Abstract—This paper presents a cognitive learning system for robot recognition and composite action learning. The cognitive system of the robot is an artificial neural network trained to recognize and handle objects through imitation and backpropagation algorithm learning. The robot is first trained to learn the representation of action words, object categories and grounded language understanding. Following a human tutor's linguistic instructions, the robot autonomously transfers the grounding form directly basics knowledge to new higher level composite knowledge.

Keywords—cognitive robotics, neural network, humanoid robot.

I. INTRODUCTION

Research on autonomous and intelligent robots has become of critical importance due to the availability of humanoid robot platforms and their potential application in service robotics such as robot companions for the elderly. For example, the Honda Asimo robot [1], besides the remarkable kinematic movements, has recently been used to demonstrate the acquisition of cognitive tasks such as object recognition and naming. The humanoid robot “Partner”, developed by Toyota [2], has been used to perform fine, complex movements to play the violin. The iCub robot [3] has been developed as an open source platform to perform cognitive robotics experiments.

Most of behavioral experiments developed by intelligent robots in recent years require the demonstration of the task in strictly defined and controlled experimental conditions. However, to be able to handle the complexity of dynamically changing and unpredictable scenarios, as in service robotics, it is important to design robots capable to adapt to such dynamic environments. Cognitive robotics, which takes inspiration from studies on human cognition and neuroscience, can help the design of intelligent robots able to handle complex, dynamic tasks [4].

One important aspect of human cognition, which can be used to improve the design of intelligent robots, is that of linguistic communication. Human language is a formidable communication system that allows us not only to describe the status of events and objects in the real world, but also to teach new concepts via linguistic description. For example, a teacher or caregiver can describe objects and actions visible to the child by teaching the names of these entities. In addition, a child can be taught the names of new objects and action via combination of previous learned names (e.g. explain that a “zebra” is a “horse” with “stripes”, when a child has never see the zebra animal and is only familiar with the names of horses and striped patterns. Harnad [5] proposed an idea that symbols should be intrinsically linked to the agent’s ability to acquiring knowledge and categories from everyday experience. It is necessary for basic knowledge (words) to be directly grounded on physical categories. Subsequently, new knowledge can be formed through the combination of previously grounded basic words and concepts. For solving the symbol grounding problem, hybrid symbolic-connectionist models were originally proposed as ideal candidates [6]. More recently, alternative approaches have been introduced that are based on cognitive robotics methods [7].

In cognitive robotics research, various approaches and solutions have been proposed for social learning and communication between humans and robots to create and share knowledge via linguistic communication. For example, Steels [8] has implemented language games with robotic agents. Other methods [9][10][11] have also been proposed to deal with the symbol grounding problem, using fully connectionist models. For instance, in [11][12] the neural networks acquire a small set of basic categories through direct sensorimotor grounding symbol. The same networks were subsequently trained to acquire new higher-order categories by combination of the name of basic category. These networks were able to transfer the grounding from sensorimotor categories to higher order categories, which learned through symbol combination. Such an approach has also been used in evolutionary simulations of language origins [13][14][15]. Research on the connectionist implementation of grounded symbolic cognitive agents is still in progress. Some focus on the design of modular connectionist architectures (e.g. [16]).
Moreover, this approach has been extended to the grounding of abstract concepts [17].

This work builds on this connectionist approach to language grounding, and combines it with cognitive robotics modeling to focus on the indirect grounding of high-order concepts. The main theoretical contribution of this paper is to focus on the parallel development of action, vision and linguistic models to achieve social interaction and self-learning capabilities for robot to enhance cognitive development. In particular it extends the previous models in the literature (e.g. [11][12]) by using a modular ensemble of neural networks, and by looking at the parallel learning of perceptual categories and motor concepts. This neural architecture will allow a robot to first acquire a basic lexicon via direct sensorimotor learning, and then using linguistic instructions from a human teacher to learn complex, higher order concepts.

II. SIMULATION AND METHOD

A. The robot

1) Virtual robot specifications

The robotic model used for the experiments is a simulation of a humanoid robot, controlled by neural network, which is designed to learn composite concepts via linguistic instructions. In the simulation, the robot has the ability to learn the category of visual objects and to interact with them.

