STUDIA UNIV. BABEŞ–BOLYAI, INFORMATICA, Volume **LXI**, Number 1, 2016

A COMPARATIVE STUDY OF ARTIFICIAL INTELLIGENCE METHODS FOR KINECT GESTURE RECOGNITION

ALINA DELIA CĂLIN

Abstract. This paper analyses a natural interface sensor based gesture recognition for the purpose of capturing and using indirect user input during gaming and create a more personalised and enjoyable experience. We have compared 38 classifiers on our own database of 30 different body postures and analysed the results for the best performing of these, in terms of precision, accuracy and time. We have found that the best performing classifiers to use in a real-time system are SimpleLogistic, MultiClassClassifier and RandomForest. Also, next steps are discussed in terms of combining methods for more complex poses and gestures detection, extending the database of body postures and exploring as well the prediction potential of such a system.

1. Introduction

Considering some of the recent main uses of artificial intelligence in the games domain, like solving difficult games and adapting games to enhance user experience, this paper looks into the most practical non-entertainment uses of video games, such as learning (educational), rehabilitation (physical and cognitive therapies in healthcare) or solving world problems, like the case of minority games (economical). The present requirements are focused on improving human-computer interaction, into a more natural and intuitive way, and to make use also of the indirect input from the user and to adapt the system to their actual needs.

One major focus in this paper is game personalization, mostly from the point of view of making use of the newly developed interaction hardware, like the 3D Kinect camera. The next sections will present a review of the latest results obtained for using AI in games, especially for Kinect-based interaction,

Received by the editors: February 23, 2016.

2010 *Mathematics Subject Classification.* 68T50, 68T05.

1998 *CR Categories and Descriptors.* H.5.2 [**Information interfaces and presentation**]: User Interfaces – *Natural language*; I.2.1 [**Artificial Intelligence**]: Applications and Expert Systems – *Games*.

Key words and phrases. gesture recognition, AI, Kinect, personalised games.

like gesture recognition, for which a performance comparison of the most used methods is presented. Further, we have created our own database of poses and compared a range of 38 classifiers on two different interpretations of the dataset. These results are then analysed, with the purpose of deciding the most promising methods and directions to be followed for research in this domain.

2. Related Work

Artificial intelligence has been widely used to create and solve complex games. Some examples of the best results are based on methods like: neural networks, reinforcement learning, evolutionary algorithms, adversarial learning and digital pheromones. These are essential for creating competitive games features: user profiles, complex and realistic non-player characters, and mainly personalised gameplay and adaptive game difficulty for a better enjoyment and engagement of the user [2]. This is important because games can be used effectively for educational or medical purposes, as they engage the user and mask the serious educational or therapeutic purpose. These are essential for creating competitive games features: user profiles, complex and realistic non-player characters, for a better enjoyment and engagement of the user.

Bakkes et al. [2] specify the psychological foundation and motivation for creating personalized games and measures the effect on player satisfaction and engagement, from the perspective of eight different components of the game that can be adapted to the user: game space, mission/task, character, game mechanics, narrative, music/sound, player matching (in multiplayer games) and difficulty balancing. These aspects are of utmost importance when designing games, considering their power of entertainment and engagement from the user, provided that games are able to adapt these parameters according to every user's preferences automatically. For educational and clinical games, the more the users are engaged into playing the games, the more they will benefit from their educational, clinical or therapeutic purpose.

Recently developed hardware sensors, such as Microsoft Kinect, are able to integrate full body interactivity, making video games a great tool for rehabilitation in both physical and cognitive therapies. In this direction, further work would imply combining different approaches and even domains, in order to gain more on the whole and be able to create easily a personalised experience, by making use of the physical body language input provided by these sensors, while interacting with the system.

