A comparison of fuzzy approaches for training a humanoid robotic football player

ACAMPORA, Giovanni, DI NUOVO, Alessandro <http://orcid.org/0000-0001-5308-7961>, SICILIANO, Bruno and VITIELLO, Autilia

A Comparison of Fuzzy Approaches for Training a Humanoid Robotic Football Player

Giovanni Acampora*, Alessandro Di Nuovo†, Bruno Siciliano‡ and Autilia Vitiello§
*Department of Physics "Ettore Pancini", University of Naples Federico II, Italy, Email: giovanni.acampora@unina.it
†Sheffield Robotics, Department of Computing, Sheffield Hallam University, United Kingdom, Email: a.dinuovo@shu.ac.uk
‡Department of Electrical Engineering and Information Technologies, University of Naples Federico II, Italy
Email:bruno.siciliano@unina.it
§ Department of Computer Science, University of Salerno, Italy, Email: avitiello@unisa.it

***Abstract*—Fuzzy Systems are an efficient instrument to create efficient and transparent models of the behavior of complex dynamic systems such as autonomous humanoid robots. The human interpretability of these models is particularly significant when it is applied to the cognitive robotics research, in which the models are designed to study the behaviors and produce a better understanding of the underlying processes of the cognitive development. From this research point of view, this paper presents a comparative study on training fuzzy based system to control the autonomous navigation and task execution of a humanoid robot controlled in a soccer scenario. Examples of sensor data are collected via a computer simulation, then we compare the performance of several fuzzy algorithms able to learn and optimize the humanoid robot's actions from the data.**

I. Introduction

Cognitive robotics is a recent branch of robotics that aims to provide robots with human-like intelligent behavior. This fast growing research area is based on the embodied cognition principle, which affirms that a human being learns many cognitive skills by interacting with its environment and other humans using its limbs and senses, and consequently the body largely influences its internal model of the world [1].

Cognitive robotics assumes that an artificial cognition model can be embodied in a physical body (a robot), which will enable the agent to interact with the surrounding environment, something it is expected could lead to the emergence of an intelligent behavior [2], [3]. A humanoid robot is, thus, designed to test this hypothesis by allowing cognitive learning scenarios to be acted out by an accurate reproduction of the perceptual system and the movements of a small child so that it can interact with the world in the same way that a child does. The use of humanoid platforms can help scientists studying cognitive development and working in disciplines such as developmental psychology or epigenetic robotics, increase their understanding of cognitive systems [4].

Cognitive robotics models are often learned from example data collected in experiments with human being or based on artificial neural networks, e.g. [5], [6], [7], which are best suited to simulate the human brain but these have the drawback that are difficult to interpret and, thus, it is hard to identify the relationships between input and outputs. On the contrary, fuzzy logic systems proved to be good controllers for robots, e.g. [8], [9], but they rarely used in cognitive robotics despite they are well known to have the interesting characteristic of being human interpretable. For this reason, they can, therefore, provide useful information for a deeper understanding of the cognitive processes. The interpretability can allow further research on the artificial robotic model to make alterations and make experiments that will be difficult in the real world, for instance, to simulate the effects of brain damages and use the damaged model to test rehabilitation procedures [10].

From this research hypothesis, this paper presents a comparative study on applying fuzzy algorithms to learn the cognitive architecture of a robotic football player. Examples of robot's sensor data and desired actions are collected via a computer simulation in a RobotCup soccer scenario, and then a Fuzzy Rule-based System (FRBS) is learned and optimized from the data examples. Some well-known fuzzy-based algorithms have been explored and their performance compared in order to identify the one that can provide the best performance in this innovative scenario. Precisely, the list of the compared fuzzy-based algorithms includes Fuzzy K-Nearest Neighbor (FKNN) classifier [11], three Adaptive Neuro-Fuzzy Inference System (ANFIS) [12] based approaches and an Adaptive Neuro-Fuzzy Classifier (ANFC-LH) proposed in [13]. The analysis of the performance has been carried out by using well-known evaluation metrics in data mining. Among the compared algorithms, FuzzyKnn has emerged as the best fuzzy approach to control the task execution of a robotic football player.

