In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 759-766, Daejeon, Korea (2016)

Multi-modal Integration of Dynamic Audiovisual Patterns for an Interactive Reinforcement Learning Scenario*

Francisco Cruz, German I. Parisi, Johannes Twiefel, and Stefan Wermter

***Abstract*— Robots in domestic environments are receiving more attention, especially in scenarios where they should interact with parent-like trainers for dynamically acquiring and refining knowledge. A prominent paradigm for dynamically learning new tasks has been reinforcement learning. However, due to excessive time needed for the learning process, a promising extension has been made by incorporating an external parent-like trainer into the learning cycle in order to scaffold and speed up the apprenticeship using advice about what actions should be performed for achieving a goal. In interactive reinforcement learning, different uni-modal control interfaces have been proposed that are often quite limited and do not take into account multiple sensor modalities. In this paper, we propose the integration of audiovisual patterns to provide advice to the agent using multi-modal information. In our approach, advice can be given using either speech, gestures, or a combination of both. We introduce a neural network-based approach to integrate multi-modal information from uni-modal modules based on their confidence. Results show that multi-modal integration leads to a better performance of interactive reinforcement learning with the robot being able to learn faster with greater rewards compared to uni-modal scenarios.**

I. INTRODUCTION

Human-Robot Interaction (HRI) has become an increasingly interesting area of study among developmental roboticists since robot learning can be speeded up with the use of parent-like trainers who deliver useful advice, allowing robots to learn a specific task in less time compared to a robot exploring autonomously [1]. In this regard, the parent-like trainer guides the apprentice robot with actions that allow to enhance its performance in the same manner as external caregivers may support infants in the accomplishment of a given task, with the provided support frequently decreasing over time. This teaching technique has become known as parental scaffolding [2].

When interacting with their caregivers, infants are subject to different environmental stimuli which can be present in various modalities. In general terms, it is possible to think about some of those stimuli as guidance that the parent-like trainer delivers to the apprentice agent. Nevertheless, when more modalities are considered, issues can emerge regarding the interpretation and integration of multi-modal information, especially when multiple sources are conflicting or ambiguous (e.g. yielding low confidence levels [3]). As a consequence, the actions to follow may not be clear and misunderstood, and hence, may lead the apprentice agent to a decreased performance when solving a task [4].

* The authors gratefully acknowledge partial support by the Universidad Central de Chile, CONICYT scholarship 5043, the DAAD German Academic Exchange Service (Kz:A/13/94748) under CASY project, the German Research Foundation DFG under project CML (TRR 169), and the Hamburg Landesforschungsförderungsprojekt.

Francisco Cruz, German I. Parisi, Johannes Twiefel, and Stefan Wermter are with the Knowledge Technology Institute, Department of Informatics, University of Hamburg, Germany. Emails: {cruz, parisi, twiefel, wermter}@informatik.uni-hamburg.de. See: http://www.informatik.uni-hamburg.de/wtm/.

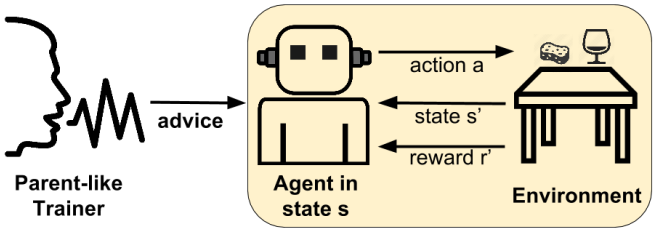


Fig. 1. An interactive reinforcement learning approach with policy shaping. The agent autonomously performs action a in state s obtaining reward r' and reaching the next state s'. In selected states, the parent-like trainer advises the apprentice agent changing the action to be performed in the environment.

In this work, we present a multi-modal interactive reinforcement learning scenario which consists of a robot learning a domestic task. The robot can manipulate two objects with the goal of cleaning a table. During the apprenticeship process, advice can be provided by a parent-like trainer using audiovisual inputs, respectively speech and gestures. Our proposed architecture is able to process information from multiple sources with the use of a neural associative memory that computes multi-modal advice as a function of the recognition and confidence of uni-modal modules. We present a set of experiments using 7 possible advice classes from audiovisual inputs, showing that multi-modal integration leads to a better performance of interactive reinforcement learning, with the robot being able to learn using a smaller number of training episodes compared to uni-modal scenarios.

II. RELATED WORK

A. Interactive Reinforcement Learning

Reinforcement Learning (RL) [5] is an approach based on behavioural psychology where an agent autonomously explores its environment in order to find an optimal policy to perform a given task. As such, the agent selects at each state an action to perform for obtaining a reward and reaching a new state. This cycle is shown in the yellow square in Fig. 1.

A common problem which still remains open is the excessive time in terms of training episodes required by the agent to learn a proper policy. In this regard, interactive

reinforcement learning (IRL) has added an external parent-like trainer (as shown in Fig. 1) in order to speed up the apprenticeship process by either reward or policy shaping [6] [7]. Although IRL has been implemented in robotic scenarios, a general problem is that the communication interface between the trainer and the robot has not been developed in a natural manner for domestic scenarios. For instance, Suay & Chenova [8] addressed an IRL task where the parent-like trainer was able to deliver guidance using a graphic interface built from a camera image and adding buttons and bars for interaction. Another IRL approach was proposed by Knox [9], in which the device used to deliver feedback to the robot was a presentation control (a presenter), allowing to change between positive and negative reward.

In both the aforementioned approaches, the interfaces are useful in terms of accomplishing the interaction with an external trainer. Nevertheless, these interfaces are quite tedious and impractical for non-expert trainers taking into account home-like environments, where external trainers should be able to use their natural communication skills (e.g. speech and gestures). Therefore, it is much more desirable to have more natural interactive scenarios where external parent-like trainers can deliver their instructions similar to caregivers instructing infants.