2.1. **Gesture recognition with Kinect.**

Kinect is a natural interface sensor created for gaming, but with a huge potential and perspectives for general computer interaction based on human body language, as it detects and track

20 human body joints in 3D. This data can be used for recognising body gestures, actions, poses, detecting face emotions, finger sign language, interactions with the environment and environment objects. For detecting user body language related emotions, Saha et al. [3] have compared classifiers k-nearest neighbour, SVM (Support Vector Machine), Neural Network with Back-Propagation Learning (NNBPL), Binary Decision Tree and Ensemble Tree Classifier. For a set of 5 gestures (scared, angry, happy, sad, relaxed) the best results were on Ensemble Tree (90%) and NNBPL (89%), followed by SVM (87%) and k-NN (86%). Results of up to 100% can be obtained when classifying a small number of very distinct poses (such as sit/stand/lie down with NNBPL, SVM, decision tree, and naive Bayes compared in [4]). But accuracy and precision are very dependent on the gestures to be recognized and the methods used. Wang et al. [5] obtain good results (85%-100% accuracy) using Hidden Markov Models (HMM) on a set of 8 distinct gesture (fly twice, wave hands twice, circle, heart, both pull, both push, Buddha gesture, Applaud 4 times) while [6] obtain 88,2% accuracy on a different set of 20 gestures using an proposed method that gave better results than HMM, Dynamic Temporal Warping (DTW), NN or action graph on bag of 3D points.

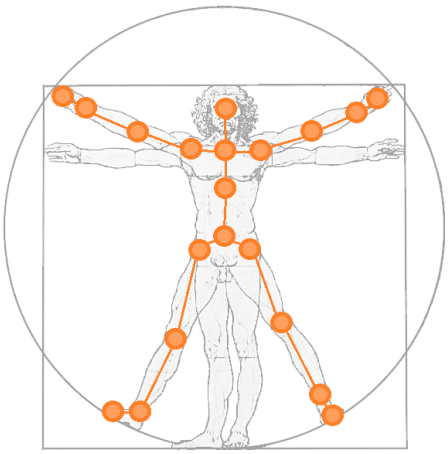

FIGURE 1. Kinect skeleton with the 20 body joints, adapted from [1].

3. Performance Comparison of Several Classifiers

Considering that poses represent a static body configuration and that gestures can be defined as a sequence of poses, the two should be approached in a distinct way for obtaining best results. In this paper we will focus on pose

detection, by studying a range of classifiers and comparing their performance results based on a database of poses that are likely to be meaningful in the context of interacting with a serious game, in order to translate the user's emotions and gestures, and personalise the system accordingly.

3.1. **Methods.** We have created a database containing 20 different poses extracted from two individuals with different body constitutions (one male, one female, with a difference in height of 11 cm), each pose having between 15 and 30 different entries (summing a total of 489 entries), and have used it for training and testing several classifiers using Weka 3.7 (a wide collection of machine learning algorithms) and 10-fold cross-validation [7]. Poses were represented by all the 20 joints provided by the Kinect sensor in 3D and indicate possible actions like talking on the phone, scratching the head (thinking), praying, hands crossed, hands out in wonder, hands on hips, hands up (winning), covering ears or in thinker pose, using one hand or both where applicable, with the user either standing or sitting. When comparing performance, we have taken into account precision and accuracy, but also the computing time, in order to establish their potential to be used in real time applications.

In order to avoid confusion of similar gestures, we have taken into account that some poses are much better recognized by the sensor while the user is standing, so for the poses where mainly the upper body was relevant in determining the pose, regardless of lower body position (sitting or standing), we have split the data into standing poses and sitting poses, obtaining a total of 30 distinct classes from the initial 20, with 15–20 entries each. We call this dataset 30S as it has 30 classes with sitting poses differentiated. The initial dataset with mixed sit/stand poses combined into 20 number of classes is called 20M. For the purpose of detecting similar poses that are most likely to be confused, we have summed up the confusion matrices of the top 11 classifiers for each dataset results accordingly.