II. Materials

A. Nao robotic platform and Webots simulator

The robotic platform used in our experiments is a NAO robot version 4 [14] that is 58 cm tall humanoid robot It is equipped with many sensors: Tactile Sensors, Ultrasonic sensors (sonars), A Gyro, An Accelerometer, Force Sensors, Infrared sensors, 2 Cameras, 4 Microphones and high accuracy digital encoders on each joint. Fig. 1 presents the NAO robot parts.

In experiments with robotic platforms, there are many situations in which realistic computer simulation is preferred to the use of real platforms. Among the advantages of simulation, we recall that computer simulation allows researchers to set up an

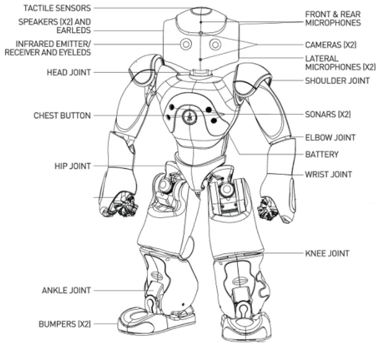


Fig. 1: The NAO Humanoid robot

experimental scenario with several robotic agents and without facing the problem of building in advance, and maintaining, a complex environment such us a soccer field. Thus, simulation can be used as a tool for testing algorithms in order to quickly check for any major problems prior to using the physical robot, and it drastically reduces the time of the experiments such as in evolutionary robotics and reinforcement learning [15], [16]. For these reasons researchers often use computer simulation as the physical body for their cognitive models, e.g. [17]. Indeed, in this work, we used Webots to simulate the NAO robot and the environment as shown in Fig. 2.

Webots is a realistic mobile robot simulator allowing a straightforward transfer to real robots [18]. Webots provides a rapid prototyping environment for modelling, programming and simulating mobile robots. Webots relies on ODE (Open Dynamics Engine) to perform accurate dynamic physics simulation. With Webots it is possible to define and modify a complete mobile robotics setup, and even several different robots sharing the same environment. A number of researchers have used Webots to simulate the Aldebaran Nao humanoid platform for a variety of tasks, mainly in the area of applications related to the RoboCup Standard Platform League, for which Nao is the robot currently employed [19]. Robot soccer field inspired from the RoboCup 2014 Standard Platform League [20]. The soccer field is built on a total carpet area of length 10.4 m and width 7.4 m. The field dimensions (within the white lines) are 9 x 6 m.

B. Task and Input/Output data

As can be seen in Fig. 2, the experimental task we designed for the robot is to avoid some other robotic players, to reach the ball and to kick it towards the goal.

For executing the task, the controller can use four input data: Sonar Left (SL), Sonar Right (SR), Camera Top (CT), Camera Bottom (CB). Sonar is used to detect opponent robots and it will give distance values of obstacles, values are between 2.55 (no obstacles) and 0.0 (collision). Images coming from robot's cameras are processed to calculate the position of the ball in the vision fields (top and bottom, see Fig. 1). CT and CB can be -1.00 (not in the vision field) or a value between 0 (left) and 1 (right). These inputs will be the features involved in the training process of the robotic player.

The output data is: Side step left (TL), Move Forward (MF), Kick the ball (Kick); Side step right (TR). Each one of these outputs can be 0 or 1. Six actions are associated with these outputs: *move forward* (when MF is set to 1 and the remaining outputs to 0); *move toward left* (when TL is set to 1 and the remaining outputs to 0); *move toward right* (when TR is set to 1 and the remaining outputs to 0); *kick the ball* (when Kick is set to 1 and the remaining outputs to 0); *move diagonally toward left* (when TL and MF are set to 1 and the remaining outputs to 0); *move diagonally toward right* (when TR and MF are set to 1 and the remaining outputs to 0). These actions will be the class labels involved in the training process of the robotic player. Therefore, our problem is a multi-classification one with six classes.

The input/output data sample sequences were collected using a pre-programmed algorithm, which was acting as a teacher guiding the robot walk to keep the ball in the middle of one vision field and move in the opposite direction if one of sonar detected an obstacle, i.e. an opponent robot. To collect the data sequences, the robot was placed in 33 random starting positions in the field, 11 left, 11 center and 11 right, and then performed a sequence of movements as described in Figure 2. To increase variability and generalization, opponent robots were placed differently in the field for the three starting position categories (left, center, right). The resulting 33 sequences of input/output have a different number of items.

