and noise (related to the commonly inaccurate background subtraction), we extract a radial summary feature which is based on the silhouette's contour points $\boldsymbol{P} = \{p_1, p_2, ..., p_n\}$, where $p_i = (x_i, y_i)$. First, the silhouette's contour is divided in S radial bins of the same angular width taking the silhouette's centroid as the origin. The centroid C is computed as $C = (x_c, y_c)$, with $x_c = \frac{\sum_{i=1}^{n} x_i}{n}, y_c = \frac{\sum_{i=1}^{n} y_i}{n}$. The corresponding radial bin of each contour point can be obtained as follows (for the sake of simplicity $\alpha_i = 0$ is considered as $\alpha_i = 360$):

$$\alpha_i = \begin{cases} \arccos(\frac{y_i - y_c}{d_i}) \cdot \frac{180}{\pi} & \text{if } x_i \geq 0, \\ 180 + \arccos(\frac{y_i - y_c}{d_i}) \cdot \frac{180}{\pi} & \text{otherwise,} \end{cases} \quad (1)$$

$$s_i = \left\lceil \frac{S \cdot \alpha_i}{360} \right\rceil, \quad \forall i \in [1...n], \quad (2)$$

where d_i stands for the Euclidean distance between contour point p_i and the centroid.

Then, for each radial bin, a summary value v_j is computed. This value is defined as the statistical range of the distances from the contour points to the centroid.

$$\begin{aligned} v_j = & \max(d_k, d_{k+1}, ..., d_l) - \min(d_k, d_{k+1}, ..., d_l) \\ & / \ s_k ... s_l = j \wedge k, l \in [1...n], \quad \forall j \in [1...S], \end{aligned} \quad (3)$$

TABLE II
CLASS-SPECIFIC RECOGNITION RATES OF THE AS2 DATASET AT THE END OF EACH STAGE OF THE LEARNING (*corresponds to Dynamic B*)

	Generation	Global	a01	a04	a07	a08	a09	a11	a12	a14
a01 + a04	50	85.19%	88.9%	80.0%						
+ a07	100	84.44%	74.1%	72.0%	100.0%					
+ a08	150	86.50%	85.2%	84.0%	81.5%	96.7%				
+ a09	200	81.55%	77.8%	64.0%	81.5%	86.7%	93.3%			
+ a11	250	85.46%	74.1%	84.0%	77.8%	93.3%	80.0%	100.0%		
+ a12	300	85.75%	81.5%	64.0%	81.5%	96.7%	80.0%	100.0%	90.0%	
+ a14	350	89.7%	59.3%	84.0%	88.9%	96.7%	90.0%	100.0%	93.3%	100.0%

TABLE III
NUMBER OF SELECTED INSTANCES PER ACTION CLASS (TRAINING SUBSET) FOR THE AS2 DATASET AT THE END OF EACH STAGE OF THE LEARNING. THE SECOND ROW (GREY CELLS) SHOWS THE TOTAL AVAILABLE NUMBER OF INSTANCES PER ACTION CLASS (*corresponds to Dynamic B*)

		a01	a04	a07	a08	a09	a11	a12	a14	
	Generation	27	25	27	30	30	30	30	29	**% used**
a01 + a04	50	10	12							22/52
+ a07	100	12	13	18						43/79
+ a08	150	13	12	18	20					63/109
+ a09	200	13	7	20	16	26				82/139
+ a11	250	14	11	16	19	22	30			112/169
+ a12	300	10	13	15	15	23	14	11		101/199
+ a14	350	10	13	15	15	19	15	10	24	121/228

TABLE IV
NUMBER OF KEY POSES PER ACTION CLASS FOR THE AS2 DATASET AT THE END OF EACH STAGE OF THE LEARNING (*corresponds to Dynamic B*)

	Generation	a01	a04	a07	a08	a09	a11	a12	a14	Total	Average
a01 + a04	50	56	59							115	57.50
+ a07	100	52	66	10						128	42.67
+ a08	150	75	62	22	15					174	43.50
+ a09	200	69	35	22	30	41				197	39.40
+ a11	250	6	38	65	15	55	16			195	32.50
+ a12	300	7	20	16	75	68	9	18		213	30.43
+ a14	350	27	9	24	32	25	38	71	40	266	33.25

Finally, the feature vector $\bar{\boldsymbol{V}}$ is obtained by concatenating the normalised summary values:

$$\bar{v_j} = \frac{v_j}{\sum_{q=1}^{S} v_q}, \quad \forall j \in [1...S], \tag{4}$$

$$\bar{\boldsymbol{V}} = \bar{v_1} \parallel \bar{v_2} \parallel ... \parallel \bar{v_S}. \tag{5}$$

Since this feature vector contains S elements, the same amount of genes ($f = S$) are employed for the individuals of the evolutionary algorithm. By applying feature subset selection to this descriptor, redundant or noisy body parts can be ignored, and the most characteristic parts which are involved in the actions can be retained.