3.2. **Results.** Results obtained from the classifiers are presented in Figures 2 and 3, from which we can observe that generally the best results in terms of precision, accuracy and time taken to build the model, for both datasets, are obtained with classifiers SimpleLogistic, MultiClassClassifier and RandomForest.

For the first dataset (20M), the best results are obtained by SimpleLogistic and LMT (accuracy 0.982, precision 0.983, but different times - 3.6 s and 19.51 respectively), followed closely by MultilayerPerceptron (accuracy 0.965, precision 0.967, time 19.99 s). Also, five classifiers (in decreasing order of performance: MultiClassClassifier, Logistic, RandomForest, RandomComettee and SMO) obtain vales above 0.9 for both parameters. Looking at the type of the classifiers with the best performance, we can see that all 4 from

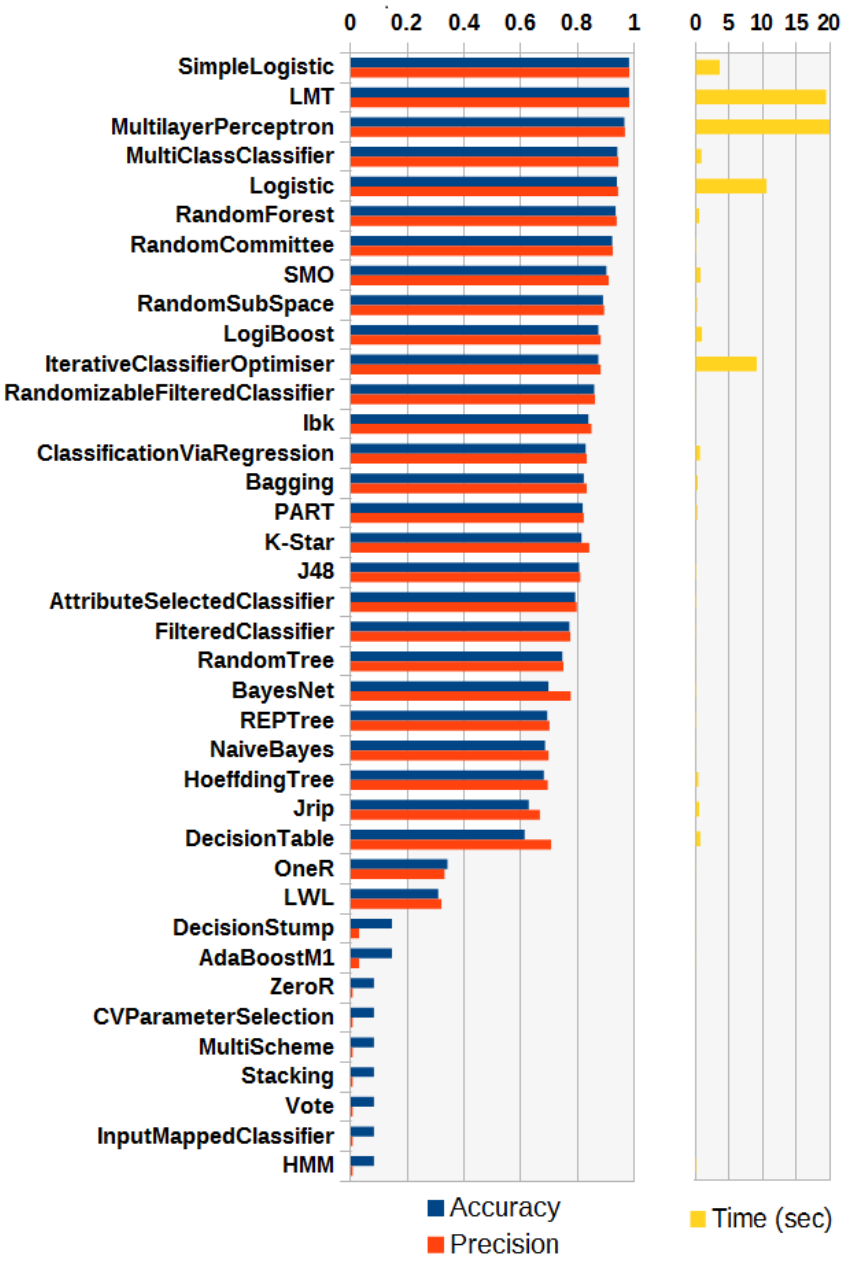

FIGURE 2. Clasificator data mixed DB (20 classes - 20M).

