Real-Time Multimodal Signal Processing for HRI in RoboCup: Understanding a Human Referee

Filippo Ansalone[1,*,†], Flavio Maiorana[1,*,†], Daniele Affinita[1,*], Flavio Volpi[1,*], Eugenio Bugli[1,*], Francesco Petri[1,2,*], Michele Brienza[1,*], Valerio Spagnoli[1,*], Vincenzo Suriani[3,*], Daniele Nardi[1] and Domenico D. Bloisi[4]

[1]*Sapienza University of Rome*
[2]*Institute for Cognitive Sciences and Technologies, National Research Council, Italy*
[3]*University of Basilicata*
[4]*University of International Studies of Rome – UNINT*

Abstract

Advancing human-robot communication is crucial for autonomous systems operating in dynamic environments, where accurate real-time interpretation of human signals is essential. RoboCup provides a compelling scenario for testing these capabilities, requiring robots to understand referee gestures and whistle with minimal network reliance. Using the NAO robot platform, this study implements a two-stage pipeline for gesture recognition through keypoint extraction and classification, alongside continuous convolutional neural networks (CCNNs) for efficient whistle detection. The proposed approach enhances real-time human-robot interaction in a competitive setting like RoboCup, offering some tools to advance the development of autonomous systems capable of cooperating with humans.

Keywords

Human-Robot Interaction, Audio Communication, Gesture Communication, Soccer Robots

1. Introduction

Human-robot communication has evolved significantly, but it becomes challenging in competitive environments such as RoboCup, where robots must interpret human signals with high accuracy. In these settings, the challenge is to reduce the reliance on network-based communications in favor of multimodal signal processing. This shift aligns with the growing interest in developing robots capable of understanding human gestures and audio cues, such as referee signals during matches. The challenge lies in the robots' ability to process and interpret these multimodal signals in real-time, despite the constraints of limited computational resources. In the context of RoboCup, where human referees convey critical game states and events through gestures and whistles, the need for precise and efficient recognition systems becomes evident. This paper explores the integration of multimodal perception of gestures and whistles using the NAO robot platform, focusing on achieving robust performance under real-time conditions while being compliant with the official competition rules.

We employ a two-stage pipeline approach for gesture recognition, combining keypoint extraction and classification to interpret referee poses accurately. Simultaneously, we utilize continuous convolutional kernel neural networks (CKCNNs) [1] for whistle detection, balancing accuracy with computational

11th Italian Workshop on Artificial Intelligence and Robotics (AIRO 2024)
*Corresponding author.
†These authors contributed equally.
ansalone.1950936@studenti.uniroma1.it (F. Ansalone); maiorana.2051396@studenti.uniroma1.it (F. Maiorana); affinita.1885790@studenti.uniroma1.it (D. Affinita); volpi.1884040@studenti.uniroma1.it (F. Volpi); bugli.1934824@studenti.uniroma1.it (E. Bugli); francesco.petri@uniroma1.it (F. Petri); brienza@diag.uniroma1.it (M. Brienza); spagnoli.1887715@studenti.uniroma1.it (V. Spagnoli); vincenzo.suriani@unibas.it (V. Suriani); nardi@diag.uniroma1.it (D. Nardi); domenico.bloisi@unint.eu (D. D. Bloisi)
0009-0002-0492-4748 (F. Ansalone); 0009-0003-2059-7254 (F. Maiorana); 0009-0000-9347-9847 (D. Affinita); 0009-0004-9822-5124 (F. Volpi); 0009-0000-9540-681X (E. Bugli); 0009-0008-6208-1498 (F. Petri); 0009-0000-1549-9500 (M. Brienza); 0009-0008-0284-9602 (V. Spagnoli); 0000-0003-1199-8358 (V. Suriani); 0000-0001-6606-200X (D. Nardi); 0000-0003-0339-8651 (D. D. Bloisi)

Figure 1: Overview of the Robocup SPL field during the standby phase (left) and referee gesture detection from the robot's perspective (right). The right image highlights the region of interest (ROI) and displays the skeleton.

efficiency. The proposed methods demonstrate the potential for enhancing human-robot interaction in competitive environments, contributing to the ongoing development of robot synergy with humans.