B. Multi-modal Integration

People are constantly subject to different perceptual stimuli through different modalities such as vision, audition, and touch among others. Such modalities are used to perceive information and process it independently, in parallel, or integrating the received information to provide a coherent and robust perceptual experience. Similarly, humanoid robots work with many of these sensory modalities and the way of processing and integrating the information coming from various sources is currently an important research issue in autonomous robotics. In HRI scenarios, robots can take advantage of such multi-sensory information in order to improve their capabilities when any sensory modality is limited, lacking, or unavailable.

For instance, early work by Andre et al. [10] proposed a multi-modal integration of speech and gestures for human-computer interaction using a tactile glove to identify hand gestures and a microphone array for speech recognition. The system functionality was limited to manipulate geometric objects on topographical maps. In robotic scenarios, Wermter et al. [11] designed a neurobiologically inspired robot for multi-modal integration and topological organization of actions with an associative memory. Their work integrated motor, vision, and language representations for learning by demonstration. Lacheze et al. [12] presented an approach for the recognition of static patterns fusing audio and video. In their work, auditory information was used to recognize objects that were partially occluded and therefore difficult to detect using only vision. Sanchez-Riera et al. [13] presented a scenario with a robot companion that performs audio-visual fusion for speaker detection using a multi-modal Gaussian mixture model. The approach detected multiple speakers in a domestic scenario with information from two microphones and two cameras mounted on a humanoid robot. Kimura & Hasegawa [14] used an incremental neural network to integrate real-time information in order to estimate attributes for unknown objects. The method used an RGB-D camera, a stereo microphone, and pressure and weight sensors to process different modalities. Ozasa et al. [15] proposed the integration of image and speech recognition confidence values to improve the recognition accuracy of unknown objects using logistic regression. In their approach, the confidence integration does not consider the case in which predicted labels are in contradiction. Moreover, in order to obtain improved recognition, it is also necessary to estimate proper logistic regression coefficients.

Nevertheless, in domestic scenarios and dynamic environments, assistive robot companions still need to understand and interpret instructions faster and more efficiently, yielding the integration of available multi-sensory information with different confidence levels in a consistent mode.

III. ROBOTIC DOMESTIC SCENARIO

In previous work [1], we developed an IRL scenario with automatic speech recognition to guide an apprentice robot in the achievement of a task. In this paper here, we extend the approach to incorporate visual information and integrate it with audio as a more robust guidance during the apprenticeship process. The robotic scenario consists of a humanoid robot in front of a table to clean it. The scenario comprises two objects that the robot can manipulate using its gripper. The two objects are:

i. *cup*, which is initially placed at any location of the table and should be moved in order to finish the cleaning task,
ii. *sponge*, which is used along with the robot's hand to clean different positions of the table.

For each object, we defined three locations: the *right* and *left* parts of the table, and an additional position defined as *home*, where the sponge should be placed when not in use. Moreover, in this scenario the robot is allowed to perform seven action classes:

i. *get*, which allows the robot to pick up the nearest object to its gripper,
ii. *drop*, which allows the robot to put down the object held in its hand,
iii. *go <location>*, which moves the robot's gripper to some of the defined locations; therefore, there are three different action classes which are yielded from this action, i.e. *go home*, *go left*, and *go right*,
iv. *clean*, which allows the robot to clean the table surface at the current hand position,
v. *abort*, which cancels the execution of the cleaning task at any time.

The vector state is represented using four variables as $s_t = [handObject, handPosition, cupPosition, sideCondition]$, where *handObject* is the object held in the robot's hand, *handPosition* is the current hand position of the robot, *cupPosition* is the position of the cup, and

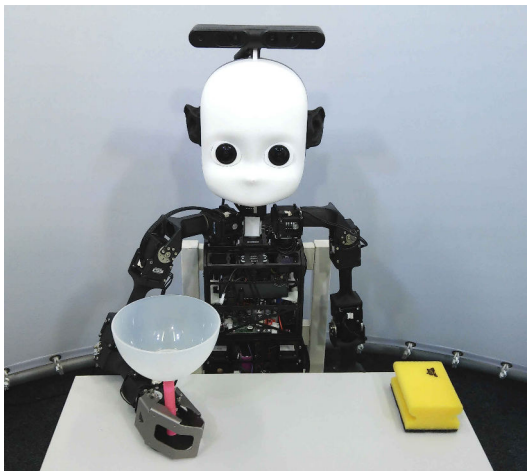


Fig. 2. Cleaning scenario with the NICO robot. Our scheme is composed of two objects, 3 locations, and 7 action classes.

sideCondition is the surface condition of each part of the table.

The robot task finishes when both sides of the table are cleaned, obtaining a reward of 1. In case that the robot cannot continue the task execution, then it receives a negative reward of -1. During a learning episode, intermediate states lead to a small negative reward of -0.01 to encourage faster transitions to the final state. RL is performed using SARSA algorithm with learning rate $\alpha = 0.3$, discount factor $\gamma = 0.9$, and ϵ-greedy action selection with $\epsilon = 0.1$. In the IRL approach, we use probability of advice of 0.3.

Although the robot is able to perform actions autonomously using RL, we use a parent-like trainer to advise the robot at specific steps about what action to perform next in order to reduce the time required to learn the shortest sequence of actions for finishing the task.

For our scenario, we define a set of possible advice classes that can be given to the robot by a parent-like trainer. Each advice class has a spoken representation in a domain-based language and a visual representation with gestures from vision. The advice can be delivered at any time using speech, gestures, or both with the following advice classes: *go left, go right, go home, get, drop, clean,* and *abort*.