The agent in simulation is a small-size humanoid robot (Fig. 1). Each of the two arms consists of five-segments attached to a torso, and each of the two legs has four-segment linked to the torso. The robot has a total of 22 Degree of Freedom (DOFs): shoulder joint in vertical plane (2: one for left and one for right arm), shoulder joint in horizontal plane (2), upper arm joint (2), elbow joint (2), lower arm joint (2), neck joint (2: one for vertical plane and one for horizontal plane), hip joint in vertical plane (2), hip joint in horizontal plane (2), upper leg joint (2), lower leg joint (2), knee joint (2). In the current simulation, only two arms and the neck joint will be used, since the robot is standing on the ground and does not move. Nine objects were used for the training: Square, Cross and Dot, each one with either the color Red, Blue or Green. The robot can receive in the input retina different views of each object, and is taught the names of each object category. Moreover, the robot has to lean 16 basic actions: open arms, close arms, lift arms, arms down, elbow up and elbow down, those above if for the left and right robot arm movements, other four basic actions is neck up, neck down, neck left and neck right. The corresponding names for these basic actions are: "OPEN_LEFT_ARM", "LIFT_LEFT_ARM", "CLOSE_LEFT_ARM", "DOWN_LEFT_ARM", "LEFT_ELBOW_UP", "LEFT_ELBOW_DOWN", "OPEN_RIGHT_ARM", "CLOSE_RIGHT_ARM", "LIFT_RIGHT_ARM", "DOWN_RIGHT_ARM", "RIGHT_ELBOW_UP", "RIGHT_ELBOW_DOWN", "NECK_UP", "NECK_DOWN", "NECK_RIGHT", and "NECK_LEFT".

2) Physics engine for simulation

The simulation is implemented using ODE (Open Dynamics Engine, www.ode.org) [18] which is an open source, high performance library for simulating rigid body dynamics. ODE is fully featured, stable, and mature. It has advanced joint types and integrates collision detection with friction. This is often used for simulating vehicles, objects in virtual reality environments and virtual creatures. Recently, it is being increasingly used for simulation studies on robot learning systems [19].

The robot in ODE will learn to demonstrate a variety of basic actions, each associated with a linguistic label. Firstly, the robot directly acquires the basic actions by observing human demonstrations and learns to categorize and name visual objects. Then the agent learns the basic names corresponding to these primitive actions. Subsequently, it autonomously uses the linguistic symbols that were taught in the previous learning stage to acquire new higher-level, composite actions.

B. Neural network controller

The robot is endowed with a neural controller including three neural feedforward network modules for learning to integrate perceptual, cognitive and motor knowledge. Three training stage are used for the training of this neural controllers: Linguistic-Motor (LM) stage, Linguistic-Visual (LV) stage and Higher-Level action (HL) stage. In Fig.2 (top) we show the neural network controller of the robot. This can be subdivided in three functional modules (Fig.2 bottom) based on different use of input and output units. During the LM module training, the robot’s motor output signal is trained to be associated to linguistic input nodes. In the LV modules training stage, the robot creates object categories which correspond to visual input and linguistic input nodes for recognize object and object naming. After completion of the LM and LV modules training, the robot’s motor control signal will correspond to visual input nodes by higher-level knowledge training (HL).

Each neural module has three layers of units, with connection to and from hidden units modularly organized. The network contains 24 linguistic input units and 28 visual input...
units to control 8 motor output units, and 6 object category output units (Fig. 2 top). The 28 visual input units include 25 (5×5) retina input units, which record object shape information, and 3 input units for color encoding. The 12 hidden units are divided into three groups: two groups of 3 units each, and one group of 6. Two groups specialized for visual category, color and shape. The last group is for motor control. Six object category output units encode the category names. This modular organization of the hidden unit connectivity is needed to allow the compositional learning of higher-order concepts, as in [14].

The networks were trained using the error back-propagation algorithm. The basic grounding consists of two stages (Fig. 2 bottom): \(a\) Linguistic-Motor (LM) learning stage, \(b\)-\(c\) Linguistic-Visual (LV) stage for object name learning and the second for visual category learning.

\[\text{Figure 2. }\] The robot’s neural control system (top) and the three functional modules (bottom) obtained by using different combinations of input/output units. The three modules are: \(a\) LM basic action learning, \(b\) LV object name learning, \(c\) LV visual feature learning.

\[\text{Figure 3. }\] Motor control model and model’s flowchart

2) \(LV\) learning stages

The Linguistic-Visual stage contains three training cycles, one for the object name learning (Fig. 4a), the second for visual category learning (Fig. 4b), and the third when both language and visual input are presented into simultaneously for name-feature learning. During the object name learning stage, the retinal units in the network were set to 0, and only the linguistic input units were activated. The outputs used for the backpropagation training were the object color and shape units. The flowchart shows what happens when the robot hears the object names.