2. Related Work

Interpreting human behavior has long been a central challenge in robotics. Humans communicate through various modalities, including vision, audio, and motion. This multimodal nature provides rich information that sensory input can capture and analyze.

Recent advances in Deep Learning have facilitated the integration of multimodal data, significantly improving the comprehension of relationships within individual modalities, a key factor for precise message interpretation [2] [3].

In the context of RoboCup, human-robot interaction is predominantly one-way, with human referees conveying game states and events to robots. A significant trend in the RoboCup SPL league is the progressive reduction of network communication in favor of human-like signal interpretation, allowing robots to interpret human signals more naturally.

In human soccer matches, gestures serve as a critical means of communication, especially in noisy environments such as stadiums. Previous works have extensively explored gesture recognition among agents using deep learning models, as seen in [4]. A common approach is a two-stage pipeline, in which the person's skeleton is first extracted [5] [6], followed by the classification of keypoint evolution over time. Specifically, Di Giambattista et al. [7] employed OpenPose with Part Affinity Fields to extract the skeleton, using a subsequent network to analyze the relative positioning of keypoints for final pose prediction. Alternatively, single-stage pipelines [8] [9] offer end-to-end models, but require consideration of both spatial and temporal data from image sequences, often resulting in significantly larger models. Given that the NAO robot is an edge device with limited computational resources, we opted for a two-stage pipeline to maintain efficiency while ensuring accurate pose recognition.

Audio processing to detect specific sounds is an active research field, finding applications in various domains such as environmental monitoring [10], security [11], and sports analytics [12]. In particular, whistle detection has received attention in the context of sports, where referees' whistles are used to signal important events during matches. Unlike gestures, the whistling signal itself does not convey a specific meaning directly. Instead, it must be interpreted in the context of the current situation and game state, requiring a grounding [13] mechanism to relate the sound to relevant game events. A potential approach for whistle detection is to use LSTMs [14], which offer the advantage of modeling long-range temporal dependencies and providing a larger context for analysis. However, they tend to be computationally expensive and slower due to their recurrent structure [15]. Alternatively, computing the Fourier transform of the audio signal and using CNNs to process the resulting spectrogram is a more efficient solution [16]. Given that whistle recognition does not require modeling extensive temporal context, CNNs provide a better balance between accuracy and computational efficiency for our task.

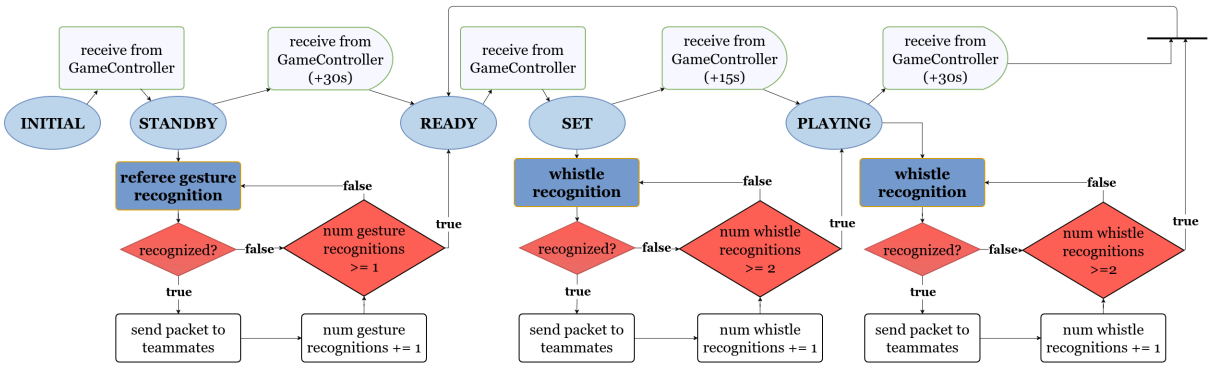

Figure 2: Triggered states during a game, for each robot in the field. It is important to highlight the integration of the modules that process referee's signals within the pipeline: a certain number of robots have to recognize a specific referee's signal (gesture for 4 consecutive camera frames or whistle) to move instantaneously to the following state, bypassing the delay associated to the message from Game Controller.

