SCIENCES SORBONNE UNIVERSITÉ

MASTER Spi Sciences pour l'ingénieur

Mini Project Social Robotic - Learning from Human Feedback

Solenn Dumont Le Brazidec, Noama Adra

20 décembre 2022

Introduction

The goal of this project is to discover human feedbacks in machine learning and to get better performance with a human-in-the-loop approach using any learning algorithm as a baseline. The paper chosen is "Learning Behaviors with Uncertain Human Feedback" [1]. We evaluated this algorithm with different type of feedbacks.

The report is composed of five different points : a benchmark and explication of several learning algorithms, a description of the article we chose and the first results. Then, the adaptation of the algorithm to other feedbacks and the comparison of the new results.

1 Benchmark of similar existing projects/approaches

In this section we will be briefly presenting five different articles that use variations of human feedback.

Reference	Feedback type	Environment	Algorithm used	Type of learning
1	Binary corrective feedback	Gym environment (Lunar lander , Mountain Car)	PPMP	R.L
2	Unconditional human language(Speech)	Virtual box with several objects	MDP	IRL
3	Human intrinsic reactions(Brain Activity : Electric Potential)	Atari environment	MDP and DQN	Active Learing
4	Human language(Speech)	Fetch block-stacking	GCRL	R.L.
5	Uncertain feedback(Text)	Virtual dog to catch rat	Expectation Maximization and Gradient Descent	R.L.

Table 1 – Benchmark

The results and commentaries of each article will be discussed in the next section.

Results

1. "Deep Reinforcement Learning with Feedback-based Exploration & binary corrective feedback" :

In this article, they introduced a new algorithm : PPMP which stands for Predictive Probabilistic

Merging Policies. A predictor is used to manage corrections and improve memorisation. Its purpose is to give us values of estimates of the corrected policy. The algorithm was then trained on several continuous control problems in open AI gym such as Mountain Car, Lunar Lander. For the evaluation process it was compared to other algorithms : Deep Deterministic Policy Gradient (DDPG) and DCOACH with the same environment. The methods proposed outperforms in every aspect. [2]

2. "Learning Rewards from Linguistic Feedback" :

In this article, the environment composed of a black square where several shapes of different colors were placed . The number of objects increased depending on the level of the game, the agent had to collect the objects and get a reward. The agent receives instructions as a form of positive or negative feedback. He has to interpret the feedback and deduce the teacher's preferences to choose the action. Three different models were tested : a "Literal", "Pragmatic" and "End-to-end Inference Network". All models learned from a live learners. Results show that the "pragmatic" model has the nearest-human performance. The results improved over successful levels. [3]

3. "Deep Reinforcement learning with implicit human feedback" :

In this article, the agent learned from the human intrinsic reactions. An electroencephalogram was recorded, then the signals were decoded to be later interpreted by the agent as a feedback. Two different kinds of loops were developed : one where the human feedback was provided inside the loop and one where it was provided outside the loop. The performances improves by 2.25 times over training time. [4]

4. "Overcoming Referential Ambiguity in language-guided goal-conditioned Reinforcement Learning" :

The main goal of the article is to find the best combination of learner/teacher for a given problem. Here, the experience is about grabbing objects asked by the teacher : There are three different objects called blocks : a red plain one, a blue plain one and a blue striped one. The goal is to put two blocks close to each other. Those three objects create ambiguities caused by there resemblances. The teacher is only allowed to describe an object by one feature.