In the case of a multi-view scenario, this feature is first obtained for each view. Then, by means of feature fusion, a multi-view pose representation is built using feature concatenation.

TABLE V
COMPARISON BETWEEN INCREMENTAL AND NON-INCREMENTAL LEARNING. RESULTS SHOW THE AVERAGE RECOGNITION RATES OF THREE RUNS AT THE 350^{th} GENERATION (*first column corresponds to the final rates shown in Fig. 7 and 8*)

Dataset	Incremental learning	Non-incremental learning
AS1	90.06%	91.50%
AS2	88.56%	90.80%
AS3	95.21%	96.99%

Note that in this case a standard 1-point crossover and the same mutation probabilities from the RGB-D experimentation are employed.

2) Benchmarks: For the evaluation on RGB images, both single- and multi-view datasets have been chosen. The popular Weizmann dataset [35] includes ten different actions performed by nine actors. For the recorded static-front view, human silhouettes are available. These have been obtained automatically by means of background subtraction (the ones without post-alignment are employed). We use all the 93 available sequences and perform a *leave-one-actor-out* cross validation (LOAO).

The MuHAVi dataset [41] includes images from multiple viewpoints. Its subset MuHAVi-MAS provides manually annotated silhouettes of very good quality for two camera views and two subjects. It comes in two versions, with either 8 (MuHAVi-8) or 14 (MuHAVi-14) different action classes. The same LOAO cross validation has been tested on this dataset (called *Novel Actor* test by its authors).

Finally, we also tested our system with the challenging INRIA XMAS dataset (IXMAS) [11]. Four side views and a top view are available for up to 11 actions which have been performed by 12 different actors, three times each. Subjects were allowed to choose their location and orientation freely,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAMD.2014.2315676, IEEE Transactions on Autonomous Mental Development

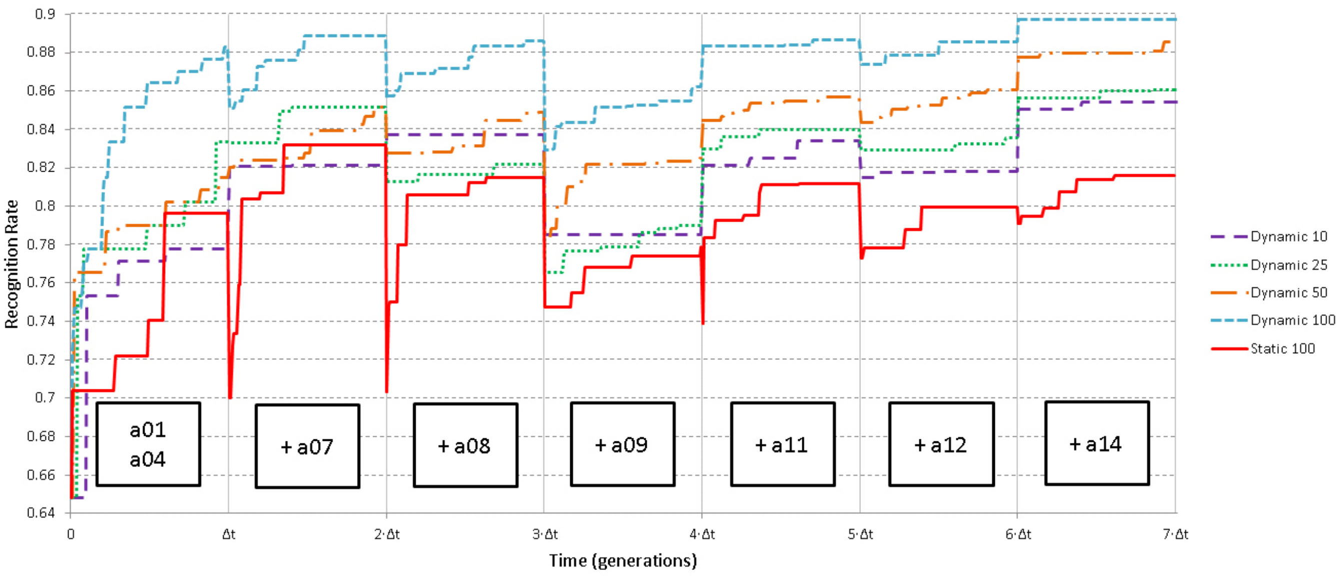


Fig. 9. In this graph, different adaptation times are compared. A static run of 100 iterations per learning stage and three dynamic runs of respectively 10, 25, 50 and 100 generations (Δt) per learning stage are shown. The smaller dynamic runs are interpolated to 100 iterations for visual purposes.