III. Methods

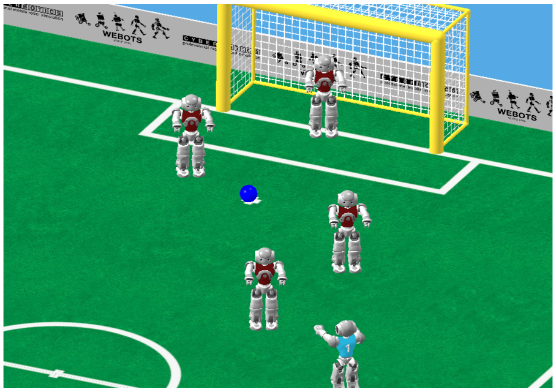
This section is devoted to discuss the fuzzy approaches explored to control the robotic football player.

A. Fuzzy K-Nearest Neighbor (FKNN)

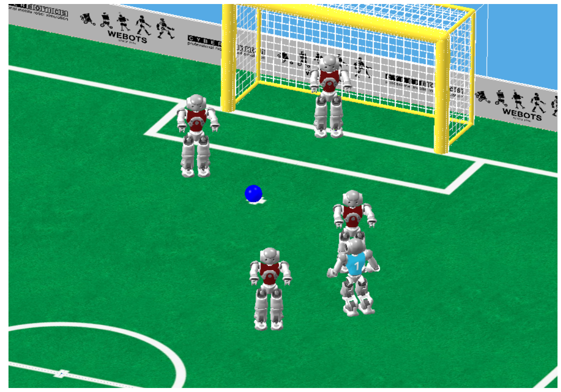
FKNN represents a fuzzy version of the K-Nearest Neighbor (KNN). KNN is a lazy learner that does not require to process labeled example set during a training phase, but it directly uses them during the classification phase to associate a new example with a label searched among its K nearest neighbors. FKNN is similar to KNN in the sense that it must search a label for the new example among its K nearest neighbors, but it considerably differs in the procedure adopted to obtain the K nearest neighbors. In particular, FKNN assigns fuzzy memberships to labeled examples in order to determine the grade of membership of a new example to each considered class. In this way, FKNN provides a level of assurance to accompany the resultant classification. For instance, if a new example is assigned 0.9 membership in one class and 0.05 membership in two other classes, it is possible to be reasonably sure the class of 0.9 membership is the class to which the new example belongs. The setting for FKNN is simple as well as for KNN. Indeed, the only parameter to be set is the number of nearest neighbors (K) be considered during classification task.

B. Adaptive Neuro-Fuzzy Inference System (ANFIS)

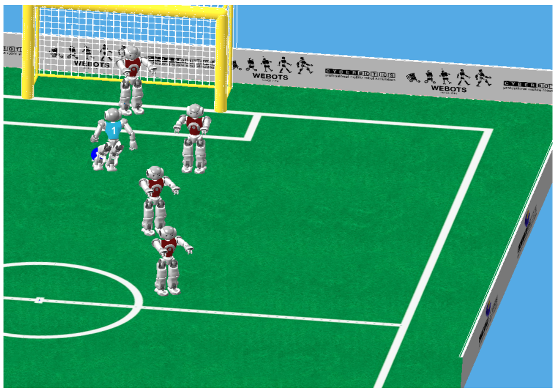
ANFIS is an adaptive network and, as such, it works to achieve the desired input-output by updating parameter sets



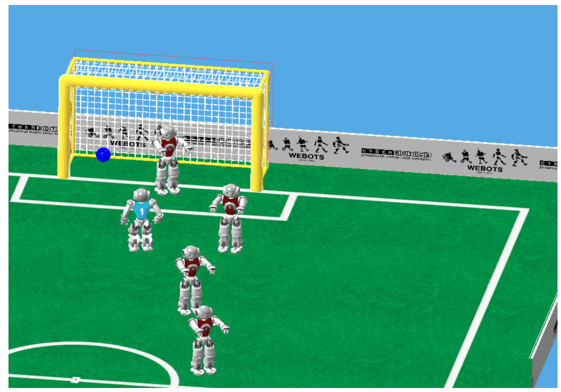
(a) Ball detected, starting



(b) Opponent Dribbling



(c) Kicking



(d) Goal!