3. Methodology

To handle the detection of signals coming from the referee, a pipeline has been designed, involving many robots of the team. In Fig. 2, the pipeline is shown with the game states and the teammates.

3.1. Whistle Recognition

For whistle recognition, we employed Continuous Kernel Convolutional Neural Networks, which extend classical CNNs by using a kernel **parametrized by a small neural network**. CNNs excel in efficiently learning functions over structured data, like images or audio, by leveraging translation equivariance, albeit with a fixed receptive field size. In contrast, continuous kernel convolutions adapt to varying input lengths and resolutions, offering several advantages in audio processing:

- The same architecture accommodates different preprocessing techniques, such as varying sampling rates, window sizes, or feature extraction methods (e.g., STFT or MFCC).
- The number of parameters of the network is decoupled from its receptive field, allowing to have a long-range kernel with a relatively small number of parameters.

In our application, the basic building block is the CKBlock:

```
input -> BatchNorm -> CKConv -> GELU -> DropOut -> Linear -> GELU -> + -> output
  |_____________________________________________________________________|
```

The CKConv layer is the core of the architecture, since it contains the kernel generation and convolution operation. The convolution operation is defined as $(x * \psi)(t) = \sum_{c=1}^{N_{in}} \sum_{\tau=0}^{t} x_c(\tau) \cdot \psi_c(t-\tau)$, which means that the convolver is now viewed as a vector-valued continuous function $\psi : \mathbb{R} \to \mathbb{R}^{N_{out} \times N_{in}}$, parametrized with a small neural network MLP^{ψ}:

- The input is a relative position $(t - \tau)$ of the convolvee
- The output is the value $\psi(t - \tau)$ of the convolutional kernel at that position

The main consequence of this is that the kernel is arbitrarily large.

The entire network is a sequence of 4 CKBlocks, with a final fully connected part. More specifically, the convolutional layers have a hidden size of 32. The convolutional kernels are structured as simple 3-layer MLPs with hidden size 16. We chose as kernel size 31, since an overly large kernel would overfit the training data, while a too small kernel would need a deeper network. Overall, we reached a network size of 59.1k trainable parameters.

3.1.1. Data gathering

Structure and preprocessing In addition to the task of classifying an audio sample as either whistle or no-whistle, a critical challenge in RoboCup games is ensuring accurate predictions in the presence of background noise, such as crowd sounds, robot movements, and other environmental sounds. Therefore, the dataset [16] is a collection of audio files collected both in lab conditions and during the actual matches, using the robots' microphones. Since, on average there are few whistles in a match, the result is a heavily unbalanced dataset, with a ratio of $10:1$ (60000 no-whistle samples, 6000 whistle samples). The dataset was manually cleaned, removing many samples where the only noise source was the robot walking, or where there was silence. Also, the labelling happened manually through the software Audacity by extracting the audio events, defined as start and end of the whistle, in text files. These were then associated with the corresponding audio samples using the library *Librosa* [17].

Feature extraction To extract the features, we perform a frequency analysis of the audio signal using short-time fourier transforms. The result is, for each audio, a series of vectors of shape `(1,NUMBER_FREQUENCIES)`, where each vector represents the frequency amplitudes of a window. We extracted 1024 frames per window at 44100 Hz. This resulted in every data sample being a vector of shape `(1,513)`.

3.2. Gesture Recognition

For the recognition of a referee pose, we propose a 2 step architecture based on a pretrained key point extractor and then a classification module.

Since one of the goals in RoboCup is to optimize as much as possible each algorithm to grant a fast real-time execution, we had to rely on MoveNet Lightning[18] which is a deep learning architecture based on MobileNetV2 [19] specifically developed for real-time applications which takes as input a 192x192 RGB image. We adapt the Nao camera frames, featuring a resolution 640x480, by scaling and padding to match the input shape. Due to the distance of the referee from the robots, the image scaling down leads to a detail loss on our region of interest (ROI) and the key point extractor does not recognize the pose correctly. To overcome this issue, we implemented a crop on the ROI containing the referee, and then resized and padded it to the desired input shape. This crop is also useful to prevent the MoveNet to focus on a different person which is standing at the border of the field which may cause false readings making the entire pipeline more robust. Figure 1 illustrates an example of the ROI selection and the pose estimation network in action, estimating the referee's skeleton.