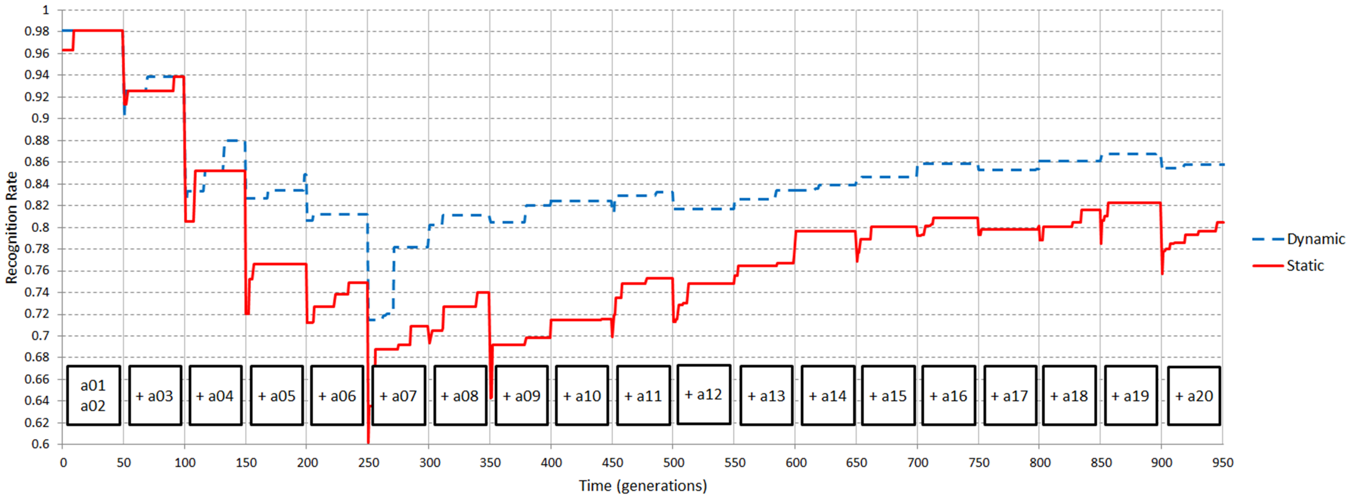


Fig. 10. Results of the learning for the whole MSR Action3D dataset. A static and a dynamic run are shown for the incremental learning of 20 action classes.

which implies that this dataset is also testing view-invariance implicitly. In this case, automatically obtained silhouettes of a much lower quality are provided. Again, LOAO cross validation has been employed to obtain the recognition rate on this dataset.

3) Results: Fig. 12 shows the results that have been obtained on the Weizmann dataset with the adaptive human action recognition method and the static approach. Again, each 50 generations, a new action class is incorporated to the evolutionary process. As it can be seen, depending on the newly introduced action (and on the already learnt ones) the static result decreases or increases. Especially, the *skip* action, that is commonly excluded by other authors [42], [43], causes a dramatic performance drop from which the static approach does not achieve to recover. In contrast, the dynamic approach handles this drop and also others very well. In few iterations, the system adapts to the new data and achieves to return high results steadily. Furthermore, the drops are not as relevant as with the static approach, which means that the dynamic approach leads to a more stable and general learning model.

Fig. 13 and 14 show the results on the MuHAVi dataset. Whereas on MuHAVi-8 the static approach starts to suffer an important performance decrease when the fifth action class (*run*) is introduced, the dynamic approach shows to support the incremental learning exceptionally well. With MuHAVi-14, 100 generations have been employed for each stage, since due to the higher number of action classes (14) the system requires more time to adapt to the new data. Again, our proposal shows that it successfully adapts to the new action classes to recognise, and it is less affected by the new data. The obtained recognition rates are globally better and more stable. As an example, the final feature subset selection of the best

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAMD.2014.2315676, IEEE Transactions on Autonomous Mental Development

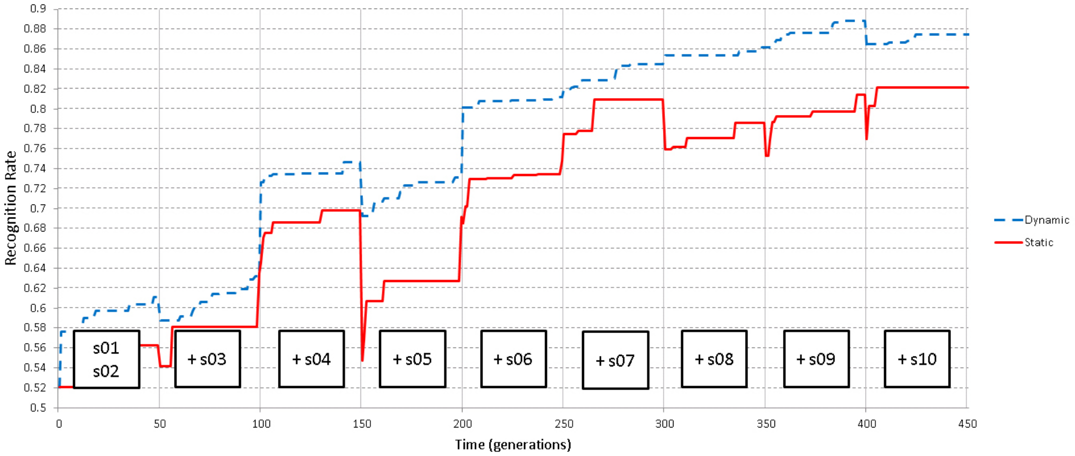


Fig. 11. Results of the learning of new actors for the AS2 dataset. In this case, all the actions are known from the beginning, but each 50 generations, samples of a new actor are learnt. A static and the average of three dynamic runs are shown.