For instance, let us now suppose that the cup is located on the left side of the table at the beginning. The initial position of the robot hand is the location *home*, and we want to finish with the hand free and above *home* with both sides of the table clean. The following example shows the shortest episode to complete this task successfully: *get, go right, clean, go home, drop, go left, get, go right, drop, go home, get, go lef trainer-liket, clean, go home, drop*. Fig. 2 shows an example of the domestic scenario with our Neural Inspired COmpanion (NICO) robot.

Fig. 3 shows the overall architecture of our system, where we use a microphone and a depth sensor to capture the advice from the parent-like trainer that is subsequently integrated and sent to the IRL algorithm as one single piece of consistent advice. The integrated advised action is then sent to the NICO robot to be performed using the *pypot* library [16], allowing to control the robot actuators either in real or simulated environments.

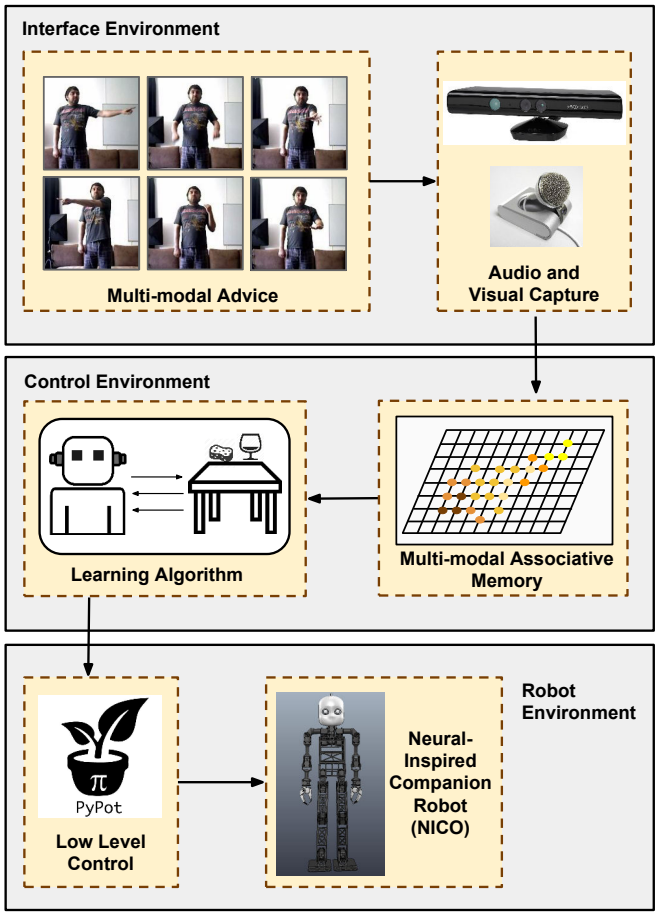


Fig. 3. Overall view of the system architecture. In the interface environment, we use the robot with a microphone and a depth sensor to capture advice from the parent-like trainer. In the control environment, we integrate the advice and send it to the IRL algorithm with a confidence value associated to decide when a valid advice is considered according to a defined threshold. The integrated advised action is then sent to the robot environment where a NICO robot performs the action using the *pypot* library which allows to control the robot actuators either in the real or simulated environment. In this paper, we are particularly focused on the speech and gesture representations and the integration of them.

IV. OUR APPROACH

In our architecture, a parent-like trainer interacts with an apprentice robot using speech and gestures as guidance for the cleaning scenario. In this work, we are particularly focused on processing audiovisual inputs and their integration. The following subsections describe how each modality module is implemented and how they are integrated in order to obtain a unified advice to shape a more effective guidance for the robot learning task.

A. Automatic Speech Recognition

To understand the verbal commands, the apprentice robot processes audio data and recognizes the given advice by applying an automatic speech recognition (ASR) system that

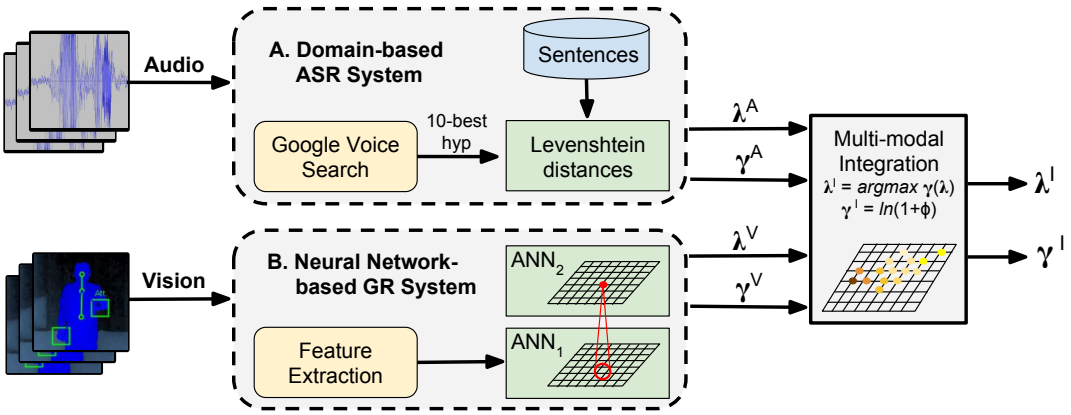


Fig. 4. Overall view of the system architecture. The domain-based ASR system (on top) processes the audio input modality to obtain an audio advice label λ^A and an audio confidence value γ^A and the neural network-based gesture recognition system (at bottom) processes the visual input modality to obtain a visual advice label λ^V and a visual confidence value γ^V. Afterwards they become the input of the multi-modal integrative system implemented by an associative neural architecture to obtain the integrated advice label λ^I and the integrated confidence value γ^I.

is based on *Google Voice Search* (GVS) [17], a cloud-based ASR service to process audio data captured by a local microphone and generating hypotheses for the corresponding text representation. As GVS is usually applied in web searches, the involved language models are optimized for that task and not for the given scenario. To overcome the issue of out-of-domain language models, we utilized *DOCKS* [18], a post-processing technique to fit the ASR hypotheses provided by GVS to the given HRI domain.