In the object name learning cycle, the linguistic input consists of labels such as “red_box”, and the output selected as teaching input is the red color (out of the three possible colors) and the box shape (out of the three possible shapes).

The other LV training cycles is for the visual feature category learning, where the linguistic units in the network were set to 0, only activating the retinal input. The robot learns to category the object features by identifying the color and shape information from the object visual input through the CCD camera connected to the virtual robot. The CCD camera is used to represent the real scene the robot will see. Therefore, the agent can see the object view and during the image process of color filter and threshold, the neural network learns to categorize object features into specific color and shape categories. The color filter used to process the camera input is YCbCr, in order to filter out red, “NECK_DOWN”. Eight output units in the neural controller represent each of the robot joints: shoulder horizontal, shoulder vertical, elbow, neck horizontal, and neck vertical. This learning stage implements the process of basic symbol grounding, by which the basic actions’ names are directly grounded in the robot’s sensorimotor experience. The flowchart in Fig.3 shows the functional progression of steps in this training stage.

\[\text{Figure 4. }\] Motor control model and model’s flowchart
green and blue three colors with a ranges. YCbCr is one of the color spaces often used on the conversion to separate the color components (e.g. brightness and chrominance) from RGB color models. When the object color is identified, the original picture size is resized to 5×5 a pixel image. At this stage, the object shape information is still retained in it.

To allow the robot to learn the link between object names and visual inputs and the color/shape features in output, a third training cycle is presented when both language and visual input are given in input to the robot’s neural controller (see Fig. 5).

The training stimuli consisted of 30 images each constituted by a 5×5 pixel captured and resized by a camera which set in reality environment. These images contain objects resulting from the combination of 3 different shapes (e.g. square, cross and dots) and 3 other unusual configurations (“non-objects”) (Fig. 6a). Each shapes, square, cross, dots and non-objects have 3 different colors (e.g. red, green and blue), therefore, the total of object is twelve. In the LV training stage, each object will take the image in five different angles (Fig. 6b). For the object naming training cycles, the robot hears the object name through the microphone, sends the linguistic node to neural network and it responds with the output units encoding the category name. In visual object category training cycles, the networks sees the retina image and send the image color and shape information to neural network and responds in object category output units. In this training stage, only 3 objects (red_dot, green square, blue cross) and 3 non-objects (three colors of non-object) with five different views have been trained. The other 6 objects (red_cross, blue_square, green_dot, red_square, green cross and blue_dot) with five different views will be test for robot self-learning acknowledges (Fig. 6a).

3) Higher-Level (HL) learning stage

In order to autonomously learn new behaviors, effective use and integration of basic knowledge to expand of new knowledge and behavior is necessary. The higher level learning stage allows the learner to autonomously acquire higher order, complex actions without the need of a sensorimotor manually setting. This is achieved only through a linguistic instruction strategy. The human only has to give the robot a new linguistic instructions consisting of the names of two basic actions and the name of a new higher lever action. In this study, the robot has to learned four higher-level actions shows in TABLE I. The linguistic instructions are as follow:

• “LIFT_L_ARM+LIFT_R_ARM=LIFT_BOTH_ARM”,
• “DOWN_L_ARM+ DOWN_R_ARM =DOWN_BOTH_ARM”,
• “OPEN_L_ARM+ OPEN_R_ARM =OPEN_BOTH_ARM”,
• “CLOSE_L_ARM+ CLOSE_R_ARM =CLOSE_BOTH_ARM”,

Figure 4. The training stage for combined objects name learning and visual feature learning

Figure 5. Training sequences for LV training stage.

Figure 6. (a) Image source of sample training and testing image, (b) five angle view with each training sample.
A higher lever action (HL1) is based on the combination of two basic actions, and a further higher level concept (HL2) is based on the combinations of one HL1 word and one basic concept. The higher level descriptions above consist of three words respectively naming a new higher order action word and two basic/lower level actions (e.g. $BA^+BA^+$ = HL1, $BA^+HL1^+$ = HL2 and $HL1^++HL1^+$ = HL2, etc.). Grounding transfer takes place from the directly grounded two basic action words to the new higher lever action word. This enables the robot to correctly execute the higher lever actions by combining the actions (e.g. pushing both arms toward the object and grabbing it.)