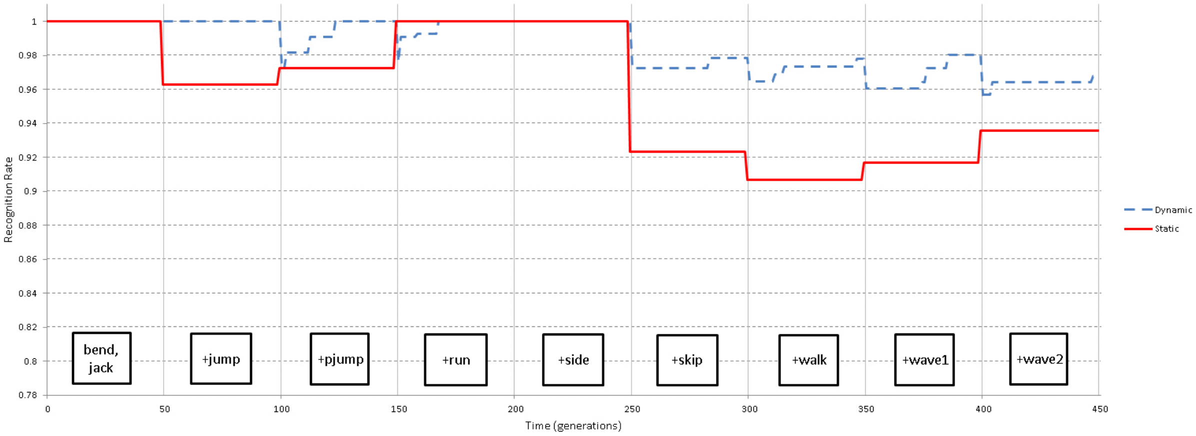


Fig. 12. Results of the learning of the Weizmann dataset. Recognition rates for the static learning and the average of three runs of the dynamic learning are shown. The number of employed radial bins $S = 14$ and the default value for $K_1, K_2, ..., K_A$ is 10 (with a range from 4 to 30).

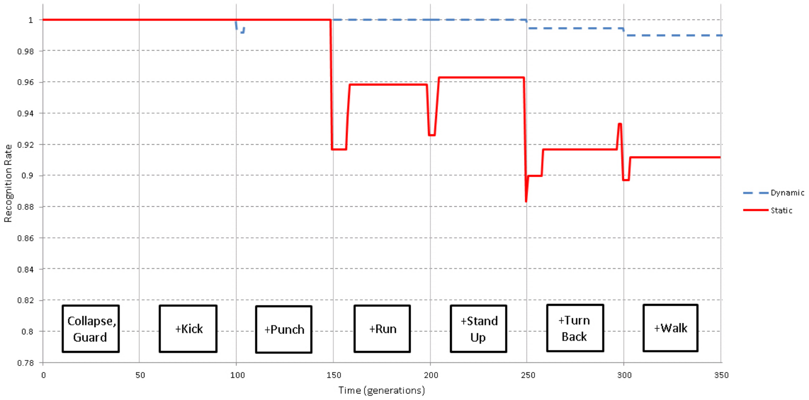


Fig. 13. Results of the learning of the MuHAVi-8 dataset. Recognition rates for the static learning and the average of three runs of the dynamic learning are shown. The number of employed radial bins $S = 18$ and the default value for $K_1, K_2, ..., K_A$ is 7 (with a range from 4 to 30).