After the key point extraction, we needed to extract a good feature because classifying directly on the raw key point coordinates would be a much harder problem, especially with a small dataset. To address this problem we decided to calculate the angles of the joints that are more useful for our task. So for both the left and right sides of the body, the algorithm computes the angles between:

- Hip - Shoulder - Elbow
- Shoulder - Elbow - Wrist

This procedure eventually computes less features that are, on the other hand, much more representative of our problem. In general, given 3 points (A, B, C) the angle is computed as:

$$\theta = atan2(BC_y, BC_x) - atan2(BA_y, BA_x)$$

This feature is better not only because it is easier to interpret but also because it grants scale and rotation invariance which are very useful considering that both the dataset and the classifier architecture were small. These two properties, together with the intrinsic translation equivariance provided by the CNN architecture, contribute to a generally more robust pipeline.

When a robot sees the pose for at least 4 consecutive frames, the recognition is succesful and a packet is sent to the team so that every robot can enter the ready state, as shown in Fig. 2.

	Accuracy	Precision	Recall
Test	98.02%	79.36%	90.35%
Real (Play)	75%	100%	80%
Real (Ready/Set)	100%	100%	100%

(a) Whistle Recognition Results based on 147440 test samples (frequency windows) and 73 real situations over 8 games

	Accuracy	Precision	Recall	F1-Score
Test	99%	99%	99%	99%
Real	50%	100%	50%	66%

(b) Gesture Recognition Results based on 153 test samples and 18 real situations over 8 games

Table 1
Overall performance evaluation of both networks used to interpret the human referee, reporting metrics from both the dataset and real scenarios.

3.2.1. Data gathering

The released rule that has to be followed states that *"To announce the transition from standby to ready state, the referee will raise both hands over their head"*.

To this end, the dataset was collected by our team in a private environment, allowing for consistent conditions throughout the data acquisition process. This approach facilitated data gathering, which was subsequently manually labeled to ensure a high quality labeling.

4. Results

We evaluated separately the whistle and the gesture subsystems. Table 1 shows the results of the models on the test data and the real scenario. In the whistle test data case, we reached a lower precision, due to the highly imbalanced dataset. Whereas, in a real scenario, the detector worked pretty well. Lowers precision could be a problem in cases of similar sounds to whistles that could cause false detections. This can be easily mitigated by using a consensus approach. In case of the real scenario, the distinction between playing and not playing is made to show the difference between these two cases. When the robots are playing, the whistle always comes after a goal is scored, and usually the crowd cheers in such a situation. Therefore, especially when referees do not whistle loudly, the model is not able to distinguish the whistle sound from the crowd noise. On the other hand, when the robots are not playing, it means they are waiting for a kick-off. In this case, there is usually less noise, and the model is able to detect the whistles with high accuracy. The same pattern occurs in the gesture recognition case, in which high precision was preferred over recall to avoid incurring rule penalties.

Both pipelines are fast enough to run on a NAO robot in about 0.8 ms (whistle) and 200 ms (gesture).

5. Conclusions

This paper presents an approach to detecting audiovisual signals from a human in the context of a robot soccer game in real-time. Using a two-stage pipeline for gestures and a CCNN for whistles, we balanced computational efficiency with accuracy on the NAO robot platform.

Our results showed strong performance in whistle detection, while gesture recognition faced challenges in real-world conditions, particularly in noisy environments. Future work will focus on enhancing noise resilience and improving gesture recognition to better handle dynamic scenarios.

Acknowledgments

This work has been carried out while Francesco Petri and Michele Brienza were enrolled in the Italian National Doctorate on Artificial Intelligence run by Sapienza University of Rome. We also acknowledge partial financial support from PNRR MUR project PE0000013-FAIR.

References

[1] D. W. Romero, A. Kuzina, E. J. Bekkers, J. M. Tomczak, M. Hoogendoorn, Ckconv: Continuous kernel convolution for sequential data, 2022. URL: https://arxiv.org/abs/2102.02611. arXiv:2102.02611.