For our HRI scenario, a set of robot commands is defined and represented by a list of sentences. To identify the best-matching hypothesis out of the list of sentences, the phonemic representation of the ASR hypothesis is compared to the phonemic representation of each sentence in the list. For this task, the Levenshtein distance [19] is employed to calculate the difference between phoneme sequences. After calculating the Levenshtein distance between the ASR hypothesis and each sentence of the list, the sentence possessing the shortest distance is chosen as the best matching result. To improve the technique, the Levenshtein distance is calculated for the ten best hypotheses provided by GVS. Section A in Fig. 4 summarizes the functional principle of the ASR system employed in our architecture.

The predicted audio label is computed as $\lambda^A = \operatorname{argmin} \mathcal{L}(h_i, s_j)$, where $\mathcal{L}$ is the Levenshtein distance in our ASR system. The confidence value was computed as $\gamma^A = \max(0, 1 - \mathcal{L}(h_i, s_j)/|s_j|)$ with $h_i \in H$ (set of the 10-best hypotheses) and $s_j \in S$ (set of reference sentences) both in phonemic representation.

B. Gesture Recognition

For gesture recognition, we used an extended version of the HandSOM framework [20] for learning gestures from depth map videos using self-organizing neural networks. Our learning architecture consists of two hierarchically arranged self-organizing neural networks (Fig. 4.B). The use of hierarchical self-organization has been shown to be an effective method for recognizing human motion [20] [21]. Furthermore, for each predicted label we also estimate a confidence value that expresses the degree of belief that the prediction is correct based on a set of predictions over a given time window. We now describe the gesture features that we extract from depth video sequences used as input for the neural network learning architecture and the hierarchical processing for learning a set of training gestures and predicting gesture labels from novel input.

1. Feature Extraction

Hand motion from depth images was extracted to represent gestures as hand-independent motion sequences. To encode motion patterns, only the motion information of the most salient hand performing a gesture was taken into account. In case that both hands are used, the type of interaction between the hands is considered, i.e. *physical* if the two hands overlap, or *symmetric*, if they follow the same (mirrored) behavior (Fig. 5.d). We consider a set of motion descriptors for a given set of tracked body joints, i.e. hands and head. For each frame i, the gesture feature vectors were of the form $\mathbf{m}_i = (s_i, \mathbf{v}_i, \varphi_i, h_i, \lambda_i)$, where s_i is the hand interaction type, λ_i is the annotated gesture label, $\mathbf{v}_i$ is the hand 3D motion intensity in terms of pixel difference from consecutive frames, φ_i is the hand angle with respect to the y axis in the image plane, and h_i is the distance from the head [20]. Training videos were recorded with a Kinect sensor operating at 30 frames per second, from which we estimated the 3D skeleton model using the OpenNI/NITE framework. To attenuate noise, we computed the median value for each joint every 3 frames, resulting in a total of 10 feature vectors per second. These vector sequences are then clustered by a hierarchical learning architecture to obtain a representation of prototype gestures from a set of training samples.

2. Learning Architecture

Our learning model consists of two hierarchically arranged Growing When Required (GWR) networks [23] that incre-

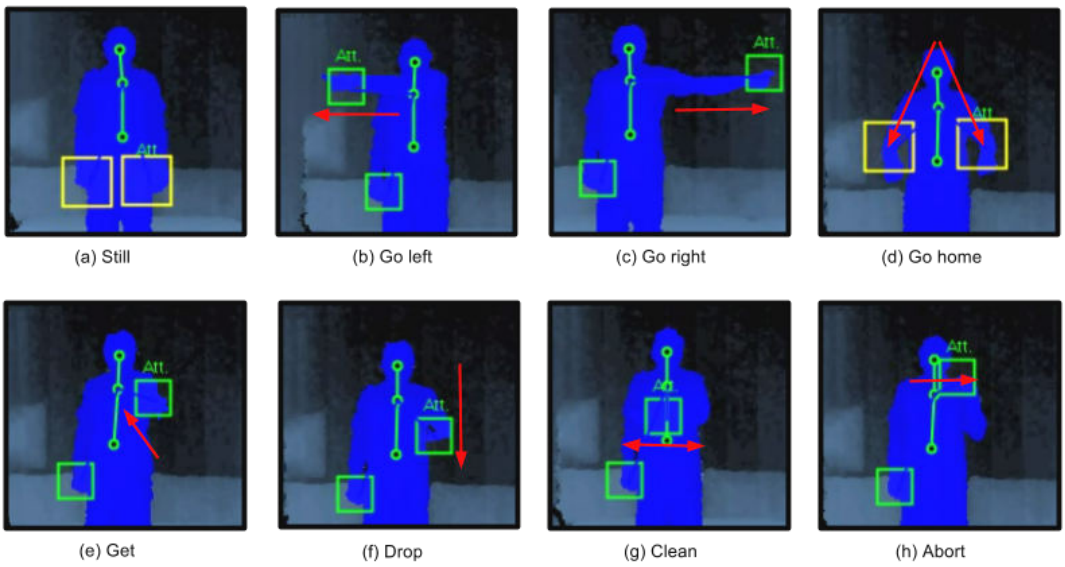


Fig. 5. Gestures used as advice in the robotic scenario. Red arrows represent the hand movement performed to advice the robot. The motion from the most salient hand is used to estimate the motion vector. In case that both hands are used, the type of hand interaction is considered (details in the text). Since gesture labels are seamlessly predicted from depth map video sequences, we add the label *still* to indicate no advice at that moment.

mentally obtain generalized representations of sensory inputs to learn latent spatiotemporal structure. Hierarchical learning is carried out by training the higher-level network with neuron activation trajectories from the lower-level network.