The best pair of learner/teacher, according to their results is a pragmatic learner (a learner who adapts to the teacher, improving his instructions policy during training) and a pedagogical teacher (a teacher who chooses instructions to avoid ambiguities). [5]

5. "Learning Behaviors with Uncertain Human Feedback" :

In this article, Human feedback is viewed as Gaussian probability distribution(positive it increases, negative it decreases). The environment developed is a virtual dog placed in square box where the age has to catch as many rats as possible. They combine two algorithms, "Expectation Maximization" and "Gradient Descent". A model is considered as a good performing one when the agent catches more rats during training phase and learn with the least number of steps possible. Their model learned approximately in 6 steps. [1]

2 Learning Behaviors with Uncertain Human Feedback

2.1 Presentation

We use [6] as a base of our project. It is an implementation of the article "Learning Behaviors with Uncertain Human Feedback", implementing the ABLUF algorithm. In the environment (Fig. 1, 2, 3), the dog is trying to catch the mouse. The goal of the teacher is to help him with that by giving him feedback at every steps. The test stops after 4 states and each state is about 15 steps depending on the teacher's satisfaction. The experiment in the article invited 40 trainers to participate.

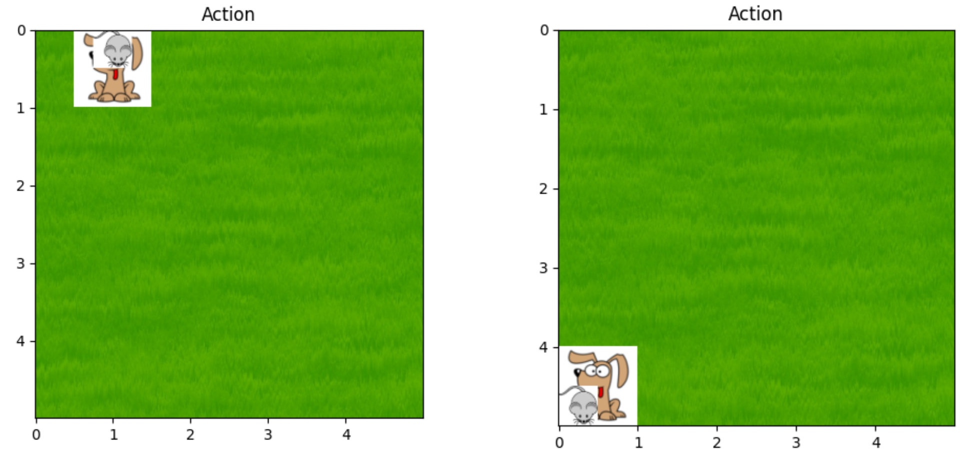

FIGURE 1 – Good feed back

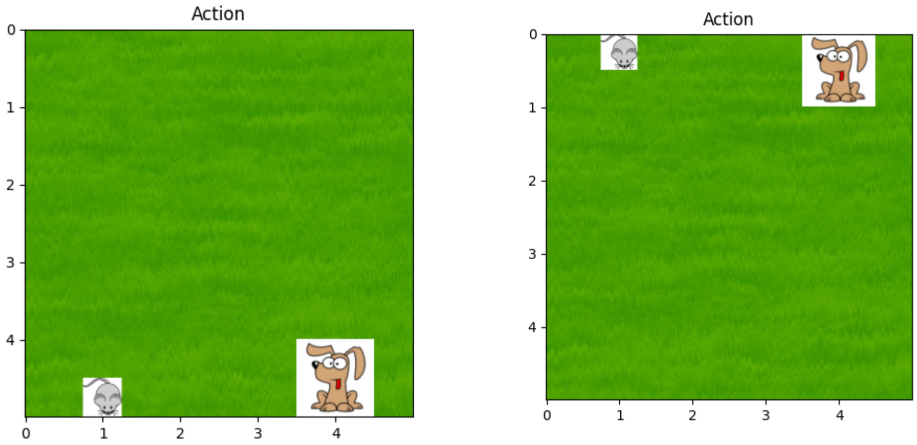

FIGURE 2 – Bad feed back

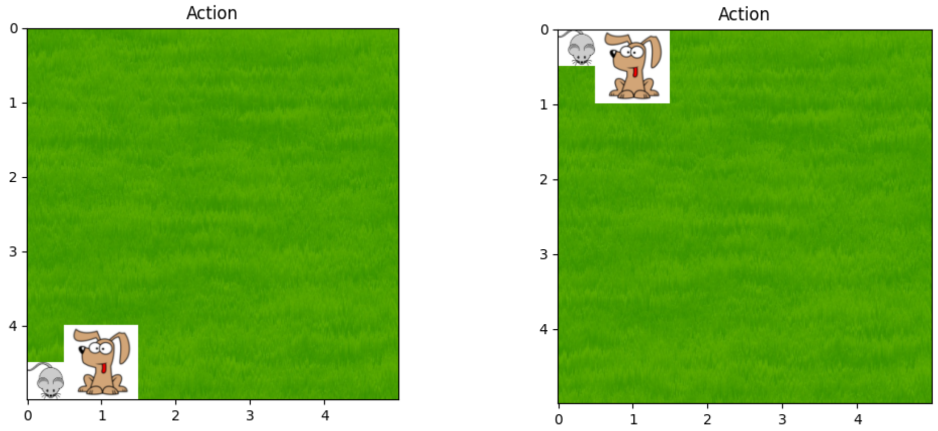

FIGURE 3 – Neutral feed back

In the article, the experiment lasts 4 states and about 60 steps. If the teacher thinks that the dog has learned enough, he can then put "3" to go to the next state.

There are three different feed backs :

0. Good feed back : mostly when the dog caught the rat (Fig 1) ;
1. Bad feed back : when the dog is far away from catching the rat (Fig 2) ;
2. Neutral feed back : when the dog is really close and the situation creates ambiguity for the teacher (Fig 3).

2.2 Results of experiment

After some tests, the results are not very satisfactory. Given the difference between the number of tests we can do by ourselves and the number of experiments done in the article, it seems very difficult, if not impossible to reproduce the results requested. Otherwise, it would be necessary to automate the tests, which would remove the principle of the article which consists in learning certain behaviors from human feedback. Our code can be found here [7].

The figures from our experiment (left) are to be compared with the ABLUF results in the article (the red one in the right figures), since that is the one used. (cf Fig. 4, 5, 6)