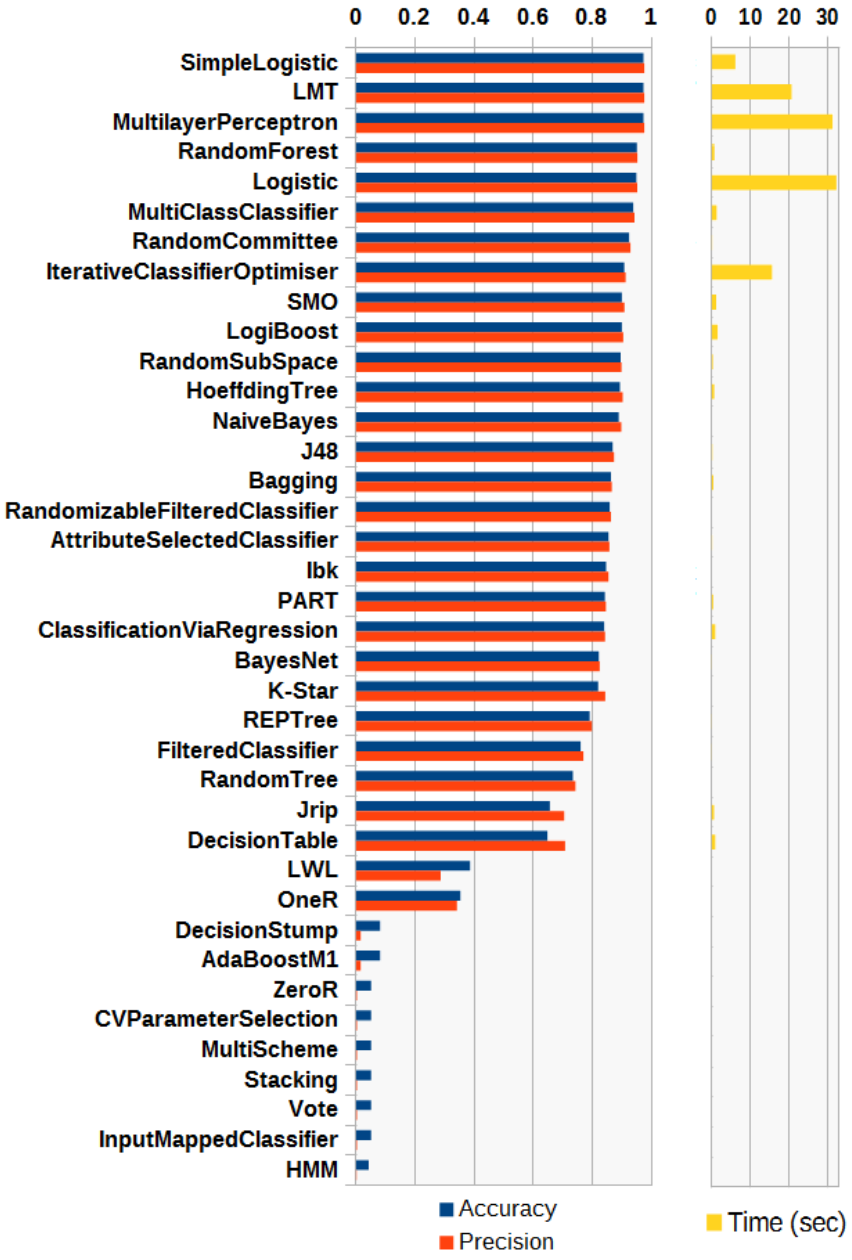

FIGURE 3. Clasificator data sitting DB (30classes - 30S).