Fig. 2: Example of actions the robot does to reach and kick the ball in the simulated RobotCup environment: (a) The robot uses vision to detect the ball; (b) The robot use sonars to dribble opponent robots and vision to stay on target; (c) Ball is on sight, Robot starts the kick action; (d) Robot scores a goal!

according to given training data and a gradient-based update procedure [21]. In particular, this updating feature is used in ANFIS to learn and adapt the parameters of a given Takagi-Sugeno-Kang (TSK) system during a set of epochs. The initial TSK can be set through different procedures. If a grid partition is used, it is necessary to set the number of membership functions associated with each input ($numMfs$) and the membership function type associated with inputs ($inMfType$) and with output ($outMfType$). If the initial TSK is built by using subtractive clustering, it is necessary to set mainly only the cluster center's range of influence for data (rad). Grid partition and subtractive clustering are the most popular approaches for generating the initial TSK. However, recently, also Fuzzy C-Means (FCM) clustering has been used for this aim. In this case, it is necessary only to specify the number of clusters ($numClusters$) to be generated by FCM.

C. Adaptive Neuro-Fuzzy Classifier (ANFC-LH)

ANFC-LH is a classifier network that uses adaptive Linguistic Hedges (LHs). In particular, this approach adds a new layer to a common neuro-fuzzy classifier such as ANFIS to indicate the effect of LHs. The LHs that are constituted by the power of fuzzy sets introduce the importance of the fuzzy sets for fuzzy rules [13]. To tune LHs values, ANFC-LH uses the Scaled Conjugate Gradient (SCG) training algorithm, whereas, it uses the K-means clustering method to construct the fuzzy rules. Hence, ANFC-LH requires setting the number of clusters ($numClusters$)to be used in order to work.

IV. Results and Discussion

This section is devoted to show the results of the comparison among the considered fuzzy approaches applied to train the task execution of the robotic football player. In detail, Section IV-A describes the configurations of the carried out experiments, Section IV-B discusses the used evaluation metrics, and, finally, Section IV-C shows the results of the comparison.

A. Experimental configuration

This work discusses the effects of applying three fuzzy approaches (ANFIS, FKNN, and ANFC-LH) to train a robotic football player in the execution of six actions (see Section II-B). Nevertheless, the comparison precisely involves five fuzzy approaches because three versions of ANFIS depending on the method of generation of the starting TSK are considered. In detail, ANFIS$_{gp}$ takes into account an initial TSK generated by using a grid partition approach; in ANFIS$_{sc}$, the initial TSK is generated through the subtractive clustering and, in ANFIS$_{fcm}$, the initial TSK is generated by using FCM clustering. Table I shows the parameters used for running each one of the considered fuzzy approaches. These parameters represent the best settings experimentally.

TABLE I: Settings of the compared fuzzy approaches

Approach Name	Setting
FKNN	$K = 3$
ANFC-LH	$numClusters = 2$
ANFIS$_{gp}$	$numMfs = 3$, $inMfType = gbell$, $outMfType = constant$
ANFIS$_{sc}$	$rad = 0.2$
ANFIS$_{fcm}$	$numClusters = 7$

All compared fuzzy approaches have been undergone to a training phase involving 500 epochs except for FKNN since it is not characterized by a training phase being a lazy learner. During the experiments, ANFIS$_{fcm}$ and ANFC-LH have shown a non-deterministic behavior. Therefore, in this comparison, their results are related to the most frequent behavior carried out after ten training phases. In addition, it is important to note that all ANFIS approaches provide outputs that are not integers. Since we are facing a classification problem, ANFIS outputs will be rounded to determine the class labels. However, ANFIS approaches can give in output a value that corresponds to no class label. To solve this ANFIS fault, during the comparison, the outputs of ANFIS approaches have been adjusted by setting all negative outputs to the first class label and all outputs greater than the number of classes to the last class label. All compared fuzzy approaches have been implemented in Matlab$^{\text{TM}}$.

The experiments have followed the hold-out approach. In detail, once data have been collected as described in Section II, they have been divided into two sets: one composed of data contained in the first 8 sequences for each of the three starting position categories to be used during the training phase and one composed of data contained in the remaining sequences used for the testing phase. Hence, 24 sequences compose the training data and 9 sequences compose the testing data. This setting was chosen because it reflects the 70-30 ratio that is often used in data mining.