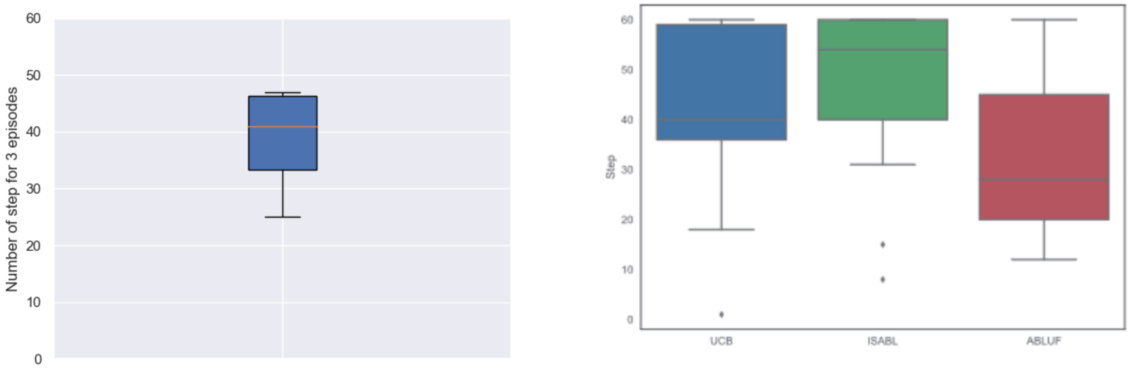

FIGURE 4 – Comparison with article (right) : Number of steps per episode

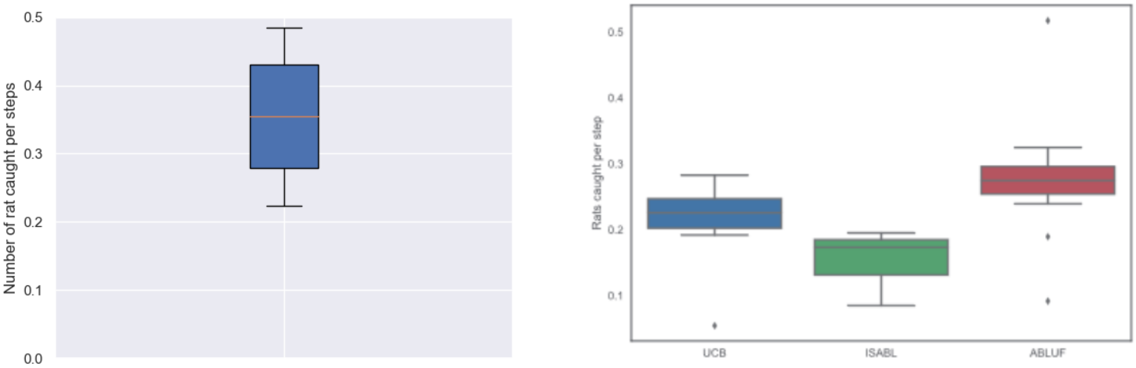

FIGURE 5 – Comparison with article (right) : Number of rat caught per steps

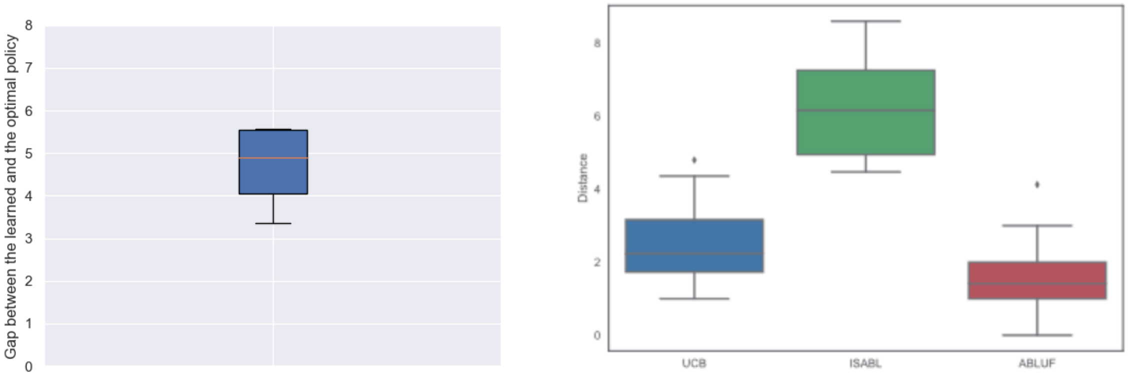

FIGURE 6 – Comparison with article (right) : Distance between the optimal and the learning policy

3 Alternative approaches

We will be using two different types of human feedback : speech and gesture recognition.

3.1 Gesture recognition

To implement the gesture recognition, we will use the model found here [8] with openCV and tensorflow.

We still have three feed backs, so we need three gestures and one more to get to the next state.

0. Good feed back : ’long live’ or ’stop’ gesture ;
1. Bad feed back : ’peace’ or ’two fingers’ gesture ;
2. Neutral feed back : ’fist’ gesture ;
3. Next state : ’thumbs up’ gesture.

These gestures have been selected so that there is the least chance of getting the wrong sign. The "long live" gesture and the "peace" gesture can be confused with the "fist" gesture, which is not as bad as confusing the good and bad feed back. (Annexe : 5)

3.2 Speech recognition

One of the alternative feedback that we implemented was done using Speech recognition. In order to acquire and use speech, we used an existing python module that can be found in the library “SpeechRecognition”. We then followed the tutorial [9] and adapted the problem to our case.

The user then can say four different words, so that it will easier to process later on :

Here’s what we used to present the feedback

0. Good feed back : Saying the word "Good" ;
1. Bad feed back : Saying the word "Bad" ;
2. Neutral feed back : Saying the word "Neutral" ;
3. Next state : Saying the word "Next".