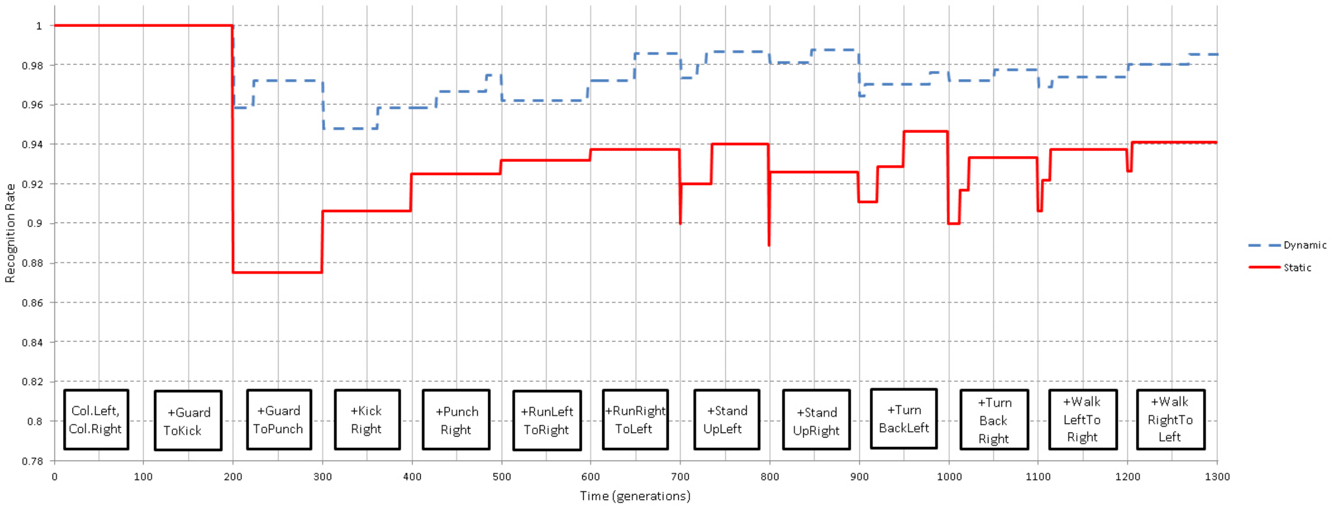


Fig. 14. Results of the learning of the MuHAVi-14 dataset. Recognition rates for the static learning and the average of three runs of the dynamic learning are shown. The number of employed radial bins $S = 10$ and the default value for $K_1, K_2, ..., K_A$ is 4 (with a range from 4 to 30).

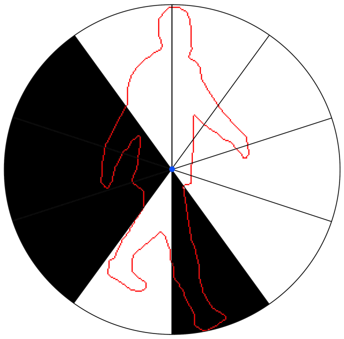


Fig. 15. Final feature subset selection that has been obtained for the MuHAVi-14 dataset. White bins are selected, black ones are discarded.

performing individual is shown in Fig. 15.

Finally, the method has been tested also on the much larger IXMAS dataset. Fig. 16 compares the dynamic and the static behaviour. Since the dataset includes nearly 2 000 video sequences, only a single full run could be executed. Nonetheless, also on this challenging data, the proposed method presents an importantly improved behaviour with lower performance drops and a continuously higher recognition rate over the different learning stages. It can be seen that in this case, although the learnt selection of features, training instances and parameter values successfully supports the inclusion of new actions to recognise, less improvement is observed during the learning stages. This is related to the limited amount of generations used for each learning stage, as well as to the data variance.

V. Conclusion

In this paper, an adaptive human action recognition method that can be applied to intelligent environments or autonomous robots has been presented. Based on an evolutionary algorithm, a dynamic learning of human actions is supported by evolving a bag-of-key-poses model. At the same time, through evolution, the best performing selection of training instances, features and parameters is sought. Therefore, the contribution of the evolutionary approach is two-fold. One the one hand, it serves as optimisation method in order to improve the adaptation to the new data to be recognised and increase the recognition rate. On the other hand, it guides the dynamic behaviour in which new data is incrementally learnt and the bag-of-key-poses model is evolved. In this way, the method is able to support the inclusion of new data successfully with small performance changes, that are overcome in few generations in which the method adapts to the new data. Two learning trajectories have been considered and tested, the inclusion of data from unknown action classes or new samples of unknown actors. The approach has been validated on two different data input types: RGB-D data obtained with a Microsoft Kinect device, and traditional RGB images. An extensive analysis on the AS2 subset from the MSR Action3D dataset shows that, although the obtained results vary due to the random initialisation and non-deterministic behaviour, the proposed dynamic approach achieves superior results in comparison to the static incremental learning. This holds true for the tests performed on the other subsets and the whole dataset. Considering the specific changes at the feature level, the validation on RGB images has been detailed. On multiple publicly available datasets of different difficulty and image quality, good to outstanding results have been obtained throughout the whole learning.

In future work, the proposed adaptive learning should be compared to other state-of-the-art incremental learning and continuous adaptation methods. In order to make this comparison possible, a consensus must be reached on how this performance can be measured in terms of continuous recognition rates and adaptation times. In the present work, the objective was to maximise the global recognition rate, since all the action classes have been considered as equally important. If this were not the case, or the best individual action-class recognition rates should be obtained, an approach based

