



Universidad
Carlos III de Madrid



This is a postprint version of the following published document:

Baldominos, A.; Isasi, P.; Saez, Y. (2015). Monte Carlo schemata searching for physical activity recognition. *7th International Conference on Intelligent Networking and Collaborative Systems (INCoS 2015), proceedings*, pp. 176-183. DOI: <https://doi.org/10.1109/INCoS.2015.24>

© 2015 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

Monte Carlo Schemata Searching for Physical Activity Recognition

Alejandro Baldominos, Pedro Isasi and Yago Saez
Computer Science and Engineering Department
Universidad Carlos III de Madrid
28911 Leganes (Spain)

abaldomi@inf.uc3m.es, isasi@ia.uc3m.es, ysaez@inf.uc3m.es

Bernard Manderick
Artificial Intelligence Lab
Free University of Brussels
1050 Brussels (Belgium)
bmanderi@vub.ac.be

Abstract—Medical literature have recognized physical activity as a key factor for a healthy life due to its remarkable benefits. However, there is a great variety of physical activities and not all of them have the same effects on health nor require the same effort. As a result, and due to the ubiquity of commodity devices able to track users' motion, there is an increasing interest on performing activity recognition in order to detect the type of activity carried out by the subjects and being able to credit them for their effort, which has been detected as a key requirement to promote physical activity.

This paper proposes a novel approach for performing activity recognition using Monte Carlo Schemata Search (MCSS) for feature selection and random forests for classification. To validate this approach we have carried out an evaluation over PAMAP2, a public dataset on physical activity available in UCI Machine Learning repository, enabling replication and assessment. The experiments are conducted using leave-one-subject-out cross validation and attain classification accuracies of over 93% by using roughly one third of the total set of features. Results are promising, as they outperform those obtained in other works on the same dataset and significantly reduce the set of features used, which could translate in a decrease of the number of sensors required to perform activity recognition and, as a result, a reduction of costs.

Keywords—activity recognition; Monte Carlo Schemata Search; feature selection; classification

I. INTRODUCTION

Medical literature have extensively explored the positive effects of physical activity. These effects involve benefits for the society, from school-aged children and youth [1] and teenagers and adolescents [2], [3] to adults and the elderly [4]. On the other hand, lack of physical exercise may lead to the appearance of several disorders [5], [6].

Nowadays, there is an increasing interest of recognizing the physical activity performed by a subject, not only due to medical reasons, but as a way to provide self-awareness and credit subjects for their effort, this being a key requirement for developing systems that promote physical activity [7]. This interest lead to the rise of the field of *activity recognition*, which is also gaining relevance due to the availability of commodity hardware (i.e. smartphones, wearables...) which are each day owned by more people and have sensors able to track the user's motions, providing enough information to detect the type of physical activity performed by the subject.

An extensive survey of techniques used for sensor-based activity recognition can be found in a recent work from Chen et al. [8].

Evidence of this increasing interest can be found in the number of patent registrations in the recent past which aim at monitoring and quantifying physical activity of final users [9], [10], as well as at recognizing the activity [11] providing in some cases customized personal training [12], [13].

This paper proposes an approach for performing supervised learning of a classification model which will be able to predict the activity of a new user in real-time. To improve the results feature selection will be carried out by means of a new method based on Monte Carlo Tree Search called Monte Carlo Schemata Search (MCSS). The feature selection step could be viewed as a binary combinatorial problem where each feature is either selected or not from the whole set in order to achieve better results. A trade-off between classification accuracy and the number of required sensors is encountered when trying to increase the classification accuracy while reducing as much as possible the number of sensors.

Binary combinatorial problems involves finding a binary string that represents the optimal combination of yes/no alternatives. The difficulty of this problems resides in the weakness of the nature of the binary representation. The Genetic Algorithm perspective, one of the most popular techniques for binary combinatorial problems, tries to overcome this problem with the implicit parallel evaluation of the schemata performed by the method.

However this implicit evaluation is not followed of a direct use of the schemata for generating new solutions, and consequently new and better schemata.

In this work we follow the idea of evaluating how promising a schema could be, in terms of its ability to generate good solutions. However, generating good schemata has many intrinsic problems. First, the space of schemata is much bigger that those of the solutions. In a binary combinatorial problem of size L , the searching space has a size of 2^L while the size of the schemata space is 3^L . So finding good schemata must be much harder than finding good solutions. Second, schemata evaluation is a difficult task because each schemata contains many examples, whose number grows up

exponentially with the size of the schemata, and they have as well a hierarchical structure that makes them to share many examples, or even being one’s subset of other.