Some difficulties were encountered, such as not always understanding the meaning of the word , at the beginning the user had the option to say “Next State” to go to the other state but with some tests we realized that it wasn’t transcribing as wished, so the solution provided, was to shorten the word and only say next to indicate passing the next state.

Another type of common error was more related to integrating the package with our model. Our initial version kept crashing and we were not able to record all the training, we solved this problem by using code inspired by the following [10]. When we encounter a problem during the recognition state we consider it as neutral feedback.

4 Results of the different feedbacks

We decided to use of the algorithm as our base : "ABLUF", we will be comparing the type of feedback given.

Here are some the results :

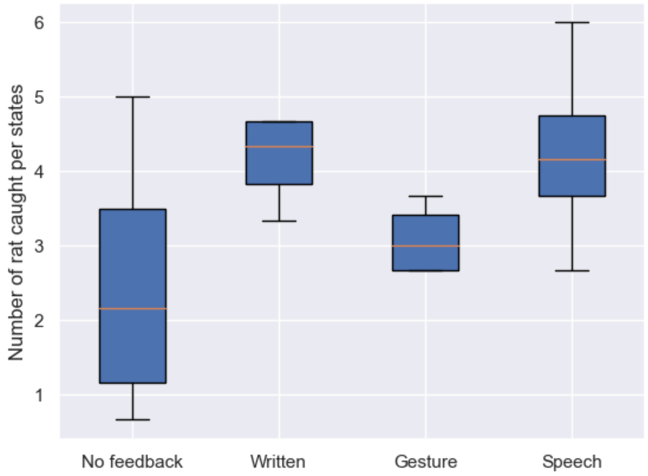

FIGURE 7 – Number of rats caught per state for the different feedbacks

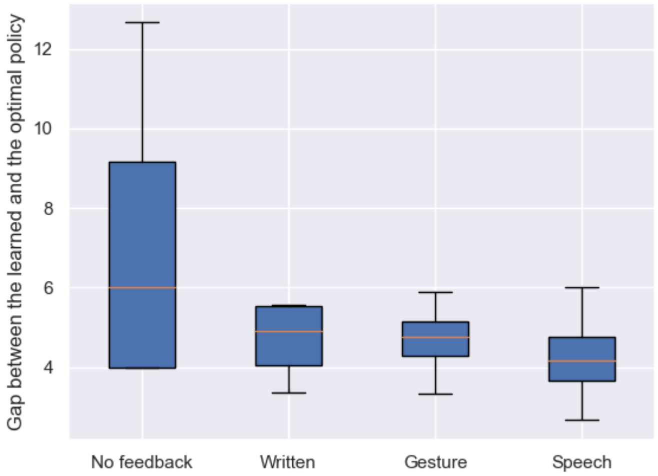

FIGURE 8 – Gap between the learned and optimal policy

5 Discussion of our methods and results

In figure 7, we can notice that the mouse get caught on a average of 4 times per state when human feedback is included. Speech and Written have pretty similar results. The case with no feedback has the lowest average, the rat is caught on average 2 times per state. It were expected since the model was training on its own , no feedback was taken in consideration while computing the policy. The gap between learned and optimal policy (Fig.8) is the highest for the case of no feedback and the lowest for the Written feedback. Here Gesture and Speech feedback have close results.

The conclusion on each feedback we implemented can be found in the table 2 where the advantages and limitations of each feedback is described.

Feedback type	Advantages	Limitations
Written	Least errors while taking user input, no delay between feedback and new action	Takes a lot of time
Gesture	More interesting to learn with actual human gesture similar to real life learning methods	Not detecting the gesture , learner can make mistakes
Speech	Close to real life approaches	Time consuming (around 30/40 minutes per training) , Recognising the wrong word , delay between recognition and taking new action

TABLE 2 – Comparison of the feedback implemented

Références

[1] B. A. Xu He, Haipeng Chen, “Learning behaviors with uncertain human feedback,” 2020, accessed : 20 décembre 2022.