B. Evaluation metrics

The comparison among the considered fuzzy approaches is based on a set of performance metrics typically used in data mining as described in [22]. In detail, the correctness of a classification can be evaluated by computing the number of correctly recognized class examples (true positives or tp), the number of correctly recognized examples that do not belong to the class (true negatives or tn), and examples that either were incorrectly assigned to the class (false positives or fp) or that were not recognized as class examples (false negatives or fn) [22]. These four counts are the components of a confusion matrix for the case of the binary classification. Starting from the confusion matrix, it is possible to compute the following set of metrics: *Precision*, *Recall* and *Fscore*. However, our problem is a multi-class problem. Therefore, as described in [22], in this scenario, for each metric, it is necessary to combine its values obtained by each class in an overall value. This combination can be achieved in two ways: considering the average of the same measures calculated for each class (macro-averaging shown with an M index), or considering the sum of counts to obtain cumulative tp, fn, tn, fp and then calculating a performance measure (micro-averaging shown with m indices) [22]. In an unbalanced data scenario where there are classes with more instances than others, micro-averaged metrics are biased toward the most populated ones, whereas, macro-averaged metrics are biased toward the least populated ones. Since our data are unbalanced, it will be relevant to consider both combination strategies.

Another important evaluation metric is AUC (Area Under the Curve), that captures a single point on the Reception Operating Characteristic (ROC) curve. However, there is yet no well-developed multi-class Reception Operating Characteristic analysis [23]. Therefore, we do not include AUC in the list of measures used in our comparison. An alternative to the evaluation performed through the AUC metric is given by the Cohen's kappa [24]. This metric can be applied in multi-classification and it takes random successes into consideration as a standard, in the same way as the AUC measure [24][25]. Therefore, also this metric is considered in the list of measures used in our comparison. This section is concluded with the formal definition of the used evaluation metrics:

$$Precision_m = \frac{\sum_{i=1}^{C} tp_i}{\sum_{i=1}^{C}(tp_i + fp_i)} \tag{1}$$

$$Recall_m = \frac{\sum_{i=1}^{C} tp_i}{\sum_{i=1}^{C}(tp_i + fn_i)} \tag{2}$$

$$Fscore_m = \frac{2 \cdot Precision_m \cdot Recall_m}{Precision_m + Recall_m} \tag{3}$$

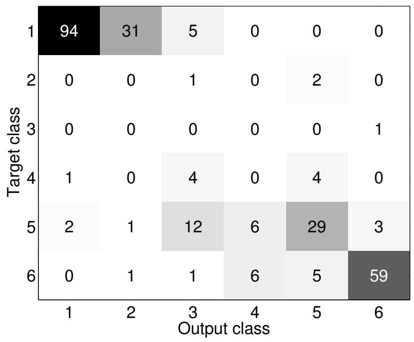
$$Precision_M = \frac{\sum_{i=1}^{C} \frac{tp_i}{(tp_i + fp_i)}}{C} \tag{4}$$

$$Recall_M = \frac{\sum_{i=1}^{C} \frac{tp_i}{(tp_i + fn_i)}}{C} \tag{5}$$

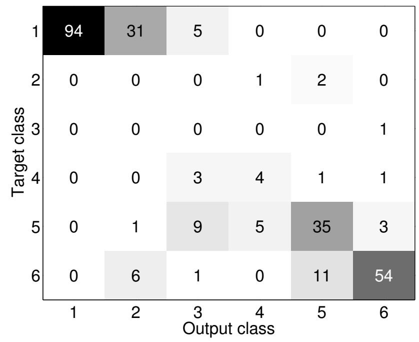
$$Fscore_M = \frac{2 \cdot Precision_M \cdot Recall_M}{Precision_M + Recall_M} \tag{6}$$

$$Kappa = \frac{N \cdot \sum_{i=1}^{C} X_{ii} - \sum_{i=1}^{C} \sum_{j=1}^{C} Xij \cdot Xji}{N^2 - \sum_{i=1}^{C} \sum_{j=1}^{C} Xij \cdot Xji} \tag{7}$$