In Fig. 7 the neural network controller first produces the output corresponding to the input of the first word “CLOSE_LEFT_ARM.” This force is not applied to the joint motors, but is temporarily stored. The same procedure is repeated to the second word “CLOSE_RIGHT_ARM,” when the joints values will be generated and recorded for all action executions for the generation of the teaching input signal from the activation of the next input node. During the second phase, both self-generated output signals will be used as teaching input of the new word “CLOSE_BOTH_ARM.” Subsequently, the input node corresponding to the “CLOSE_BOTH_ARM” action is activated, and the network produces a motor response in the output nodes.

The previous teaching input is now used to compute the error and apply the back-propagation algorithm. These two steps are repeated for each combined action description in training stages HL1 and HL2 (TABLE II ).

The training protocol described above was replicated with three different neural controllers. Each of these was generated with a different set of random weights, all initialized in the range ±1. The training threshold error for the three learning stages Linguistic-Motor (LM), Linguistic-Visual (LV) and Higher-Level learning (HL) was set to 0.05. The parameters of the back-propagation algorithm were as follow: LM stage, learning rate 0.2, LV stage, learning rate 0.1, HL stage, learning rate 0.01. The values produced the best training performance after preliminary tests on different learning rate values. The weights were updated at the end of every training cycle.

### A. ML training result

TABLE III shows the number of training cycles to reach the threshold error of 0.05. This was the approximate minimum number of training cycles necessary to reach a good learning performance. The ML simulation experiments are show in Fig. 8, with the robot demonstrating the basic actions.

<table>
<thead>
<tr>
<th>Basic action training stage</th>
<th>Cycle</th>
<th>Training Time (min)</th>
<th>Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>2901510</td>
<td>41.42</td>
<td>&lt;0.05</td>
<td></td>
</tr>
</tbody>
</table>

### III. RESULT

<table>
<thead>
<tr>
<th>No.</th>
<th>HL1 Combination No.</th>
<th>Combination No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Lift Both Arm</td>
<td>Lift Left Arm + Lift Right Arm</td>
</tr>
<tr>
<td>2</td>
<td>Down Both Arm</td>
<td>Down Left Arm + Down Right Arm</td>
</tr>
<tr>
<td>3</td>
<td>Open Both Arm</td>
<td>Open Left Arm + Open Right Arm</td>
</tr>
<tr>
<td>4</td>
<td>Close Both Arm</td>
<td>Close Left Arm + Close Right Arm</td>
</tr>
<tr>
<td>5</td>
<td>Lift Both Elbow</td>
<td>Left Elbow Up + Right Elbow Up</td>
</tr>
<tr>
<td>6</td>
<td>Down Both Elbow</td>
<td>Left Elbow Down + Right Elbow Down</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>No.</th>
<th>HL2 Combination No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>GRAB Object 4+5</td>
</tr>
<tr>
<td>8</td>
<td>Down Object 2+6</td>
</tr>
</tbody>
</table>

# Figure 7. The Structures for training higher-level action.
Figure 8. The robot’s the basic actions following linguistic input. (first row: INITIAL POSE, NECK_UP and NECK_RIGHT. Second row: OPEN_RIGHT_ARM, LIFT_LEFT_ARM, NECK_DOWN and NECK_RIGHT. Third row: LIFT_RIGHT_ARM, RIGHT_ELBOW_UP, OPEN_LEFT_ARM and LEFT_ELBOW_UP)

B. LV object name learning results

In the LV name learning training stage, the robot was taught three object names: “RED_DOT”, “BLUE_CROSS” and “GREEN_SQUARE”. The training result for these words is shown in TABLE IV. To confirm that the robot can understand the meaning of object names and the associated object features, an object naming test experiment is necessary. In this test, we give the robot 9 object names, including 6 objects never used during the training (i.e., “GREEN_DOT”, “BLUE_DOT”, “RED_CROSS”, “GREEN_CROSS”, “RED_SQUARE” and “BLUE_SQUARE”). Fig. 9 shows the test procedure for the object naming system, with the human user saying the object name to the robot via the microphone, and the robot showing the object which combines the colors and shapes. This experiment was repeated for 100 times, the recognition rate is shown in TABLE V.