Functions make the top, 2 are Trees and 2 are Meta classifiers. None of the Rules, Lazy or Bayes have good results (all are below 0.85 precision and accuracy).

	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t
a = HandsCrossed	388	0	3	1	0	0	0	0	0	45	0	1	0	0	0	0	0	0	0	2
b = HandsInPocket	0	150	0	7	0	0	0	0	0	2	0	1	5	0	0	1	0	10	0	0
c = HandsOnEars	5	0	347	0	1	0	0	0	1	3	2	0	0	0	0	0	2	0	2	0
d = HandsOnHips	3	21	1	360	0	0	0	2	3	0	0	2	0	3	0	0	0	1	0	0
e = HandsUpWin	0	0	3	0	154	0	2	0	0	0	2	0	0	2	0	0	2	0	0	0
f = HandUpLeft	0	0	0	0	0	269	0	1	0	0	5	0	0	0	0	0	0	0	0	0
g = HandUpRight	0	0	0	0	0	0	190	0	0	0	0	7	0	0	0	0	0	1	0	0
h = PhoneTalkLeft	0	0	0	3	0	6	0	357	0	5	9	0	2	0	0	1	0	2	0	0
i = PhoneTalkRight	2	2	0	2	0	0	8	0	328	2	0	19	0	0	0	0	0	0	0	0
j = Praying	36	0	1	0	0	1	0	0	0	347	0	0	0	0	0	0	0	0	0	0
k = ScreatchHeadLeft	1	0	2	0	0	2	0	8	0	0	383	0	0	0	0	0	0	0	0	0
l = ScreatchHeadRight	2	0	0	0	0	0	5	0	12	0	0	376	0	0	0	1	0	0	0	0
m = ServantPose	0	17	0	0	0	0	0	0	1	0	0	0	174	0	0	0	0	6	0	0
n = HandsOutWhy	0	0	0	4	1	0	0	2	0	2	0	0	0	310	0	0	0	0	0	0
o = ThinkerPose	2	0	2	0	0	0	0	0	0	0	0	0	2	0	90	2	0	0	1	0
p = Sitting	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	206	0	2	0	0
q = HandsUpWin	0	0	8	0	0	0	1	0	0	0	0	1	0	2	0	0	141	0	1	0
r = HandsDown	0	19	0	0	0	0	2	0	0	0	0	0	15	0	0	0	0	129	0	0
s = ThinkLeft	2	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	151	0
t = ThinkRight	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	2	139

FIGURE 4. Confusion matrix constructed by summing up all the confusion matrices of the first 11 best precision classifiers on the 20M dataset.

From the summed up confusion matrix in Figure 4 we can observe that the most confused poses are "praying" and "hands crossed", possibly because they are very similar and also because the Kinect sensor's accuracy is not very good when joints are inferred and very close to each other like in these two poses. Other common confusions are between "hands on hips", "hands down", " servant pose" and "hands in pocket" or between "scratch head right", "hand up right" and "phone talk right", mostly because of the similarity of the poses, but the confusion matrix is not completely symmetrical relatively to the first diagonal. Also we can notice that although "scratch head", "phone talk" and "hand up" are poses that can be done each with the left or with the right hand, only the right hand poses are being pertinently confused, which would suggest that the cause is not only the similarity between the poses, but also the pose database, in terms of number of entries and data noise.