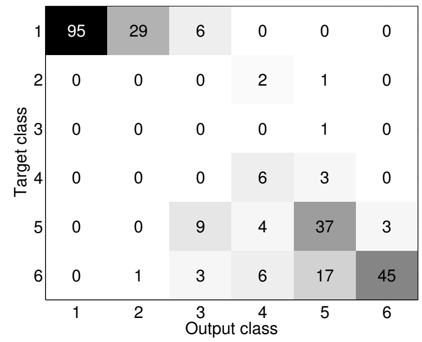
where tp_i are the true positives for the class i, and fp_i are the false positives for the class i, fn_i are the false negatives for the class i, tn_i are the true negatives for the class i, X is the confusion matrix, C is the number of classes and N is the total number of examples.



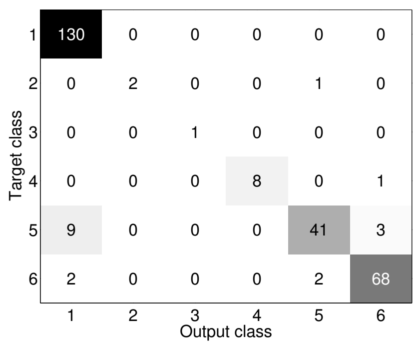
(a) ANFIS$_{gp}$



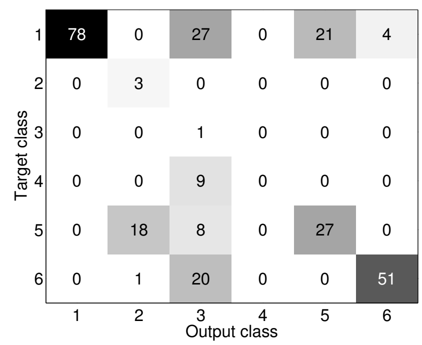
(b) ANFIS$_{sc}$



(c) ANFIS$_{fcm}$



(d) FKNN



(e) ANFC-LH

Fig. 3: Confusion matrices for all compared fuzzy approaches

TABLE II: Results for all compared fuzzy approaches. P stands for precision, R for recall and F for Fscore.

Metric	FKNN	ANFC-LH	ANFIS$_{gp}$	ANFIS$_{sc}$	ANFIS$_{fcm}$
P_m	0.9328	0.5970	0.6791	0.6978	0.6828
R_m	0.9328	0.5970	0.6791	0.6978	0.6828
F_m	0.9328	0.5970	0.6791	0.6978	0.6828
P_M	0.9664	0.4403	0.4384	0.5049	0.4830
R_M	0.8789	0.6363	0.3483	0.4296	0.4534
F_M	0.9206	0.5204	0.3882	0.4643	0.4677
$Kappa$	0.8949	0.4742	0.5595	0.5868	0.5671

C. Performance Results

Fig. 3 shows the confusion matrices for all compared approaches. Table IV-C show the performance of all compared fuzzy approaches in terms of the used evaluation metrics.

By analyzing Table IV-C, it is possible to state that FKNN is the best fuzzy approach to control the robotic football player. Indeed, it is the best performer in all considered metrics. From its confusion matrix shown in Fig. 3d, it is possible to state that FKNN performs well also in the identification of classes of a small number of examples such as class 2, 3 and 4. In particular, out of 8 examples for class 4, 100% are correctly classified by FKNN. This is a very important feature because class 4 represents the action of ”kick the ball”, and, a poor classification quality for this class negatively affects the robotic football player goals.

By analyzing the performance of the other fuzzy approaches, it is possible to state that ANFIS approaches have similar performance in terms of the micro-averaged evaluation, but not in terms of macro-averaged where only ANFIS$_{sc}$ and ANFIS$_{fcm}$ are characterized by similar values. This implies that ANFIS$_{gp}$ is less able than ANFIS$_{sc}$ and ANFIS$_{fcm}$ to classify classes with a small number of examples such as class 2, 3 and 4. Indeed, even if all ANFIS approaches classify correctly 0% of the classes 2 and 3, they are characterized by a different number of examples correctly classified of class 4. In detail, as shown in the relative confusion matrices are shown in Fig. 3, out of 8 examples for class 4, 0% are correctly classified by ANFIS$_{gp}$, 50% are correctly classified by ANFIS$_{sc}$ and 75% are correctly classified by ANFIS$_{fcm}$. Hence, it is possible to state that ANFIS$_{sc}$, but above all, ANFIS$_{fcm}$ are the best performers among ANFIS approaches to allow the robot to kick the ball since they have a higher classification quality for the class 4 representing the action of ”kick the ball”.