[2] H. Liu, T. Fang, T. Zhou, Y. Wang, L. Wang, Deep learning-based multimodal control interface for human-robot collaboration, Procedia CIRP 72 (2018) 3–8. URL: https://www.sciencedirect.com/science/article/pii/S2212827118303846. doi:https://doi.org/10.1016/j.procir.2018.03.224, 51st CIRP Conference on Manufacturing Systems.

[3] H. Su, W. Qi, J. Chen, C. Yang, J. Sandoval, M. A. Laribi, Recent advancements in multimodal human–robot interaction, Frontiers in Neurorobotics 17 (2023) 1084000.

[4] P. Neto, M. Simão, N. Mendes, M. Safeea, Gesture-based human-robot interaction for human assistance in manufacturing, The International Journal of Advanced Manufacturing Technology 101 (2019) 119–135.

[5] A. Kendall, M. K. Grimes, R. Cipolla, Posenet: A convolutional network for real-time 6-dof camera relocalization, 2015 IEEE International Conference on Computer Vision (ICCV) (2015) 2938–2946. URL: https://api.semanticscholar.org/CorpusID:12888763.

[6] Y. Xiu, J. Li, H. Wang, Y. Fang, C. Lu, Pose flow: Efficient online pose tracking, arXiv preprint arXiv:1802.00977 (2018).

[7] V. Di Giambattista, M. Fawakherji, V. Suriani, D. D. Bloisi, D. Nardi, On field gesture-based robot-to-robot communication with nao soccer players, in: S. Chalup, T. Niemueller, J. Suthakorn, M.-A. Williams (Eds.), RoboCup 2019: Robot World Cup XXIII, Springer International Publishing, Cham, 2019, pp. 367–375.

[8] F. B. Ashraf, M. U. Islam, M. R. Kabir, J. Uddin, Yonet: A neural network for yoga pose classification, SN Computer Science 4 (2023). doi:10.1007/s42979-022-01618-8.

[9] M. Ur Rehman, F. Ahmed, M. Attique Khan, U. Tariq, F. Abdulaziz Alfouzan, N. M Alzahrani, J. Ahmad, Dynamic hand gesture recognition using 3d-cnn and lstm networks, Computers, Materials & Continua 70 (2021).

[10] E. L. White, P. R. White, J. M. Bull, D. Risch, S. Beck, E. W. Edwards, More than a whistle: Automated detection of marine sound sources with a convolutional neural network, Frontiers in Marine Science 9 (2022) 879145.

[11] M. Neri, F. Battisti, A. Neri, M. Carli, Sound event detection for human safety and security in noisy environments, IEEE Access 10 (2022) 134230–134240.

[12] P.-M. Filippidis, N. Vryzas, R. Kotsakis, I. Thoidis, C. A. Dimoulas, C. Bratsas, Audio event identification in sports media content: The case of basketball, in: Audio Engineering Society Convention 146, Audio Engineering Society, 2019.

[13] M. F. Jung, Affective grounding in human-robot interaction, in: Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction, 2017, pp. 263–273.

[14] J. Li, A. Mohamed, G. Zweig, Y. Gong, Lstm time and frequency recurrence for automatic speech recognition, in: 2015 IEEE workshop on automatic speech recognition and understanding (ASRU), IEEE, 2015, pp. 187–191.

[15] H. Purwins, B. Li, T. Virtanen, J. Schlüter, S.-Y. Chang, T. Sainath, Deep learning for audio signal processing, IEEE Journal of Selected Topics in Signal Processing 13 (2019) 206–219.

[16] D. Kleingarn, D. Brämer, Neural network and prior knowledge ensemble for whistle recognition, in: C. Buche, A. Rossi, M. Simões, U. Visser (Eds.), RoboCup 2023: Robot World Cup XXVI, Springer Nature Switzerland, Cham, 2024, pp. 17–28.

[17] librosa/librosa: 0.10.2.post1, 2024. URL: https://doi.org/10.5281/zenodo.11192913. doi:10.5281/zenodo.11192913.

[18] Next-generation pose detection with movenet and tensorflow.js, 2021. URL: https://blog.tensorflow.org/2021/05/next-generation-pose-detection-with-movenet-and-tensorflowjs.html.

[19] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, L.-C. Chen, Mobilenetv2: Inverted residuals and linear bottlenecks, arXiv preprint arXiv:1801.04381 (2018).