	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	aa	ab	ac
a = HandsCrossed	201	1	1	0	0	0	0	0	0	20	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6
b = HandsInPocket	0	146	0	11	0	0	0	0	0	0	0	0	7	0	0	0	0	0	0	0	0	0	0	1	0	11	0	0	0
c = HandsOnEars	7	0	161	0	3	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4
d = HandsOnHips	4	19	0	157	0	0	1	0	0	0	0	0	2	0	1	0	0	0	0	0	0	0	0	0	0	1	1	0	1
e = HandsUpWin	0	0	4	0	155	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0
f = HandUpLeft	0	0	0	0	0	267	0	1	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0
g = HandUpRight	0	0	0	0	0	0	198	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
h = PhoneTalkLeft	0	1	0	0	0	4	0	168	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
i = PhoneTalkRight	0	2	0	0	0	0	5	0	154	0	0	15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
j = Praying	23	0	0	0	0	1	0	0	0	207	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
k = ScreatchHeadLeft	1	0	0	0	0	2	0	7	0	0	176	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
l = ScreatchHeadRight	0	0	2	0	0	0	2	0	18	0	0	174	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0
m = ServantPose	1	9	0	0	0	0	0	0	0	2	0	2	177	0	0	0	0	0	0	0	0	0	0	0	0	7	0	0	0
n = SitHandsCrossed	0	0	0	0	0	0	0	0	0	0	0	0	1	201	1	0	0	1	0	5	0	0	0	0	0	0	0	0	0
o = SitHandsOnEars	4	2	0	0	0	0	0	0	0	0	0	0	0	0	181	0	0	0	0	0	0	0	0	0	0	0	0	0	0
p = SitHandsOnHips	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	207	0	0	0	0	0	0	0	0	0	2	0	0	0
q = SitHandsOutWhy	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	152	0	1	0	0	1	0	0	0	0	0	0	0
r = SitPhoneTalkLeft	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	4	1	198	0	0	2	0	0	2	0	0	0	0	0
s = SitPhoneTalkRight	0	0	0	0	0	0	2	0	1	0	0	0	0	0	0	0	0	0	183	0	0	1	0	0	0	0	0	0	0
t = SitPraying	0	0	0	0	0	0	0	0	0	0	0	0	1	18	0	0	0	0	0	134	0	0	0	0	0	0	0	1	0
u = SitScreatchHeadLeft	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	0	0	203	0	0	0	0	0	0	0	0
v = SitScreatchHeadRight	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	196	0	0	0	0	0	0	0
w = ThinkerPose	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	97	2	0	0	0	0	0
x = Sitting	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	208	0	1	0	0	0
y = SitHandsUpWin	0	1	2	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1	0	1	0	0	147	0	0	0	0
z = HandsDown	0	18	0	0	0	0	2	0	0	0	0	0	10	0	0	0	0	0	0	0	0	0	0	0	0	135	0	0	0
aa = HandsOutWhy	2	0	0	4	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	157	0	0
ab = ThinkLeft	3	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	150	0
ac = ThinkRight	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	139

FIGURE 5. Confusion matrix constructed by summing up all the confusion matrices of the first 11 best precision classifiers on the 30S dataset.

For the second dataset (30S), SimpleLogistic and LMT remain on top but have lower performance equal with MultilayerPerceptron (accuracy 0.973, precision 0.76), time being the parameter that decides the top order (6.11 s, 20.68 s and 31.31 s respectively). In this case there are seven more classifiers above 0.9 accuracy and precision, five of which are present in the over 0.9 top for the 20M dataset, but in a different decreasing order of performance: RandomForest, Logistic, MultiClassClassifier, RandomComettee, IterativeClassifierOptimiser, SMO and LogiBoost. As type of classifiers, the top is composed of the 4 Functions, 2 Trees and 4 Meta classifiers. The confusion matrix in Figure 5 shows almost the same confusions as for the 20M dataset: "hands crossed" with "praying"; "hands on hips" with "hands in pocket", "servant pose" and "hands down"; "scratch head right" with "phone talk right". There is no relevant confusion between the sitting and standing poses or between several sitting poses, and, as the sitting poses in 30S are extracted from existing entries in the 20M dataset, this would explain the increase in classifiers' accuracy and precision for the 30S dataset.