To overcome the above feebleness we propose the use of the potential of Monte Carlo Tree Search (MCTS) to generate good candidates because:

- They could deal with incomplete information. They need not to have complete information about the efficiency of the candidates, they can estimate that efficiency from very few examples, and can improve those estimations as more information is available.
- They use a tree structure that fits very well the intrinsic nature of the schemata

Therefore, we propose a new method for generating potential good schemata in terms of their capacity to represent good solutions. The method constructs a network whose nodes are binary schema from which solutions for any optimization problems could be generated. The Monte Carlo Upper Confidence Policy (UCP) will produce gradually more promising schemata, allowing for the abrupt reduction of the search space and the maintenance of a good compromise in the generation of good solutions.

II. MONTE CARLO TREES FOR SCHEMATA GENERATION

Monte Carlo Tree search methods have been taken an increasing attention in the last 15 years. Originally his best success was in the context of the game of Go [14], where its performance was bigger that any other automatic player method so far. Since then, MCTS have been extremely successful in designing expert computer players for many other two players games, like Hex [15], [16], Kriegspiel [17], and Poker [18]. Moreover, MCTS has been shown to outperform classic alpha-beta search even in games where good heuristic evaluations are difficult to obtain.

In recent years, some other domains different from games have been addressed, mainly combining traditional ideas from *minimax* search with the construction of the trees to adapt the procedure to planning domains or resource allocation [19]. A more extensive survey of MCTS methods and applications was published by Browne et al. [20].

In general, the application domains must conform to the constraints imposed by the representation systems that, in this case, are trees whose nodes are states, and branches meaning transitions between states. This restriction has a direct impact on the scope of the method which is reduced considerably. In this work we go further by adapting the Monte Carlo Tree Search method to make it able to optimize any combinatorial optimization problem. We produce a new binary combinatorial method that we called Monte Carlo Schemata Search (MCSS) [21], because it stands in the idea of using binary schemata as states in the Monte Carlo searching tree.

In this approach we use the MCTS technique in its basic formulation letting the branches of the tree be the schemata

of a binary string where “don’t care” symbols (*) are used to indicate the possibility of either a one or a zero, therefore representing a subset of all possible solutions. The root of the tree is the most general schema composed in all their positions by the “don’t care” symbol.

In the first level we generate all possible schemata that have just one fixed position, either a zero or a one. If we assume a binary codification then the number of successors of each node is $2 \times L$, being L the length of the individual.

From the root, we build a MCT in which each successor of a node is one level more specific than the parent. To generate a descendant node, we assign either a zero or a one in the place of a random “don’t care” position. This increase in specificity continues until we reach the greatest level consisting in having no “don’t care” symbols at all, i.e., we obtain a solution.

The generation of the tree is performed by selecting and expanding a node per iteration, making the tree grow in an unstructured and unbalanced way, taking into account the estimation of the nodes as good candidates to generate good solutions. In each iteration all the nodes present in the tree could be considered for selection, except those which have been expanded in all their descendants.

The procedure is performed in four phases:

Selection: The more promising unexpanded node of the tree must be selected (Figure 1a). A tree policy is designed to define the meaning of being a promising node. There are many tree policies, in this work we propose the use of the Upper Confidence Policy (UCP), but any other could be used without further modifications in the method. The UCP entails assigning a nomination value to each node. This nomination value is composed of two terms:

- An estimation value of the quality of the schema, computed as:

$$f(S_k) = \frac{\sum_{I_i \in S_k} f(I_i)}{n}$$

where S_k is the schema to be estimated, I_i some individuals represented by that schema that constitutes the sampling set and that are generated randomly, and n the size of that sampling set, $n = |I_i|$.

- A term to encourage the exploration of less visited schemata, computed as:

$$C \sqrt{\frac{\log N_k}{n_k}}$$

where N_k is the number of individuals used for computing the estimation of the parent, and n_k is the number of individuals used for computing the estimation of the current node. Note that this term increases when n_k is small. A small value of n_k means that the schema S_k has not been visited very often, and represents unexplored searching space areas, that could have good potential.