[2] P. Christiano *et al.*, “Deep reinforcement learning from human preferences,” 2017, accessed : 20 décembre 2022.

[3] T. R. Sumers *et al.*, “Learning rewards from linguistic feedback,” 2021, accessed : 20 décembre 2022.

[4] D. XU *et al.*, “Deep reinforcement learning with implicit human feedback,” 2019, accessed : 20 décembre 2022.

[5] M. C. Hugo Caselles-Dupré, Olivier Sigaud, “Overcoming referential ambiguity in language-guided goal-conditioned reinforcement learning,” 2022, accessed : 20 décembre 2022.

[6] “Learning-behaviors-with-uncertain-human-feedback,” https://github.com/hlhllh/Learning-Behaviors-with-Uncertain-Human-Feedback, 2020.

[7] “Social robotics project,” https://github.com/SolennDumont/Learning-from-Human-Feedback-, 2022.

[8] TechVidvan, “Real-time hand gesture recognition using tensorflow & opencv,” https://techvidvan.com/tutorials/hand-gesture-recognition-tensorflow-opencv/, 2021, accessed : 19/12/2022.

[9] D. Amos, “The ultimate guide to speech recognition with python,” https://realpython.com/python-speech-recognition/.

[10] A. Zhang, “Recognize speech input from the microphone,” https://github.com/Uberi/speech_recognition/blob/master/examples/audio_transcribe.py.