TABLE IV. RESULT OF LV VISUAL FEATURE LEARNING

<table>
<thead>
<tr>
<th>Linguistic-Visual training stage</th>
<th>Cycle</th>
<th>Training Time (ms)</th>
<th>Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>5762</td>
<td>1531</td>
<td>&lt;0.05</td>
<td></td>
</tr>
</tbody>
</table>

C. LV visual feature learning result

Three objects and 3 non-objects have been used during the training: red dot, green square and blue cross. To confirm the robot can automatically learn to recognize new objects by using its basic knowledge on the visual features color and shape (e.g. red, green, blue, square, dot and cross) a test was carried out by showing in input 12 different objects. We use CCD cameras to capture objects in real-time, and test the system recognition rate of identification of objects and their name. Three cases of the simulation experiments are show in Fig.10a. There are three windows shows on the robot visual system interface diagram, the upper left windows which marked by a green arrow is the view captured through the CCD camera for real-time, after robot see the object view, the robot will identifies the object through the perception, and the result of the robot perception will show in the upper right windows which marked by a red arrow. But if robot cannot recognize the object, then robot will give the question mark (Fig. 10b). The blue arrow shows the answer. The orange arrow shows the object recognition rate (Fig. 10c). The experiment result is show in TABLE VI.

TABLE V. LV OBJECT NAME LEARNING TEST EXPERIMENT

<table>
<thead>
<tr>
<th>Type</th>
<th>Total</th>
<th>Correct</th>
<th>Wrong</th>
<th>Success rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linguistic naming</td>
<td>100</td>
<td>85</td>
<td>15</td>
<td>85%</td>
</tr>
</tbody>
</table>

Figure 9. Experiment of Object name learning.

Figure 10. Experiment of LV object recognition. (a) Success category and recognize the object, (b) cannot recognize the object but can category the features of colors, (c) Fail to recognize the object and the object features category.
TABLE VI. LV OBJECT CATEGORY EXPERIMENT RESULT

<table>
<thead>
<tr>
<th>Type</th>
<th>Total</th>
<th>Correct</th>
<th>Wrong</th>
<th>Success rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Visual Category</td>
<td>110</td>
<td>109</td>
<td>1</td>
<td>99%</td>
</tr>
</tbody>
</table>

D. HL action training result

The result of the higher-level action training stage is shown in TABLE VII. After linguistic training of HL actions, a “grounding transfer test” is carried out when the robot is requested to demonstrate a new composite action after the input of the higher-order action name. This test is carried out for both HL1 and HL2 complex action categories. Results in TABLE VIII confirm that the robot has acquired an ability to learn complex actions via linguistic combinations.

TABLE VII. RESULT OF HL1 ACTION TRAINING STAGE

<table>
<thead>
<tr>
<th>Cycle</th>
<th>Training Time (min)</th>
<th>Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>240409</td>
<td>34</td>
<td>&lt;0.05</td>
</tr>
</tbody>
</table>

TABLE VIII. RESULT OF HL2 ACTION TRAINING STAGE

<table>
<thead>
<tr>
<th>Cycle</th>
<th>Training Time (min)</th>
<th>Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>603826</td>
<td>91</td>
<td>&lt;0.05</td>
</tr>
</tbody>
</table>

Overall, results indicate that all modules were able to successfully learn the 16 basic actions, 4 higher-level behaviors, 9 objects recognize and category objects features and can corresponding the name of object when robot see the object or heard the object names.

IV. CONCLUSIONS

The model and experiment above concerned the study of symbol grounding and the symbol grounding transfer test in robot self-learning systems. The positive results of the grounding transfer test in the HL condition demonstrate that it is possible to design an autonomous robotic agents with linguistic, vision and sensorimotor modules capable of acquiring new grounded concepts.

This type of research can provide a solid basis upon which future cognitive robotic research and method could be expanded. Future research will look at the integration of HL motor and visual stimuli for complex action sequences involving object use, e.g. with transitive verbs (“I lift red_box”). Moreover, the simulator in this paper is currently being used for preliminary experiments a physical humanoid robot platform. This robot comprise a human-like face, two arms each including a five-finger hand and the platform will develop facial expression and dialogue techniques for the head to interactive with human, and apply a cognitive system used to identify the voice contents, also real-time recognition of objects for effective control two arms and hands for every level action.

Acknowledgment

The authors are grateful to the financial supports from National Science Council of Republic of China under the grant number NSC 99-2915-I-011-007 and on the EU FP7 projects ITALK and Poeticon++.

REFERENCES

[18] ODE (Open Dynamics Engine), www.ode.org