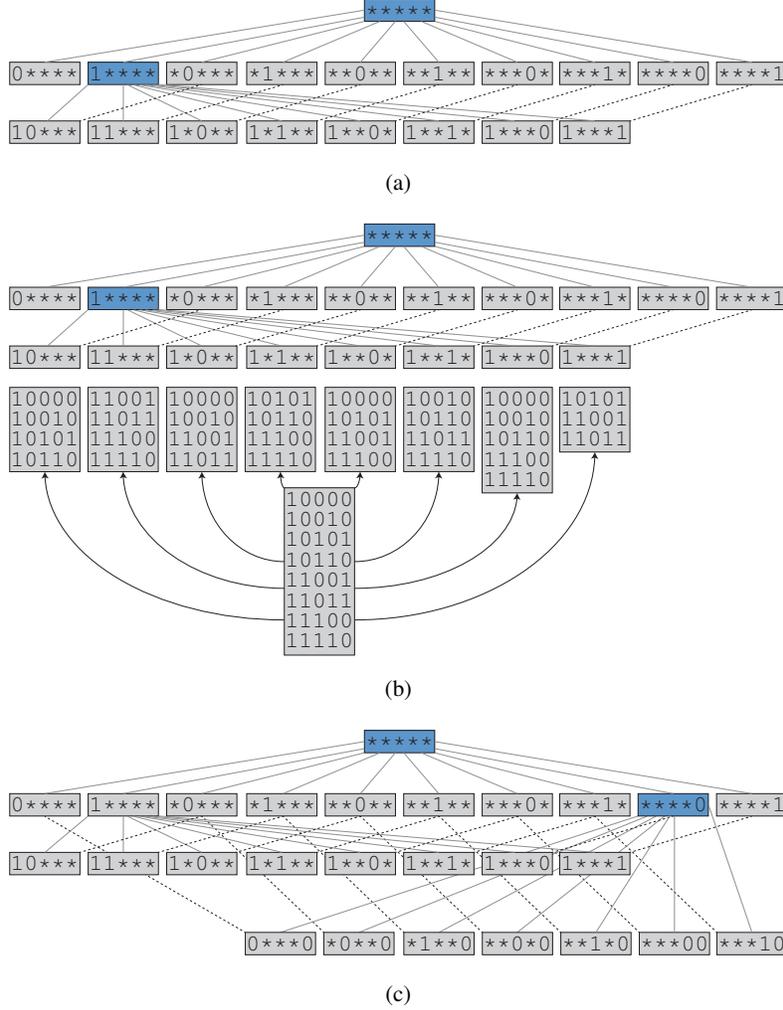


Figure 1: Procedure to generate the schemata searching net

The final value for the importance ($M(S_k)$) of a node k is then:

$$M(S_k) = f(S_k) + C \sqrt{\frac{\log N_k}{n_k}}$$

First term representing exploitation, and second term exploration. C is a parameter to balance between exploitation and exploration, as desired.

The selection procedure is performed in three steps:

- We start from the root node, labelling it as the current node. At each step, the best offspring node of the current node, in terms of above importance value, is assigned as the new current node
- If this new current node has already expanded all of their possible offspring, we repeat the process, choosing as the current node one of its offspring's
- If, by the opposite, it has some unexpanded offspring's,

then one of them is selected randomly and returned as the definitive selected one

Expansion: The selected node is expanded, generating one or many descendants, following some expansion policy, usually at random. When generating the descendant of a node, we must take into account that each node has more than one ascendant because it could match with more than one schemata. Therefore the generated nodes are linked with all the schemata of the previous level that are more general and contains it. Those links are important for the next backpropagation phase and allows a more precise estimation of the schemata.

Simulation: The new generated node must be evaluated in order to have an estimation of the accuracy of the schema that it represents. For doing that, a set of solutions are generated randomly (Figure 1b), matching the schema, and the average of the accuracy of that set is used as the