The GWR network is composed of a set of neurons and their associated weight vectors $\mathbf{w}_j$ linked by a set of edges. During the training, the network starts with two neurons and then dynamically changes its topological structure to better match the input space following competitive Hebbian learning [22]. The network growth process takes into account the overall activity of the network with respect to the input and the number of times that existing neurons have fired. The activity is calculated as a function of the distance between the input and its best-matching neuron. This allows the model to add new neurons whenever they are required, i.e. if the activity of the network with respect to the input is smaller than a given threshold a_T for a well-trained best-matching neuron (firing counter smaller than the firing threshold f_T). The GWR algorithm will then iterate over the training set until a given stop criterion is met, e.g. a maximum number of training iterations (epochs). The standard unsupervised training algorithm was presented in [23].

In our architecture, the network in the first layer receives as input the sequence of vectors $\mathbf{m}_i$. The network in the second layer is trained with neural activation trajectories from the first layer. These trajectories are obtained by computing the best-matching neurons of the input sequence $\mathbf{x}_i$ with respect to the trained network with N neurons, so that a set of trajectories is given by

$$\Omega(\mathbf{x}_i) = \{\mathbf{w}_{b(\mathbf{x}_i)}, \mathbf{w}_{b(\mathbf{x}_{i-1})}, \mathbf{w}_{b(\mathbf{x}_{i-2})}\}, \quad (1)$$

with $b(\mathbf{x}_i) = \arg\min_{i \in N} \|\mathbf{x}_i - \mathbf{w}_j\|$. After the training of the higher level network is completed, each neuron will encode a sequence-selective gesture segment from 3 consecutive frames. This mechanism allows to obtain specialized neurons coding the spatiotemporal structure of the input. For classification purposes, neurons created in this second layer are attached to gesture labels obtained from the training set. The GWR training algorithm for attaching labels to neural activation trajectories was discussed in [21]. The training parameters and number of neurons created after the training session are shown in Table 1.

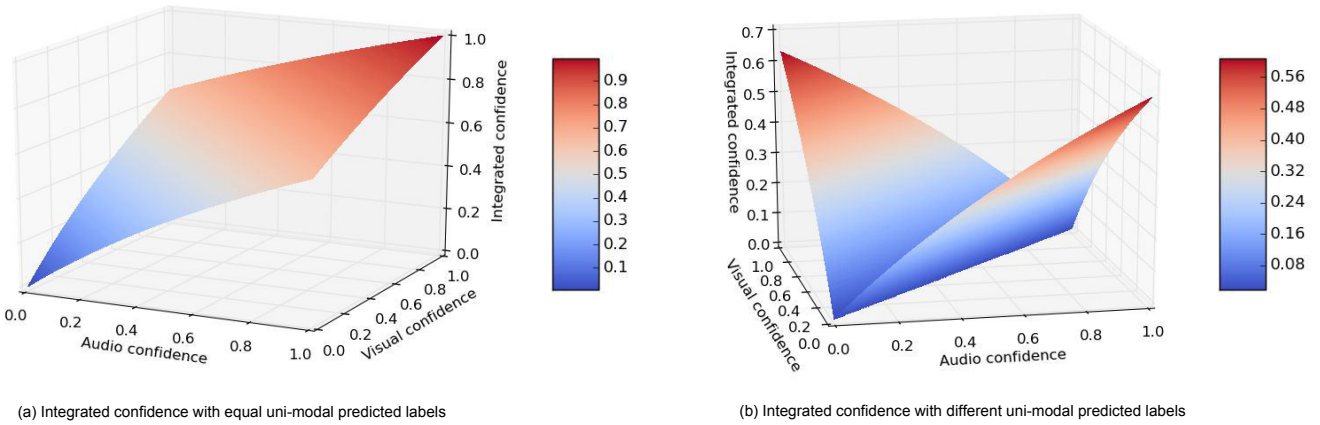
TABLE I
TRAINING PARAMETERS FOR GWR HIERARCHICAL LEARNING

Parameters	Network Layer 1, 2
Activation threshold	$a_T = \{0.85, 0.65\}$
Firing threshold	$f_T = 0.01$
Firing counter	$\tau_b = 0.3$, $\tau_n = 0.1$
Learning rates	$\epsilon_b = 0.1$, $\epsilon_n = 0.01$
Maximum edge age	200
Training epochs	100
N. of neurons after training	$\{337, 316\}$

In the hierarchical architecture, a label prediction is returned every 3 frames in a sliding window scheme. We considered the last 5 observations and computed the statistical mode that returns the most frequent value in a set. Given the set of predictions Λ^V and denoting N as the number of occurrences of the mode within Λ^V, the confidence value is then defined as $\gamma^V = N/|\Lambda|$, yielding a maximum confidence value of 1 and a minimum of 0.2. Since we processed 10 feature vectors per second and we compute the mode of the last 5 predictions, our system returns a predicted label λ^V and a confidence value γ^V for a window of 7 frames (0.7 seconds).

C. Multi-modal Integration of Audiovisual Patterns

A general overview of the architecture including the speech and gesture processing is depicted in Fig. 4, where



(a) Integrated confidence with equal uni-modal predicted labels

(b) Integrated confidence with different uni-modal predicted labels

Fig. 6. Confidence values used in the neural network-based associative architecture. While in (a) the corresponding output labels for audio and visual modalities are the same, in (b) they are different. During the training we use a grid of 20x20 values for each modality and for the validation an equal distribution with a grid of 100x100 values.