As for ANFC-LH, it performs less well than ANFIS approaches in terms of the micro-averaged evaluation, but it performs better than ANFIS approaches in the macro-averaged one. This implies that ANFC-LH is more able to identify classes with a small number of examples and less able to identify classes with a large number of examples with respect to ANFIS approaches. Indeed, for example, out of 126 examples of the class 1 (a class with a large number of examples), the percentage of examples correctly classified is 75% for ANFIS approaches and 62% for ANFC-LH, whereas, out of 3 examples of class 2 (a class with a small number of examples), the percentage of examples correctly classified is 0% for ANFIS approaches and 100% for ANFC-LH. However, in spite of ANFC-LH is able to identify classes with a small number of examples better than ANFIS approaches, it succeeds to classify correctly no examples of class 4 representing the action of "kick the ball". Hence, although ANFC-LH is characterized by a higher classification quality in the micro-averaged evaluation (i.e. the evaluation more significant in our context where unbalanced data are collected) with respect to ANFIS approaches, it is the worst performer in our task because it fails to achieve its principal objective: *kick the ball*.

In summary, in our comparative analysis about micro and macro-averaged evaluations, FKNN has emerged as the best approach for leading the robotic football player forward the final goal represented by kicking the ball thanks to its high classification quality both for classes related to all robot movements (classes 1, 2, 3, 5, 6) and the class related to the action "kick the ball" (class 4). ANFIS approaches and, in particular, ANFIS$_{fcm}$, are the second performers as they achieved a higher classification quality than ANFC-LH for the class related to the objective of the robot: *kick the ball* (class 4). Indeed, ANFC-LH has emerged as the worst performer by failing to lead the robot toward its main objective.

The ranking FKNN, ANFIS approaches and ANFC-LH is confirmed also by an analysis performed through $Kappa$ metric where all classes are treated equally. Indeed, by analyzing $kappa$ values in the Table IV-C, it is possible to state that FKNN classifies correctly the highest number of examples regardless of their class labels, ANFIS approaches classify correctly the second higher number of examples and ANFC-LH is the last one.

V. Conclusion

This paper presents the comparison of some well-known fuzzy approaches to control a robotic football player. After the performance analysis based on well-defined metrics in data mining domain, FKNN has emerged as the best performer with respect to ANFIS approaches and ANFC-LH method. In future, we will apply the same methodology to more complex play scenarios, in which also the ball and opponent robots can move. Moreover, we will investigate the use of the Fuzzy Cognitive Maps as these can be better suited for Cognitive Robotics applications.

References

[1] L. Shapiro, *Embodied cognition*. Routledge, 2010.

[2] H. Levesque and G. Lakemeyer, "Cognitive robotics," *Foundations of artificial intelligence*, vol. 3, pp. 869–886, 2008.

[3] A. Di Nuovo, D. Marocco, A. Cangelosi, V. De La Cruz, and S. Di Nuovo, "Mental practice and verbal instructions execution: a cognitive robotics study," in *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2012, pp. 1–6.

[4] A. Di Nuovo, M. Vivian, and A. Cangelosi, "Grounding fingers, words and numbers in a cognitive developmental robot," in *IEEE Symposium on Computational Intelligence, Cognitive Algorithms, Mind, and Brain (CCMB)*. IEEE, 2014, pp. 9–15.

[5] D. Vernon, G. Metta, and G. Sandini, "A survey of artificial cognitive systems: Implications for the autonomous development of mental capabilities in computational agents," *IEEE transactions on evolutionary computation*, vol. 11, no. 2, pp. 151–180, 2007.

[6] A. Di Nuovo, V. M. De La Cruz, and D. Marocco, "Special issue on artificial mental imagery in cognitive systems and robotics," *Adaptive Behaviour*, vol. 21, no. 4, pp. 217–221, 2013.