Annexe A : Gesture recognition examples

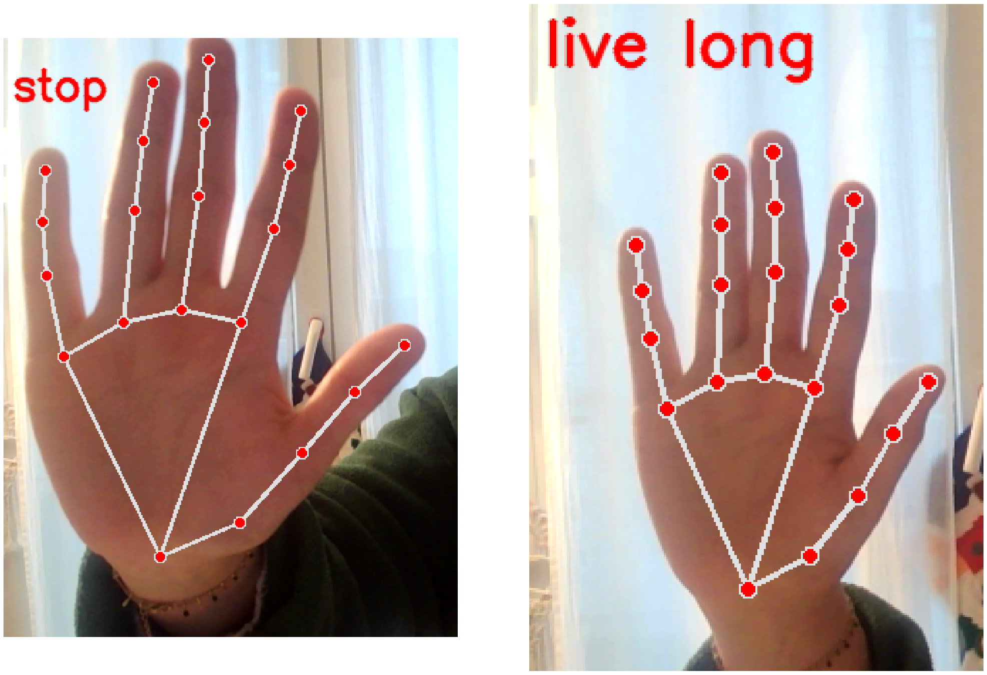

FIGURE 9 – Good feed back gesture

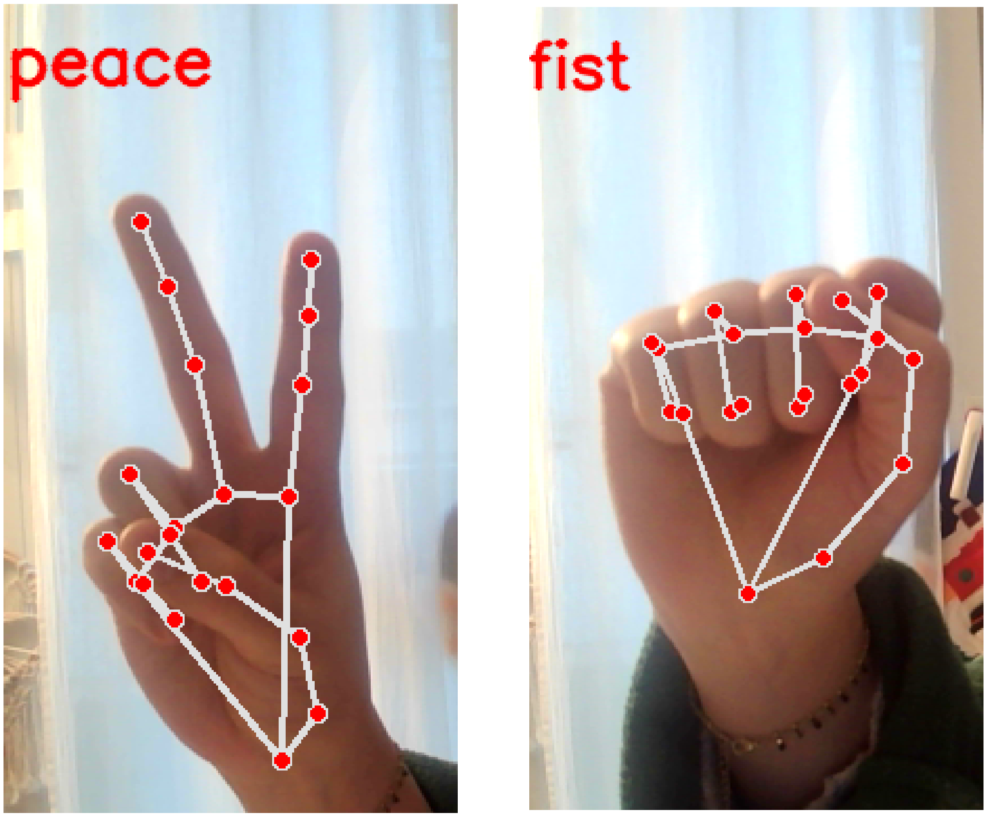

FIGURE 10 – Bad and neutral feed back

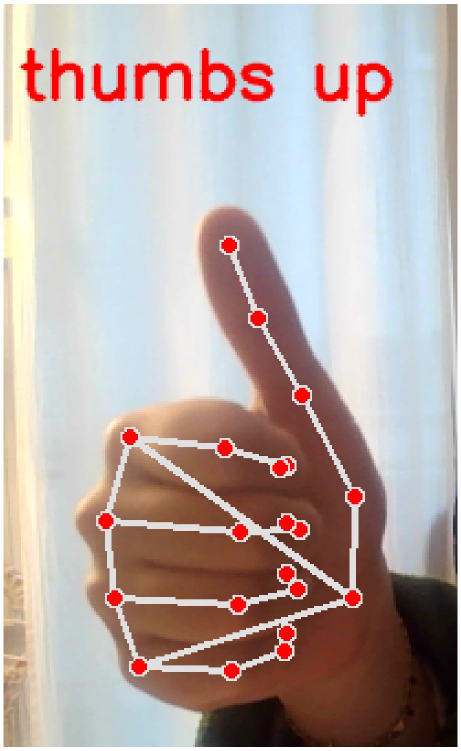

FIGURE 11 – Next state