λ and γ are the label and the confidence value respectively. First, the audio and visual sensory inputs are individually processed. Then, the outputs, i.e. predicted labels and confidence values, become inputs for the multi-modal integration system. To ingrate the two aforementioned sensory modalities, we propose a mathematical model and implement it with a neural associative memory.

1. Mathematical Model

Our mathematical function relates the predicted advice classes and confidence pairs from uni-sensory input, respectively denoted as (λ^A, γ^A) for audio and (λ^V, γ^V) for vision.

The integrated predicted label λ^I is calculated according to the highest confidence value:

$$\lambda^I = \underset{\lambda}{\operatorname{argmax}}\ \gamma(\lambda) \tag{2}$$

In other words, if the audio and visual labels λ^A and λ^V are different, then the integrated label λ^I takes the value from the modality which has the biggest confidence value.

On the other hand, the integrated confidence value is computed by the function:

$$\gamma^I = \ln(1 + \phi), \tag{3}$$

where ϕ is a time-varying parameter which depends on each label λ and confidence value γ. We call this parameter the *likeness parameter* and it is obtained according to the following equation:

$$\phi = \begin{cases} \gamma^A + \gamma^V & \text{if } \lambda^A = \lambda^V \\ |\gamma^A - \gamma^V| & \text{if } \lambda^A \neq \lambda^V \end{cases} \tag{4}$$

Therefore, if the labels λ^A and λ^V are the same, then the confidence value γ^I is calculated using $\phi = \gamma^A + \gamma^V$ in order to strengthen the integrated confidence level over the prediction made from both devices. On the contrary, if the labels λ^A and λ^V are different, then the integrated confidence value γ^I is calculated using $\phi = |\gamma^A - \gamma^V|$ in order to diminish the confidence level given the differences in the class predictions.

This function yields an integrated confidence value $\gamma^I \in [\ln(1), \ln(3)] = [0, 1.0986]$. We use a unity-base normalization to rescale the range of confidence between 0 and 1:

$$\gamma^I = \frac{\gamma^I - min(\Gamma)}{max(\Gamma) - min(\Gamma)}. \tag{5}$$

where Γ is the set of all possible confidence values γ^I.

Fig. 6 shows the integrated confidence values when the predicted audio and visual labels are the same (a) and different (b).

2. Neural Network-based Associative Architecture

To implement the proposed mathematical model, we develop an associative neural architecture with a complex-valued quadratic neuron [24] to define a new two-dimensional grid on the output space as presented in [25]. For an input vector $X \in \mathbb{C}^n$, the scalar complex output is $y = X^* A X$, where $A \in \mathbb{C}^{n \times n}$ is the weight matrix and X^* denotes the conjugate transpose. The output can be written as the summation of the individual terms that involve the components of X and A:

$$y = \sum_{j=1}^{n} \sum_{k=1}^{n} \bar{x}_j x_k a_{jk}. \tag{6}$$

The gradient descent learning rule that minimizes the mean-square error is:

$$\triangle A = \alpha \varepsilon \bar{X} X^T, \tag{7}$$

where α is a small real-valued learning rate. For a given input vector X, the desired output Y to be used in the learning algorithm is defined as the nearest intersection point of the grid lines of the complex plane. In practice, a function Ψ is defined that rounds to the nearest integer for grid lines spaced at a fixed distance δ in both directions:

$$\Psi(Y) = \frac{round(\delta Re(Y))}{\delta} + i\frac{round(\delta Im(Y))}{\delta}. \tag{8}$$

This function creates a virtual grid where the output snaps onto the nearest grid corner. The training algorithm is as follows:

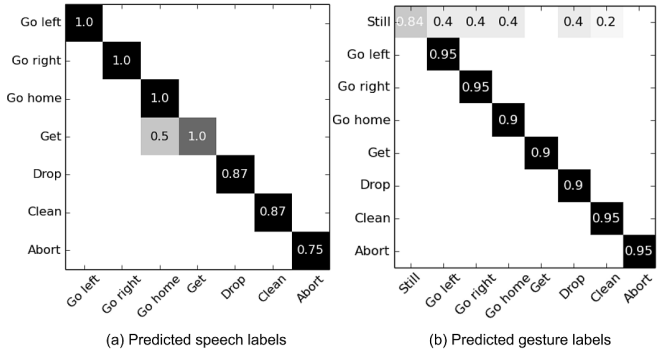


Fig. 7. Confusion matrices with the average confidence values for predicted (a) speech and (b) gesture labels. The input speech advice of *get* was predicted in one occasion as *go home* with confidence of 0.5. Nevertheless, all other predicted labels were correctly classified with high confidence values over 0.75. The gesture *still* was in some occasions misclassified with low confidence of 0.4 and 0.2. This was due to the transition from one gesture to the next and the use of the last three consecutive frames for the prediction. Regardless, all the gestures were correctly classified with high confidence values over 0.84.

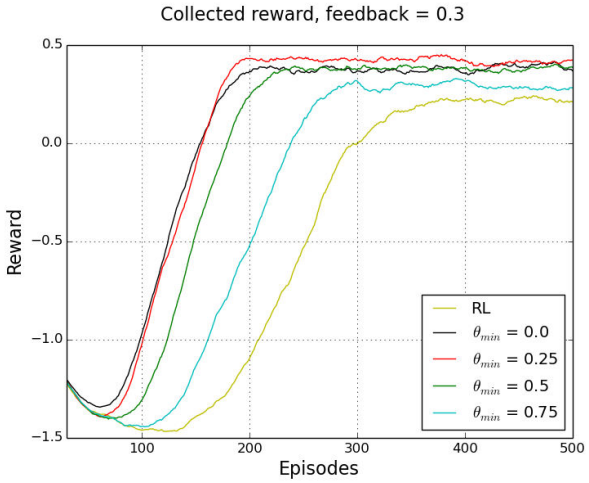


Fig. 8. Integrated rewards with different thresholds of minimal confidence level to be considered as a valid advice. The best performance is observed with $\theta_{min} = 0.25$ depicted in red. Autonomous RL is shown as a base in yellow color.