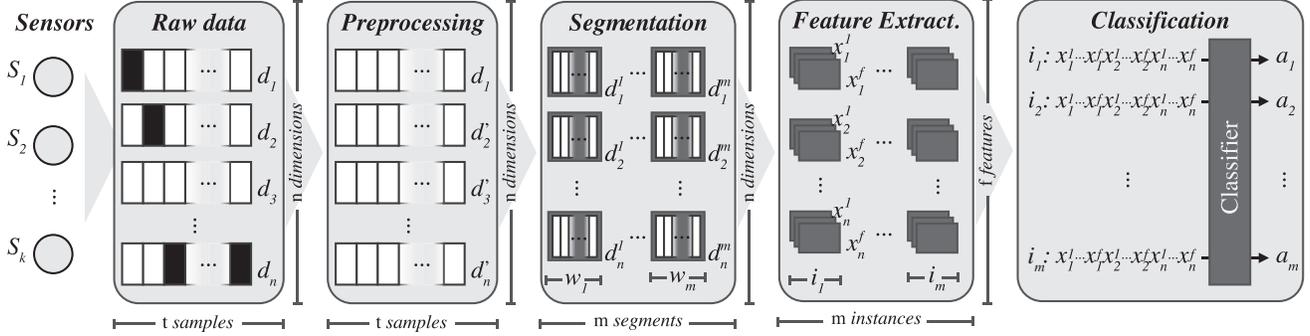


Figure 2: Steps involved in the activity recognition chain (ARC)

estimation value of the schema.

Backpropagation: All the evaluation values of all the nodes that are ancestors to the new created node are updated. The update of evaluation values of the nodes are performed following the equation:

$$f(S_k) = \frac{(t \cdot f(S_k)) + v(S_k)}{t + 1}$$

where $f(S_k)$ is the evaluation value of schema S_k , t is the number of times that schema S_k has been updated until now, and $v(S_k)$ is the average value of the evaluations of the individuals in the sampling set.

$$v(S_k) = \frac{\sum_{I_i \in S_k} f(I_i)}{n}$$

III. ACTIVITY RECOGNITION SYSTEMS

Activity recognition involves the performance of a sequence of steps which constitute the so-called activity recognition chain (ARC) [22], a general-purpose framework for acquiring data and building and evaluating activity recognition systems. Figure 2 shows the steps involved in the ARC, which include data acquisition from sensors, signal preprocessing and segmentation, feature extraction and the training of a classification model. This section elaborates on the work carried out for each of these steps.

A. Data Acquisition

The first step of the ARC involves obtaining physical activity data, a task which often involves setting up a protocol for a set of subjects to perform a sequence of activities while their motions are being tracked (either by ad-hoc sensors or by general-purpose devices such as smartphones). As this acquisition is expensive, for this work the PAMAP2 Physical Activity Monitoring dataset is used [23], [24], [25], [26], [27], [28], [29], which is publicly available at UCI Machine Learning Repository.

This dataset contains labeled information about physical activity performed by nine different subjects wearing a heart rate monitor and three wireless Intertial Measurement Units (IMUs) located over the wrist of the dominant arm, on the

Table I: PAMAP2 attributes extracted from IMUs

1	temperature ($^{\circ}\text{C}$)
2-4	3D-acceleration data (ms^{-2}), scale: $\pm 16\text{g}$, resolution: 13-bit
5-7	3D-acceleration data (ms^{-2}), scale: $\pm 6\text{g}$, resolution: 13-bit
8-10	3D-gyroscope data (rad/s)
11-13	3D-magnetometer data (μT)
14-17	orientation (invalid data)

chest and on the dominant side's ankle respectively. The data acquisition protocol [23] establishes that all subjects must perform the next activities in a certain order and must spend the same fixed time in the same exercise: lying, sitting, standing, ironing, vacuum cleaning, ascending stairs, descending stairs, walking, nordic walking, cycling, running and rope jumping.

The nine subjects (eight males and one female) taking part in the data acquisition step are aged 27.22 ± 3.31 years and have a BMI of $25.11 \pm 2.62 \text{ kgm}^{-2}$, one being left-handed and the others being right-handed [24]. While some subjects may show slight deviations with the protocol (mostly due to problems with the hardware setup and connectivity leading to minor losses of information or acceptable timing inconsistencies), subject 9 is an extreme case as his data completely differs from the protocol. For this reason, subject 9 is ignored for the experiments carried out in this paper.

The dataset provides 53 attributes, with IMUs generating 17 attributes each as shown in table I, being the other two a timestamp and the heart rate provided by the HR monitor. As data is labeled, each record in the dataset contains these 53 attributes plus the class (the physical activity).

The IMUs and the heart rate monitor have a sampling frequency of 100Hz and 9Hz respectively. As a result, records are generated every centisecond (leading to 100 samples generated by each second of subject activity), and heart rate information is unavailable in about 91% of the dataset instances.