[7] Y. Wang, "Cognitive learning methodologies for brain-inspired cognitive robotics," *International Journal of Cognitive Informatics and Natural Intelligence (IJCINI)*, vol. 9, no. 2, pp. 1–18, 2016.

[8] H. A. Hagras, "A hierarchical type-2 fuzzy logic control architecture for autonomous mobile robots," *IEEE Transactions on Fuzzy systems*, vol. 12, no. 4, pp. 524–539, 2004.

[9] O. Castillo and P. Melin, "A review on interval type-2 fuzzy logic applications in intelligent control," *Information Sciences*, vol. 279, pp. 615–631, 2014.

[10] D. Conti, S. Di Nuovo, A. Cangelosi, and A. Di Nuovo, "Lateral specialization in unilateral spatial neglect: a cognitive robotics model," *Cognitive processing*, vol. 17, no. 3, pp. 321–328, 2016.

[11] J. M. Keller, M. R. Gray, and J. A. Givens, "A fuzzy k-nearest neighbor algorithm," *IEEE transactions on systems, man, and cybernetics*, no. 4, pp. 580–585, 1985.

[12] C.-T. Sun and J.-S. Jang, "A neuro-fuzzy classifier and its applications," in *Fuzzy Systems, 1993., Second IEEE International Conference on*. IEEE, 1993, pp. 94–98.

[13] B. Cetisli, "Development of an adaptive neuro-fuzzy classifier using linguistic hedges: Part 1," *Expert Systems with Applications*, vol. 37, no. 8, pp. 6093–6101, 2010.

[14] D. Gouaillier, V. Hugel, P. Blazevic, C. Kilner, J. Monceaux, P. Lafourcade, B. Marnier, J. Serre, and B. Maisonnier, "Mechatronic design of nao humanoid," in *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*. IEEE, 2009, pp. 769–774.

[15] A. Fachantidis, A. Di Nuovo, A. Cangelosi, and I. Vlahavas, "Model-based reinforcement learning for humanoids: A study on forming rewards with the icub platform," in *IEEE Symposium on Computational Intelligence, Cognitive Algorithms, Mind, and Brain (CCMB)*. IEEE, 2013, pp. 87–93.

[16] A. Di Nuovo, V. De La Cruz, A. Cangelosi, and S. Di Nuovo, "The icub learns numbers: An embodied cognition study," in *Proceedings of International Joint Conference on the Neural Networks (IJCNN)*. IEEE, 2014, pp. 692–699.

[17] V. Tikhanoff, A. Cangelosi, and G. Metta, "Integration of speech and action in humanoid robots: icub simulation experiments," *IEEE Transactions on Autonomous Mental Development*, vol. 3, no. 1, pp. 17–29, 2011.

[18] O. Michel, "Cyberbotics ltd. webots: professional mobile robot simulation," *International Journal of Advanced Robotic Systems*, vol. 1, no. 1, p. 5, 2004.

[19] J. I. Zannatha, L. F. Medina, R. C. Limón, and P. M. Álvarez, "Behavior control for a humanoid soccer player using webots," in *Electrical Communications and Computers (CONIELECOMP), 2011 21st International Conference on*. IEEE, 2011, pp. 164–170.

[20] R. T. Committee, "RoboCup Standard Platform League (NAO) Rule Book," http://www.tzi.de/spl/pub/Website/Downloads/Rules2014.pdf, 2014.

[21] G. Acampora and A. Vitiello, "Interoperable neuro-fuzzy services for emotion-aware ambient intelligence," *Neurocomputing*, vol. 122, pp. 3–12, 2013, advances in cognitive and ubiquitous computingSelected papers from the Sixth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS-2012).

[22] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Information Processing & Management*, vol. 45, no. 4, pp. 427–437, 2009.

[23] N. Lachiche and P. Flach, "Improving accuracy and cost of two-class and multi-class probabilistic classifiers using roc curves," in *ICML*, 2003, pp. 416–423.

[24] S. Garcia, J. Derrac, J. Cano, and F. Herrera, "Prototype selection for nearest neighbor classification: Taxonomy and empirical study," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 3, pp. 417–435, 2012.

[25] A. Ben-David, "A lot of randomness is hiding in accuracy," *Engineering Applications of Artificial Intelligence*, vol. 20, no. 7, pp. 875–885, 2007.