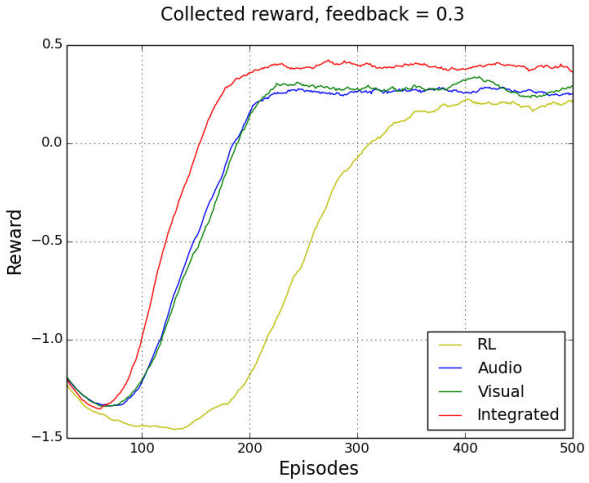


Fig. 9. Collected rewards with advice from audio and visual modalities are shown in blue and green respectively. Autonomous RL is shown as a base in yellow color. Working with advice from the multi-modal integration approach, the IRL agent is able to collect faster and greater reward in comparison to individual advice approaches.

0) Initialize the weights of the neuron with random values
1) Compute Y
2) Compute $d = \Psi(Y)$
3) Update the weights of the neuron using Eq. 7

At each iteration, the steps (1) to (3) are carried out for all the input vectors, so that a cluster in the input space will map to a similar region in the output space due to the continuity of the activation function. The stop criterion can be a fixed number of iterations, a decreasing learning rate, or a given minimum mean-square error over all inputs.

V. EXPERIMENTS AND RESULTS

For our experiment set-up, we implemented the robotic domestic scenario described in section III. We recorded pieces of advice from a parent-like trainer for all advice classes including speech and gestures with four repetitions for each one. Recorded advice allowed us to control better the experimental set-up for repeating the learning process under different situations. At recognition time, our goal was to predict the gesture label from novel audio and video sequences (λ^A, λ^V) and provide the confidence values (γ^A, γ^V) that expressed how reliable these predictions were.

Fig. 7a shows the confusion matrix with the average confidence values for the predicted speech labels whereas the confusion matrix with the average confidence values for the predicted gesture labels is shown in Fig. 7b. In the latter, we added the label *still* since the depth sensor is always processing visual information and this label allows to represent the fact that no gesture belonging to the advice classes is being recognized.

After processing each sensory input independently, the inputs integrated using our neural architecture to determine a combined λ^I and γ^I. We used a grid of 20×20 points for training and a subsequent validation grid of 100×100 points obtaining an average quantization error $e_q(n)$ of 0.05984 computed as $e_q(n) = x_q(n) - x(n)$ where $x(n)$ represents the sample sequences of the validation set, $x_q(n)$ the sample sequences of the training set, and $e_q(n)$ represents the sample sequences of the quantization error.

When working autonomously in the domestic scenario, the robot selects the actions using ϵ-greedy action selection policy with $\epsilon = 0.1$. We used interactive advice probability of 0.3 since it has been shown to be effective and small enough [4]. After the integration, we used different confidence levels to verify whether small confidence values benefit the learning scenario. Therefore, we considered $\gamma^I > \theta_{min}$ with θ_{min} being the minimum confidence threshold to be considered as a valid advice. In the case that the advice did not accomplish this minimal condition, the next action was selected through the aforementioned ϵ-greedy policy. We tested different thresholds $\theta_{min} \in \{0.0, 0.25, 0.5, 0.75\}$

observing that in general IRL works better with $\theta_{min} = 0.25$ in comparison with the other thresholds. Fig. 8 shows the average convoluted rewards for those θ_{min} values using 100 agents over 500 training episodes.

Finally, we use a fixed threshold $\theta_{min} = 0.25$ during the learning process to compare uni- and multi-modal advice in the IRL scenario. In uni-modal IRL approaches, collected rewards are close to each other in terms of the time needed for convergence (more than 200 episodes) as well as the maximal reward value (approximately 0.3). On the other hand, the multi-modal integrated IRL approach using both sensory inputs obtains the same level of reward as uni-modal approaches in fewer episodes (in this case, less than 200 episodes) and converges to greater reward (approximately 0.4). Therefore, the integrated information benefits the IRL performance, where greater rewards are accumulated faster in comparison to the use of uni-modal modules. Fig. 9 shows the average collected reward over 500 training episodes for the uni- and multi-modal learning procedure.

VI. CONCLUSIONS AND FUTURE WORK

We have proposed an interactive reinforcement learning scenario with multi-modal integration of dynamic audiovisual input advice. The architecture processes individually the input advice to classify them with a correspondent associated confidence value. Afterwards, our architecture integrates the input advice into one single label and confidence value. Although both sensory modalities show good advice prediction and confidence levels, the integrated advice leads to a better performance in our domestic scenario in terms of the accumulated reward and required learning episodes. In this regard, we have shown that our integration function allows to enhance the performance of a learning robot using multiple sources of information for a more natural trainer-like learning procedure.

Currently, our multi-modal IRL scenario runs in an off-line manner. Therefore, future work directions should consider experiments accounting for on-line interactions. Furthermore, experiments should also consider a wider number of parent-like trainers with different teaching characteristics.