B. Signal Preprocessing

This stage will receive the dataset samples as an input and will output a new set of instances, after performing the following actions:

- Removing the timestamp. This is an important step as the timestamp could constitute an identifier which would add bias to the classifier: as there is a fixed protocol, knowing the time at which an activity was performed should be enough to accurately determine the activity itself.
- Removing orientation data (12 attributes, 4 per IMU) as authors state that these attributes are invalid or irrelevant for this data collection [30].
- Completing the missing values (NaN values in the original dataset) by estimating their real values. For this step, we have considered that these values are unchanged from the previous available value. This is a good approximation as values are not expected to change significantly within one centisecond (or decisecond in the case of the heart rate).
- Removing instances labeled as *transition* which do not correspond to any activity, but rather the time after one activity ends and before the next starts.

When the preprocessing stage is completed, the transformed dataset reduces its dimensionality to 40 features plus the activity.

C. Signal Segmentation

So far, data is in the time domain (each record contains the signals provided by sensors in a given point in time). This format is suitable for training a classifier, but it is expected that higher accuracies can be attained if instances themselves capture temporal information. With this idea in mind, in the segmentation stage we will convert signals into the frequency domain by applying a Discrete Fourier Transform (DFT) over the dataset.

To do so, a sliding window of size 512 (corresponding to 5.12 seconds of physical activity data) is determined, and then the Fast Fourier Transform (FFT) is computed over the window, returning for each signal its transform in the frequency domain composed of 512 values. This sliding window is moved one instance a time and the process is repeated for each window. Additionally, we prevent instances belonging to different activities from being included in the same window.

D. Feature Extraction

Once signals are transformed into the frequency domain, they must be processed in order to extract the complete set features composing the new dataset.

For generating this set of features, a statistical summary of the 512 values obtained after computing the DFT is computed, generating 7 attributes for each original signal: the mean and the median, the standard deviation, the maximum

Table II: Number of features for each value of τ

τ	# of features	τ	# of features
1	280	5	102
2	271	6	46
3	243	7	17
4	188	8	4

and minimum values, and the 25% and 75% percentiles. Each window in the segmentation stage generates an instance in the dataset, which contains 280 features plus the class. The fact that the dataset dimensionality has increased by a factor of 7 will lead to higher training times and could potentially reduce the model accuracy due to the effects of overfitting. In order to reduce the number of features to be used for training a classifier, we use Monte Carlo Schemata search (MCSS). This process will first optimize a distinct feature set for each subject, where this set is represented by a binary chromosome of size 280, each position representing a feature and whether it is considered (1) or discarded (0) when training the model.

In particular, a local optimization of the feature set for each user will be pursued which will later be used to obtain an approximation of the best feature set for all the users. For this local optimization, the evaluation is defined as the classification accuracy over a test set.

Later, the best feature sets for each subject are aggregated in order to move from a local to a global optimization, so that the same set of features is applied to all subjects. To do so, a threshold τ is defined, so that an attribute a_i is chosen only if at least τ bits for the corresponding genes in all chromosomes are 1, i.e.:

$$a_i = \begin{cases} 1, & \text{if } \sum_{n=\{1..8\}} g_i^n \geq \tau \\ 0, & \text{otherwise} \end{cases}$$

where a_i is the i -th attribute and g_i^n is the i -th gene of the best individual for the n -th fold.

Table II shows the number of final features after this process is applied for all possible different values of τ . It should be noted than as the value of τ grows, the resulting feature set becomes a subset of the previous feature sets. Also, it can be seen that the feature set for $\tau = 1$ contains all the features, which means that there is no attribute for which the value for its corresponding gene is 0 for all chromosomes.

E. Classification

The problem of physical activity recognition is closely related to the field of supervised learning, where a classifier is trained using a set of instances for which the class is known in advance, being the class the physical activity to be recognized.

As a preliminary experimentation, standard machine learning algorithms widely addressed in the literature have

been compared before any kind of feature optimization takes place. Naive Bayes, C4.5 and Random Forest are used for establishing baseline results. As it will be shown in the next section, Random Forest significantly outperforms the other methods for all users, and as a result this technique will be used for the rest of the work, including the evaluation computation.