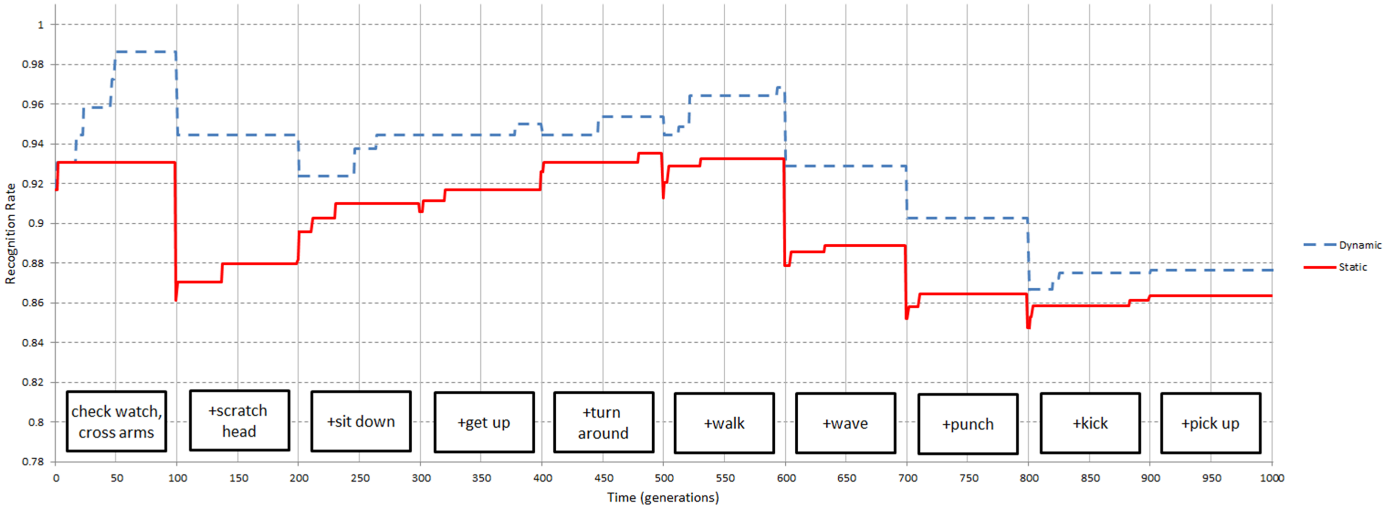


Fig. 16. Results of the learning of the IXMAS dataset. Recognition rates for the static learning and the dynamic learning are shown. The number of employed radial bins $S = 27$ and the default value for $K_1, K_2, ..., K_A$ is 130 (with a range from 4 to 130).

on a multi-objective EA could be employed. Other learning trajectories should be studied too. The intuitive idea to start with the most simple actions and continue to learn more and more difficult ones requires a measurement of difficulty, which could be based on a binary classification rate, since a multi-class one would depend on the other classes. During execution time, new data can be captured, although it would normally be unlabelled. This raises the question whether the data class could automatically be learnt assuming that it corresponds to one of the known classes. Furthermore, new data of unknown actors and actions could be learnt simultaneously. This would require to combine the two learning trajectories that have been presented.

References

[1] M. Valera and S. Velastin, “Intelligent distributed surveillance systems: a review,” *Vision, Image and Signal Processing, IEE Proceedings -*, vol. 152, no. 2, pp. 192–204, 2005.

[2] J. Aggarwal and M. Ryoo, “Human activity analysis: A review,” *ACM Comput. Surv.*, vol. 43, no. 3, pp. 16:1–16:43, Apr. 2011.

[3] V. Pavlovic, R. Sharma, and T. Huang, “Visual interpretation of hand gestures for human-computer interaction: a review,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 19, no. 7, pp. 677–695, 1997.

[4] T. Mann, Y. Park, S. Jeong, M. Lee, and Y. Choe, “Autonomous and interactive improvement of binocular visual depth estimation through sensorimotor interaction,” *Autonomous Mental Development, IEEE Transactions on*, vol. 5, no. 1, pp. 74–84, 2013.

[5] A. Yorita and N. Kubota, “Cognitive development in partner robots for information support to elderly people,” *Autonomous Mental Development, IEEE Transactions on*, vol. 3, no. 1, pp. 64–73, 2011.

[6] S. Dubowsky, F. Genot, S. Godding, H. Kozono, A. Skwersky, H. Yu, and L. S. Yu, “Pamm - a robotic aid to the elderly for mobility assistance and monitoring: a “helping-hand” for the elderly,” in *Robotics and Automation, 2000. Proceedings. ICRA ’00. IEEE International Conference on*, vol. 1, 2000, pp. 570–576.

[7] J. Cano, F. Herrera, and M. Lozano, “A study on the combination of evolutionary algorithms and stratified strategies for training set selection in data mining,” in *Soft Computing: Methodologies and Applications*, ser. Advances in Soft Computing, F. Hoffmann, M. Köppen, F. Klawonn, and R. Roy, Eds. Springer Berlin Heidelberg, 2005, vol. 32, pp. 271–284.

[8] E. Cantú-Paz, “Feature subset selection, class separability, and genetic algorithms,” in *Genetic and Evolutionary Computation GECCO 2004*, ser. Lecture Notes in Computer Science, K. Deb, Ed. Springer Berlin / Heidelberg, 2004, vol. 3102, pp. 959–970.

[9] A. Bobick and J. Davis, “The recognition of human movement using temporal templates,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 23, no. 3, pp. 257–267, mar 2001.

[10] D. Tran and A. Sorokin, “Human activity recognition with metric learning,” in *Computer Vision - ECCV 2008*, ser. Lecture Notes in Computer Science, D. Forsyth, P. Torr, and A. Zisserman, Eds. Springer Berlin Heidelberg, 2008, vol. 5302, pp. 548–561.