REFERENCES

[1] F. Cruz, J. Twiefel, S. Magg, C. Weber, and S. Wermter. Interactive reinforcement learning through speech guidance in a domestic scenario. In Proceedings of International Joint Conference on Neural Networks IJCNN, pp. 1341–1348, Killarney, Ireland, 2015.

[2] E. Ugur, Y. Nagai, H. Celikkanat, and E. Oztop. Parental scaffolding as a bootstrapping mechanism for learning grasp affordances and imitation skills. In Robotica, vol. 33, pp. 1163–1180, 2015.

[3] J. Bauer, J. Dávila-Chacón, and S. Wermter. Modeling development of natural multi-sensory integration using neural self-organisation and probabilistic population codes. In Connection Science, vol. 27, no. 4, pp. 358–376, 2015.

[4] F. Cruz, S. Magg, C. Weber, and S. Wermter. Training agents with interactive reinforcement learning and contextual affordances. Accepted to IEEE Transactions on Autonomous Mental Development, 2016, doi: 10.1109/TCDS.2016.2543839.

[5] R. S. Sutton and A. G. Barto. Reinforcement learning: An introduction, A Bradford Book, 1998.

[6] A. L. Thomaz and C. Breazeal. Asymmetric interpretations of positive and negative human feedback for a social learning agent. In Proceedings of IEEE International Symposium on Robot and Human Interactive Communication RO-MAN, pp. 720–725, 2007.

[7] S. Griffith, K. Subramanian, J. Scholz, C. Isbell, and A. Thomaz. Policy shaping: Integrating human feedback with reinforcement learning. In Advances in Neural Information Processing Systems, pp. 2625–2633, 2013.

[8] H. B. Suay and S. Chernova. Effect of human guidance and state space size on interactive reinforcement learning. In Proceedings of IEEE International Symposium on Robot and Human Interactive Communication RO-MAN, pp. 1–6, 2011.

[9] W. B. Knox, P. Stone, and C. Breazeal. Training a robot via human feedback: A case study. In Proceedings of the International Conference on Social Robotics, pp. 460–470, 2013.

[10] M. Andre, V. G. Popescu, A. Shaikh, A. Medl, I. Marsic, C. Kulikowski, and J. Flanagan. Integration of speech and gesture for multimodal Human-Computer interaction. In International Conference on Cooperative Multimodal Communication, pp. 28–30, 1998.

[11] S. Wermter, M. Elshaw, C. Weber, C. Panchev, and H. Erwin. Towards integrating learning by demonstration and learning by instruction in a multimodal robot. In Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems IROS, Workshop on Robot Learning by Demonstration, pp. 72–79. 2003.

[12] L. Lacheze, Y. Guo, R. Benosman, B. Gas, and C. Couverture. Audio/video fusion for objects recognition. In Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems IROS, pp. 652–657, 2009.

[13] J. Sanchez-Riera, X. Alameda-Pineda, J. Wienke, A. Deleforge, S. Arias, J. Cech, S. Wrede, and R. Horaud. Online multimodal speaker detection for humanoid robots. In Proceedings of IEEE-RAS International Conference on Humanoid Robots, pp. 126–133, 2012.

[14] D. Kimura and O. Hasegawa. Estimating multimodal attributes for unknown objects. In Proceedings of the International Joint Conference on Neural Networks IJCNN, pp. 1–8, 2015.

[15] Y. Ozasa, Y. Ariki, M. Nakano, and N. Iwahashi. Disambiguation in unknown object detection by integrating image and speech recognition confidences. In Computer Vision – ACCV 2012, pp. 85–96, 2012.

[16] M. Lapeyre, P. Rouanet, J. Grizou, S. N'Guyen, A. Le Falher, F. Depraetre, and P.-Y. Oudeyer. Poppy: Open source 3D printed robot for experiments in developmental robotics. In Proceedings of the Joint IEEE International Conferences on Development and Learning and Epigenetic Robotics ICDL-EpiRob, pp. 173–174, 2014.

[17] J. Schalkwyk, D. Beeferman, F. Beaufays, B. Byrne, C. Chelba, M. Cohen, M. Kamvar, and B. Strope. Your word is my command: Google search by voice: A case study. In Advances in Speech Recognition, Springer US, pp. 61–90, 2010.

[18] J. Twiefel, T. Baumann, S. Heinrich, and S. Wermter. Improving domain-independent cloud-based speech recognition with domain-dependent phonetic post-processing. In Proceedings of the 28th AAAI Conference on Artificial Intelligence AAAI, pp. 1529–1535, 2014.

[19] V. I. Levenshtein. Binary codes capable of correcting deletions, insertions and reversals. In Soviet Physics Doklady, Vol. 10, pp. 707, 1966.

[20] G.I. Parisi, D. Jirak, and S. Wermter. HandSOM - Neural clustering of hand motion for gesture recognition in real time. In Proceedings of IEEE International Symposium on Robot and Human Interactive Communication RO-MAN, pp. 981–986, 2014.

[21] G.I. Parisi, C. Weber, and S. Wermter. Self-organizing neural integration of pose-motion features for human action recognition. In Frontiers in Neurorobotics 9:3, 2015.

[22] T. Martinetz. Competitive Hebbian learning rule forms perfectly topology preserving maps. In Proceedings of the International Conference on Artificial Neural Networks ICANN, pp. 427–434, 1993.

[23] S. Marsland, J. Shapiro, and U. Nehmzow. A self-organising network that grows when required. In Neural Networks, Vol. 15, pp. 1041–1058, 2002.

[24] G. Georgiou. Exact interpolation and learning in quadratic neural networks. In Proceedings of the International Joint Conference on Neural Networks IJCNN, pp. 230–234, 2006.

[25] G. Georgiou and K. Voigt. Self-organizing maps with a single neuron. In Proceedings of the International Joint Conference on Neural Networks IJCNN, pp. 1–6, 2013.