By comparing the results obtained from the two datasets, we can generally observe that a larger number of classes requires more computing time and it also increases precision for most of the classifiers (as it clarifies the pose as a standing or sitting one, as we always take into consideration the entire body), except for the two top ones for the 20M (see Figures 6 and 7).

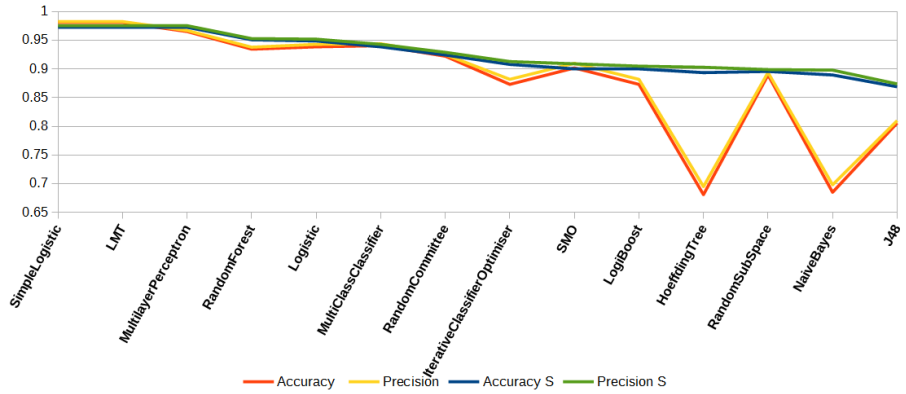

FIGURE 6. Top 14 Classifiers: Precision and Accuracy for the 30S dataset (marked with S, green and blue) and 20M data set (yellow and red).

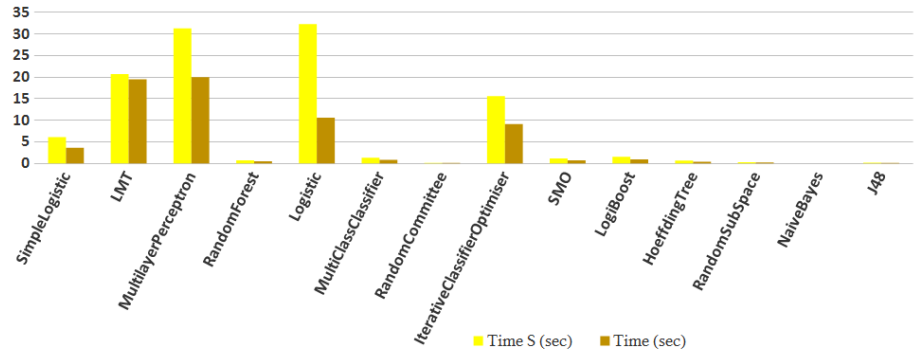

FIGURE 7. Time (seconds) taken for each classifier to construct the model. Time S refers to the 30S dataset, as previous.

Generally, time to build the model is proportional with the cross-validation time, but easier to measure than the second, so we have used it as a rough

indicator of time consumed for real-time model adjusting (based on the real-time input from the user, that would potentially improve the database, and such, the classifier's performance).

3.3. **Discussion.** The results obtained are a good indicator for the next step in regards with pose detection and gesture recognition. First, we can observe very good real-time values obtained for precision and accuracy of some classifiers, which means they are reliable and adequate for emotions based poses recognition in medical or educational video games based applications, as stated in the paper as the main practical use of this study.

Based on this findings, we emphasize three main research directions: (1) pose detection (that can be used in recognising emotions or other indirect user input and generating a corresponding response, like pausing the game or decreasing difficulty) to be extended to gesture recognition, (2) gesture prediction (using preliminary data and incipient detected gestures, it would be possible to determine a possible expected gesture before it is performed or completed, which enables intervention for preventing or changing undesired reactions of the user, for example preventing or reducing violent behaviour) and (3) gesture generator (a large database of user gestures correctly identified can be used in generating these movements for creating human behavioured avatars).