[11] D. Weinland, R. Ronfard, and E. Boyer, “Free viewpoint action recognition using motion history volumes,” *Comput. Vis. Image Underst.*, vol. 104, no. 2, pp. 249–257, Nov. 2006.

[12] L. Chen, H. Wei, and J. Ferryman, “A survey of human motion analysis using depth imagery,” *Pattern Recognition Letters*, vol. 34, no. 15, pp. 1995 – 2006, 2013, Smart Approaches for Human Action Recognition.

[13] X. Wu, Z. Shi, and Y. Zhong, “Detailed analysis and evaluation of keypoint extraction methods,” in *Computer Application and System Modeling (ICCASM), 2010 International Conference on*, vol. 2, oct. 2010, pp. 562–566.

[14] I. Laptev, “On space-time interest points,” *International Journal of Computer Vision*, vol. 64, pp. 107–123, 2005.

[15] A. Oikonomopoulos, I. Patras, and M. Pantic, “Spatiotemporal salient points for visual recognition of human actions,” *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 36, no. 3, pp. 710–719, june 2005.

[16] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, “Real-time human pose recognition in parts from single depth images,” in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, june 2011, pp. 1297–1304.

[17] L. Miranda, T. Vieira, D. Martinez, T. Lewiner, A. W. Vieira, and M. F. M. Campos, “Real-time gesture recognition from depth data through key poses learning and decision forests,” in *SIBGRAPI 2012 (XXV Conference on Graphics, Patterns and Images)*. Ouro Preto, MG: IEEE, august 2012.

[18] L. Xia, C.-C. Chen, and J. Aggarwal, “View invariant human action recognition using histograms of 3D joints,” in *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, june 2012, pp. 20 –27.

[19] S. Azary and A. Savakis, “3D Action Classification Using Sparse Spatio-temporal Feature Representations,” in *Advances in Visual Computing*, ser. Lecture Notes in Computer Science, G. Bebis, R. Boyle, B. Parvin, D. Koracin, C. Fowlkes, S. Wang, M.-H. Choi, S. Mantler, J. Schulze, D. Acevedo, K. Mueller, and M. Papka, Eds. Springer Berlin / Heidelberg, 2012, vol. 7432, pp. 166–175.

[20] H. Soh and Y. Demiris, “Iterative temporal learning and prediction with the sparse online echo state gaussian process,” in *The 2012 International Joint Conference on Neural Networks (IJCNN)*, june 2012, pp. 1–8.

[21] S. Fothergill, H. Mentis, P. Kohli, and S. Nowozin, "Instructing people for training gestural interactive systems," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '12. New York, NY, USA: ACM, 2012, pp. 1737–1746.

[22] X. Yang and Y. Tian, "EigenJoints-based Action Recognition Using Naïve-Bayes-Nearest-Neighbor," in *Second International Workshop on Human Activity Understanding from 3D Data in conjunction with CVPR, 2012*, Providence, Rhode Island, 2012, pp. 14–19.

[23] M. Raptis, D. Kirovski, and H. Hoppe, "Real-Time Classification of Dance Gestures from Skeleton Animation," in *Proceedings of the 10th Annual ACM SIGGRAPH/Eurographics Symposium on Computer Animation, SCA 2011*, 2011, pp. 147–156.

[24] A. A. Chaaraoui, J. R. Padilla-López, P. Climent-Pérez, and F. Flórez-Revuelta, "Evolutionary joint selection to improve human action recognition with RGB-D devices," *Expert Systems with Applications*, vol. 41, no. 3, pp. 786 – 794, 2014, Methods and Applications of Artificial and Computational Intelligence.

[25] J. Wang, Z. Liu, Y. Wu, and J. Yuan, "Mining Actionlet Ensemble for Action Recognition with Depth Cameras," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2012)*, Providence, Rhode Island, June 2012.

[26] M. Reyes, G. Dominguez, and S. Escalera, "Featureweighting in dynamic timewarping for gesture recognition in depth data," in *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, 2011, pp. 1182–1188.

[27] S. Sempena, N. Maulidevi, and P. Aryan, "Human action recognition using dynamic time warping," in *Electrical Engineering and Informatics (ICEEI), 2011 International Conference on*, 2011, pp. 1–5.

[28] M. N. Kapp, R. Sabourin, and P. Maupin, "A dynamic optimization approach for adaptive incremental learning," *International Journal of Intelligent Systems*, vol. 26, no. 11, pp. 1101–1124, 2011.

[29] R. Minhas, A. Mohammed, and Q. Wu, "Incremental learning in human action recognition based on snippets," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 22, no. 11, pp. 1529–1541, 2012.

[30] Z. Wang, M. Jiang, Y. Hu, and H. Li, "An incremental learning method based on probabilistic neural networks and adjustable fuzzy clustering for human activity recognition by using wearable sensors," *Information Technology in Biomedicine, IEEE Transactions on*, vol. 16, no. 4, pp. 691–699, 2012.