IV. RESULTS

To evaluate the proposed approach to activity recognition, leave-one-subject-out (LOSO) cross validation is used for the experiments. This setup involves the execution of 8 different experiments, each one choosing a different subject for the test set and the remaining 7 subjects for composing the training set.

Before feature optimization is carried out, classification models are trained and tested with each user using the whole set of 280 features in order to establish baseline results. Naive Bayes, C4.5 and Random Forest are used, and the results are shown in table III, which shows that Random Forest outperforms its competitors.

Once these baseline results are stated, feature selection can be performed using Monte Carlo Schemata search as described in section III-D. Table IV shows the accuracy achieved for each subject (which corresponds with the highest evaluation for each fold). The results of the local optimization are quite close to 100%, but they fail to provide a final set of features valid for all users, and thus a user-independent classifier cannot be built until we generalize all the per-subject feature sets into a subject-independent one, for which the procedure described in the previous section will be performed.

The results for each different value of τ are shown in table V, where each accuracy for each combination of subject and value of τ is the average of 30 different executions, each one training the model with a random sample of 10% of the training set.

Table V clearly shows how the accuracy evolves as the value of τ increases. This behavior is certainly revealing, as it shows that accuracy may increase when the set of features is reduced (see the case of $\tau = 1$ vs. $\tau = 2$), but more interestingly the average accuracy for all subjects barely changes for values of τ between 1 and 5, as the maximum difference is smaller than 0.6 percentage points. It is only when τ is greater than 6 than the accuracy starts to worsen in a significant manner. Bringing back the results from table II, this fact means that feature selection using MCSS allows reducing the total set of features from 280 to 102 with a very small impact in the accuracy, which would result in a decrease of the total number of sensors worn by the subjects.

We can compare these results with those attained by Reiss and Stricker [25] when they introduce the dataset. While they use different experimentation setups, the one comparable with our results is the so-called “all activity” recognition.

Additionally, they also use LOSO cross validation, thus making this comparison feasible. In their work, they reach the maximum accuracy using k-nearest neighbors, achieving 89.24%. Our results outperform that accuracy in about four percentage points, and due to feature selection we are able to improve the accuracy while reducing the set of features to slightly more than one third of the original set.

V. CONCLUSION

This paper has proposed a novel approach for performing human activity recognition. While experts agree on the fact that physical activity is a key aspect of human health, not all physical activities have the same effect on health nor require the same effort to be performed by the subjects. For this reason, a system able to accurately recognize the physical activity performed by a user sounds quite promising, as it is able to increase awareness about his health and habits.

The development of an activity recognition system is described in this paper by thoroughly detailing each stage in the activity recognition chain (ARC). First we explain how the data is acquired, preprocessed and segmented. In particular this paper uses the PAMAP2 Physical Activity Monitoring dataset, which is public available for download at UCI Machine Learning repository. Over this dataset we perform some basic preprocessing and the apply a DFT over a sliding window in order to transform the signals from the time domain to the frequency domain.

Later, features are extracted from the transformed signals by performing a statistical summary. An early training of classification models is performed at this stage using standard machine learning techniques, before any kind of feature set optimization takes place, in order to establish baseline results that are obtained using leave-one-subject-out (LOSO) cross validation in order to avoid bias. An average classification accuracy of 93.5% is attained.

Later, Monte Carlo Schemata search (MCSS) is used in order to optimize the feature set. MCSS has been designed for binary combinatorial optimization requiring low domain knowledge and allowing a fast and big reduction of the searching space, where not too small differences in accuracy are needed, as in the case of this problem where solutions around 94% of accuracy are attained. A local optimization of the feature set of each subject lead to accuracies about 99%, providing an uniform level of forecasting more suitable for every individual. The locally optimized feature sets are then used to generate a new subject-independent optimized feature set by defining a threshold τ , which still shows very high levels of accuracy. More specifically, the MCSS optimization lead to average accuracies up to 93.72% but more interestingly, it provides an average accuracy of 93.17% when $\tau = 5$, revealing a classification ability almost as good as before feature optimization was performed, but using only 102 features, roughly a third part of the original set. These

Table III: Classification accuracy for each fold using the whole feature set

	Subj. 1	Subj. 2	Subj. 3	Subj. 4	Subj. 5	Subj. 6	Subj. 7	Subj. 8	Avg.
Naive Bayes	73.29%	65.79%	94.19%	96.66%	90.91%	88.33%	91.73%	89.50%	86.30%
C4.5	78.31%	80.48%	74.70%	84.39%	71.27%	89.59%	72.19%	82.85%	79.23%
Random Forest	85.82%	89.73%	92.83%	96.02%	92.83%	96.50%	97.35%	97.03%	93.51%