Future improvements could also consist in adjusting the data in order to obtain the best performance, as some classifiers perform worse or better with a larger number of classes, while others are not influenced by this. This means also to consider which classifiers would be best in dealing with: similar poses (commonly confused), large number of poses and time, while keeping accuracy and precision at an acceptable high level (which also need to be determined). It is also worth considering an approach in which poses are based only on the upper body part, ignoring the lower body, thus having less data to check on and decreasing the computation time. This would avoid the confusion between poses where only the upper body is relevant, but can be performed either sitting or standing, and the necessity of separating these poses as sitting or standing.

Further work on this study will also imply extending the database with more entries of poses coming from a larger range of people with different body proportions, in order to assure the scalability, and also to create a database with gestures (sequences of poses, time dependent sequential data). Also, using more pose data on similar gestures and a sensor with higher sensitivity for gathering 3D data (like Kinect 2) will be considered, as we expect these measures to greatly increase the accuracy and precision of pose recognition for most of the classifiers.

4. Conclusion and future work

In this paper we have showed the potential of using natural interaction sensor based gesture recognition, by creating a database from collecting Kinect generated body poses and training and testing several classifiers. We have obtained very good precision and accuracy (up to 0.98) for a set of 30 poses, some of the classifier presenting a potential for usage in real-time systems as well. Moreover, we have analysed and compared the results and obtained from the 38 classifiers and their behaviour in database related changes. As such, there are several potential research directions that we can extend, from recognising body postures based emotions and adapting the system the user is interacting is accordingly for a better experience, up to predicting possible movement of the user and generating emotion related poses in avatars.

Acknowledgements

This work was partially supported by a grant of the Romanian Ministry of Education and Scientific Research, MECS – UEFISCDI, PN II – PT – PCCA – 2013 – 4 – 1797.

References

[1] ***, Tracking Users with Kinect Skeletal Tracking, Kinect for Windows SDK 1.8 Documentation, 2015, Microsoft, https://msdn.microsoft.com.

[2] Sander Bakkes, Chek Tien Tan and Yusuf Pisan, *Personalised Gaming: A Motivation and Overview of Literature*, ACM Proceedings of The 8th Australasian Conference on Interactive Entertainment: Playing the System, 2012.

[3] Sriparna Saha, Shreyasi Datta, Amit Konar and Ramadoss Janarthanan, *A Study on Emotion Recognition from Body Gestures Using Kinect Sensor*, International Conference on Communication and Signal Processing, India, April 3-5, 2014, pp. 56–60.

[4] Orasa Patsadu, Chakarida Nukoolkit and Bunthit Watanapa, *Human Gesture Recognition Using Kinect Camera*, 2012 Ninth International Joint Conference on Computer Science and Software Engineering (JCSSE), pp. 28–32.

[5] Baoliang Wang, Zeyu Chen and Jing Chen, *Gesture Recognition by Using Kinect Skeleton Tracking System*, 2013 Fifth International Conference on Intelligent Human-Machine Systems and Cybernetics, pp. 418–422.

[6] Jiang Wang, Zicheng Liu, Ying Wu and Junsong Yuan, *Mining Actionlet Ensemble for Action Recognition with Depth Cameras*, 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1290–1297.

[7] Tony C. Smith and Eibe Frank, *Statistical Genomics: Methods and Protocols, chapter Introducing Machine Learning Concepts with WEKA*, Springer, New York, 2016, pp. 353–378.

Babeş-Bolyai University, Faculty of Mathematics and Computer Science, Computer Science Department, 1 Mihail Kogălniceanu Street, 400084 Cluj-Napoca, Romania

E-mail address: alinacalin@cs.ubbcluj.ro, alinacalin@mirarehab.com