[31] M. S. Ryoo, J. Joung, S. Choi, and W. Yu, "Incremental learning of novel activity categories from videos," in *Virtual Systems and Multimedia (VSMM), 2010 16th International Conference on*, 2010, pp. 21–26.

[32] H. Yang, L. Shao, F. Zheng, L. Wang, and Z. Song, "Recent advances and trends in visual tracking: A review," *Neurocomputing*, vol. 74, no. 18, pp. 3823–3831, 2011.

[33] J. Lim, D. Ross, R. sung Lin, and M. hsuan Yang, "Incremental learning for visual tracking," in *Advances in Neural Information Processing Systems*. MIT Press, 2005, pp. 793–800.

[34] S. Jain, S. Lange, and S. Zilles, "Towards a better understanding of incremental learning," in *Algorithmic Learning Theory*, ser. Lecture Notes in Computer Science, J. Balczar, P. Long, and F. Stephan, Eds. Springer Berlin Heidelberg, 2006, vol. 4264, pp. 169–183.

[35] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri, "Actions as space-time shapes," in *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, vol. 2, oct. 2005, pp. 1395–1402.

[36] A. A. Chaaraoui, P. Climent-Pérez, and F. Flórez-Revuelta, "Silhouette-based human action recognition using sequences of key poses," *Pattern Recognition Letters*, vol. 34, no. 15, pp. 1799–1807, 2013, Smart Approaches for Human Action Recognition.

[37] ——, "An efficient approach for multi-view human action recognition based on bag-of-key-poses," in *Human Behavior Understanding*, ser. Lecture Notes in Computer Science, A. A. Salah, J. Ruiz-del Solar, C. Meriçli, and P.-Y. Oudeyer, Eds. Springer Berlin Heidelberg, 2012, vol. 7559, pp. 29–40.

[38] M. A. Giese and T. Poggio, "Neural mechanisms for the recognition of biological movements," *Nature Reviews Neuroscience*, vol. 4, no. 3, pp. 179–192, 2003.

[39] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 26, no. 1, pp. 43 – 49, 1978.

[40] R. Li, J. Lu, Y. Zhang, and T. Zhao, "Dynamic adaboost learning with feature selection based on parallel genetic algorithm for image annotation," *Knowledge-Based Systems*, vol. 23, no. 3, pp. 195–201, 2010.

[41] S. Singh, S. Velastin, and H. Ragheb, "MuHAVi: A multicamera human action video dataset for the evaluation of action recognition methods," in *Advanced Video and Signal Based Surveillance (AVSS), 2010 Seventh IEEE International Conference on*, 2010, pp. 48–55.

[42] B. Saghafi and D. Rajan, "Human action recognition using pose-based discriminant embedding," *Signal Processing: Image Communication*, vol. 27, no. 1, pp. 96–111, 2012.

[43] L. Shao and X. Chen, "Histogram of body poses and spectral regression discriminant analysis for human action categorization," in *British Machine Vision Conference (BMVC), Aberystwyth, UK*, vol. 4, 2010.

Alexandros A. Chaaraoui was born in Braunschweig, Germany, in 1987. He received the BSc. and MSc. degrees in computer science respectively in 2010 and 2011, and the PhD. degree with the highest distinction (*cum laude*) and International Honourable Mention in 2014 from the University of Alicante, San Vicente del Raspeig, Spain.

During his studies, he performed several internships in the industry and in academic institutions. In 2010, he was awarded with a Microsoft EMEA internship in the Business Division which was carried out in Redmond, WA. He is currently a post-doctoral researcher of the Domotics and Ambient Intelligence group, and research fellow at the Computer Technology Department of the University of Alicante. His main research areas are human behaviour understanding, computer vision and ambient intelligence.

Dr. Chaaraoui received the Award for Outstanding Graduate of the Year 2010 from the National Board of Computer Science Engineers in the Valencian Community. He is also a student member of the IEEE.

Francisco Flórez-Revuelta (M'01-SM'13) was born in Algeciras, Spain, in 1970. He received the University degree in computer engineering from the Polytechnic University of Valencia, Spain, in 1994 and the PhD. degree in computer engineering from the University of Alicante, Spain, in 2001.

He is a Senior Researcher with the Faculty of Science, Engineering and Computing, Kingston University, United Kingdom. Between 1995 and 2011 he hold different positions at the University of Alicante, Spain, where since 2003 he was an Associate Professor (now on leave). His main research work is focused on ambient assisted living: person-environment interaction, computer vision, and support to the activities of daily living of elderly and/or disabled people. This is the area of application of his research background: computational intelligence (neural networks, evolutionary computation), computer vision, home automation, and assistive technologies. He is the author of more than 50 journal and conference papers in these fields.

Dr. Flórez-Revuelta is also a member of the Association for Computing Machinery. In 2011 he was awarded a Marie Curie Intra-European Fellowship by the European Commission.