Table IV: Classification accuracy after optimizing the feature set for each fold using random forest

Subj. 1	Subj. 2	Subj. 3	Subj. 4	Subj. 5	Subj. 6	Subj. 7	Subj. 8	Avg.
96.60%	97.00%	98.99%	99.94%	98.74%	98.98%	99.63%	99.61%	98.69%

Table V: Classification accuracy for each fold using the reduced feature set and random forest

	Subj. 1	Subj. 2	Subj. 3	Subj. 4	Subj. 5	Subj. 6	Subj. 7	Subj. 8	Avg.
$\tau = 1$	88.60%	87.49%	93.88%	95.96%	93.82%	96.08%	96.52%	96.79%	93.64%
$\tau = 2$	89.09%	87.94%	93.79%	96.95%	93.88%	95.82%	96.76%	95.67%	93.74%
$\tau = 3$	88.89%	87.33%	93.98%	96.90%	92.60%	95.97%	96.16%	96.20%	93.50%
$\tau = 4$	88.00%	86.93%	93.21%	97.04%	94.07%	96.07%	96.37%	96.35%	93.50%
$\tau = 5$	89.59%	86.72%	94.41%	96.04%	93.73%	94.06%	95.21%	95.62%	93.17%
$\tau = 6$	88.86%	78.28%	86.81%	86.14%	83.34%	86.44%	82.78%	89.73%	85.30%
$\tau = 7$	79.66%	46.16%	85.86%	62.03%	83.99%	76.34%	71.97%	82.34%	73.54%
$\tau = 8$	47.16%	35.69%	48.24%	34.20%	50.27%	43.21%	40.35%	50.62%	43.72%

results also outperformed those established by Reiss and Stricker when they introduced the PAMAP2 dataset.

ACKNOWLEDGMENT

This work was partially funded by European Union's CIP Programme (ICT-PSP-2012) under grant agreement no. 325146 (SEACW project).

REFERENCES

- [1] Janssen and LeBlanc, "Systematic review of the health benefits of physical activity and fitness in school-aged children and youth," *International Journal of Behavioral Nutrition and Physical Activity*, vol. 7, no. 40, 2010.
- [2] S. J. H. Biddle and M. Asare, "Physical activity and mental health in children and adolescents: a review of reviews," *British Journal of Sports Medicine*, vol. 45, no. 11, pp. 886–895, 2011.
- [3] A. Merglen, A. Flatz, R. E. Bélanger, P.-A. Michaud, and J.-C. Suris, "Weekly sport practice and adolescent well-being," *Archives of Disease in Childhood*, vol. 99, pp. 208–210, 2014.
- [4] W. Chodzko-Zajko, "Exercise and physical activity for older adults," *Universal Access in the Information Society*, vol. 3, no. 1, pp. 101–106, 2014.
- [5] L. K. Ejsing, U. Becker, J. S. Tolstrup, and T. Flensburg-Madsen, "Physical activity and risk of alcohol use disorders: results from a prospective cohort study," *Alcohol and Alcoholism*, vol. 50, no. 2, pp. 206–212, 2014.
- [6] J. A. Knight, "Physical inactivity: associated diseases and disorders," *Annals of Clinical & Laboratory Science*, vol. 42, no. 3, pp. 320–337, 2012.
- [7] S. Consolvo, K. Everitt, I. Smith, and J. A. Landay, "Design requirements for technologies that encourage physical activity," in *Proc. SIGCHI Conf. on Human Factors in Comput. Syst. (CHI'06)*, 2006, pp. 457–466.
- [8] L. Chen, J. Hoey, C. Nugent, D. Cook, and Z. Yu, "Sensor-based activity recognition," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 42, no. 6, pp. 790–808, 2012.
- [9] C. A. Vock, P. Flentov, and D. M. Darcy, "Activity monitoring systems and methods," U.S. Patent, 2013.
- [10] S. G. J. Yuen, J. Park, and E. N. Friedman, "Activity monitoring systems and methods of operating same," U.S. Patent, 2013.
- [11] P. R. Kahn, A. Kinsolving, M. A. Christensen, B. Y. Lee, and D. Vogel, "Human activity monitoring device with activity identification," U.S. Patent, 2015.
- [12] D. Morris, I. Kelner, F. Shariff, D. Tom, T. S. Saponas, and A. Guillory, "Personal training with physical activity monitoring device," U.S. Patent, 2015.
- [13] K. L. White, M. L. Orenstein, J. Campbell, C. S. Self, E. Walker, M. Micheletti, G. McKeag, J. Zipperer, and M. Lapinsky, "Activity recognition with activity reminders," U.S. Patent, 2015.
- [14] B. Brüggmann, "Monte carlo go," Physics Department, Syracuse University, Tech. Rep., 1993.
- [15] B. Arneson, R. Hayward, and P. Henderson, "MoHex wins Hex tournament," *International Computer Games Association Journal*, vol. 32, no. 2, pp. 114–116, 2009.
- [16] —, "Monte Carlo tree search in Hex," *IEEE Transactions on Computational Intelligence and AI in Games*, vol. 2, no. 4, pp. 251–258, 2010.
- [17] P. Ciancarini and G. P. Favini, "Monte Carlo tree search in Kriegspiel," *Artificial Intelligence*, vol. 174, no. 11, pp. 670–684, 2010.
- [18] J. Rubin and I. Watson, "Computer poker: a review," *Artificial Intelligence*, vol. 175, no. 56, pp. 958–987, 2011.

- [19] D. Bertsimas, J. D. Griffith, V. Gupta, M. J. Kochenderfer, V. V. Misic, and R. Moss, "A comparison of Monte Carlo tree search and mathematical optimization for large scale dynamic resource allocation," *CoRR*, vol. abs/1405.5498, 2014.
- [20] C. Browne, E. Powley, D. Whitehouse, S. Lucas, P. Cowling, P. Rohlfshagen, S. Tavener, D. Perez, S. Samothrakis, and S. Colton, "A survey of Monte Carlo tree search methods," *IEEE Transactions on Computational Intelligence and AI in Games*, vol. 4, no. 1, pp. 1–43, 2012.
- [21] P. Isasi, M. Drugan, and B. Manderick, "Schemata Monte Carlo network optimization," in *PPSN 2014 Workshop: In Search of Synergies between Reinforcement Learning and Evolutionary Computation*, 2014.
- [22] A. Bulling, U. Blanke, and B. Schiele, "A tutorial on human activity recognition using body-worn inertial sensors," *ACM Computing Surveys (CSUR)*, vol. 46, no. 3, pp. 33:1–33:33, 2014.
- [23] A. Reiss, M. Weber, and D. Stricker, "Exploring and extending the boundaries of physical activity recognition," in *Proc. IEEE Intl. Conf. Systems, Man and Cybernetics (ICSMC'11)*, 2011, pp. 46–50.
- [24] A. Reiss and D. Stricker, "Towards global aerobic activity monitoring," in *Proc. 4th Intl. Conf. Pervasive Tech. related to Assistive Environments (PETRA'11)*, 2011.
- [25] —, "Introducing a new benchmarked dataset for activity monitoring," in *Proc. Intl. Symp. Wearable Computers (ISWC'12)*, 2012, pp. 108–109.
- [26] —, "Creating and benchmarking a new dataset for physical activity monitoring," in *Proc. 5th Intl. Conf. Pervasive Tech. related to Assistive Environments (PETRA'12)*, 2012, pp. 108–109.
- [27] A. Reiss, D. Stricker, and I. Lamprinos, "An integrated mobile system for long-term aerobic activity monitoring and support in daily life," in *Proc. 11th IEEE Intl. Conf. Trust, Security and Privacy in Comp. and Comm. (TrustCom'12)*, 2012, pp. 2021–2028.
- [28] A. Reiss and D. Stricker, "Aerobic activity monitoring: towards a long-term approach," *Universal Access in the Information Society*, vol. 13, pp. 101–114, 2013.
- [29] A. Reiss, G. Hendeby, and D. Stricker, "A novel confidence-based multiclass boosting algorithm for mobile physical activity monitoring," *Personal and Ubiquitous Computing*, vol. 19, no. 1, pp. 105–121, 2014.
- [30] A. Reiss, "Personalized mobile physical activity monitoring for everyday life," Ph.D. dissertation, Technical University of Kaiserslautern, 2013.