2017 AUPAIR Team Description Paper

Beom-Jin Lee[1] and Jin-Young Choi[2], Chung-Yeon Lee[1], Kyung-Wha Park[2],
Sungjun Choi[1], Christina Baek[3], Byoung-Tak Zhang[123]

[1]Department of Computer Science and Engineering, [2]Cognitive Science Program,
[3]Interdisciplinary Program in Neuroscience,
Seoul National University, Seoul, Korea
{*bjlee*, *jychoi*, *cylee*, *kwpark*, *sjchoi*, *dsbaek*, *btzhang*}@bi.snu.ac.kr
`https://bi.snu.ac.kr/Robocup/index.html`

Abstract. Team AUPAIR aims to develop intelligent mobile cognitive robots with a novel architecture based on machine learning. We envision a new paradigm of autonomous AI with state-of-the-art self-supervised machine learning methods to overcome more restricted paradigms of classical AI which use top-down/rule-driven symbolic and bottom-up/data-driven statistical systems. We propose these autonomous learning algorithms and demonstrate their capability on mobile robot platforms in real home environment settings. With the proposed architecture, we enhance functions of mobile home service robots by developing technologies such as navigation, concept building, object (including person) recognition, schedule learning and HRI. We have released and are preparing more core functions for the @home robots which will be available on our team website http://bi.snu.ac.kr/Robocup/

1 Introduction

It is our belief that fast, flexible, and robust learning in interactive dynamic environments will give rise to a new paradigm of intelligent robots that will enable the next generation of autonomous AI robot. To develop such intelligent mobile home robots, we introduce machine learning algorithms to the learning architecture. This allows the robot to potentially solve problems in a self-supervised manner and thus become more competent of dealing with higher complexity and scalability of learning. Our architecture aims to learn far more complex tasks than those whose implementation through manual programming would be impossible. Also, the learning process should be scalable to real-life environments and work over an extended period of time continuously.

2 Team AUPAIR

The term au pair is originally from the name for a domestic assistant from a foreign country working for and living as part of the host family. With this originality, team AUPAIR aims to build an intelligent mobile home robot that learns the objects, people, actions, events, episodes and schedule plans from daily to extended periods of time as a member of the family. We work towards this goal by developing specific technologies with our machine learning architectures which architecture be applicable to multiple robot platforms. Funded by the Air Force Office of Scientific Research, our team, supervised by Prof. Byoung-Tak Zhang and lead by Beom-Jin focus on the main goals of integrating the multidisciplinary machine learning based applications to the AUPAIR robot.

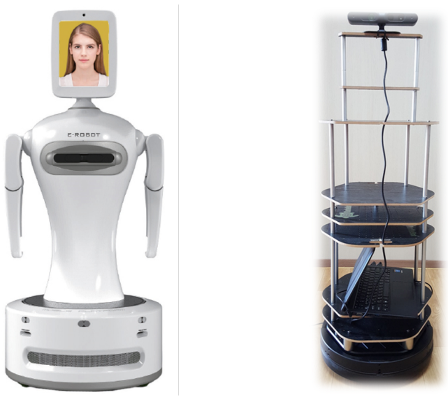

Fig. 1: Temporal robot platform for the TDP. Left: Siltbot, Right: Turtlebot

Fig. 2: Team AUPAIR members. From the left, Christina Baek, Kyung-Wha Park, Beom-Jin Lee, Jin-Young Choi, Chung-Yeon Lee, Sungjun Choi

3 Innovative Technology and Scientific Contribution

3.1 Best View Selection and Positioning

We propose an active viewpoint selecting method [1] based on human body orientation with consideration of the robot's given environment. This makes the robot position itself in front of the person to observe one's behavior. To achieve such a goal, we used convolutional neural networks (CNN) to estimate the person's orientation and specific positioning measures accounting the current position, occupancy of the environment and recognition rate of human-centric services. This ability enables the robot to better observe the human for any human-centric services such as activity and emotion recognition and other HRIs. The procedure for the viewpoint selection starts with the robot following the human, where an RGB image is fed to the CNN [2] to extract the feature and classify the human's orientation. This classified human body orientation (degree) enables the robot to calculate the utility function for the best candidate position for the robot to move to next.

The proposed method performed with a higher and more robust accuracy of body orientation estimation than the classical vision based shallow learning models (HOG+SVM) and showed usefulness in the viewpoint selection and robot positioning.

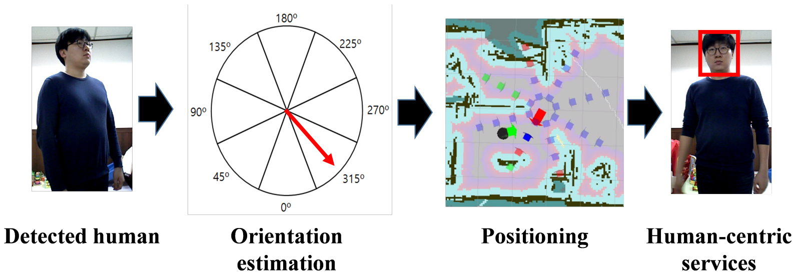

Fig. 3: Best view selection and its positioning system flow

3.2 Online Incremental Concept Building for Visual-linguistic Interaction

Learning mutually-grounded vision-language knowledge is a foundational task for assistive robots. Most knowledge-learning techniques focused on single modal representations in a static environment with a fixed set of data. Here, we explore an ecologically more plausible setting using a stream of cartoon videos to build vision-language concept hierarchies in a continuous manner. We proposed a memory model called the Deep Concept Hierarchy (DCH) model [3] that enables the progressive abstraction of concept knowledge on multiple levels. Moreover,

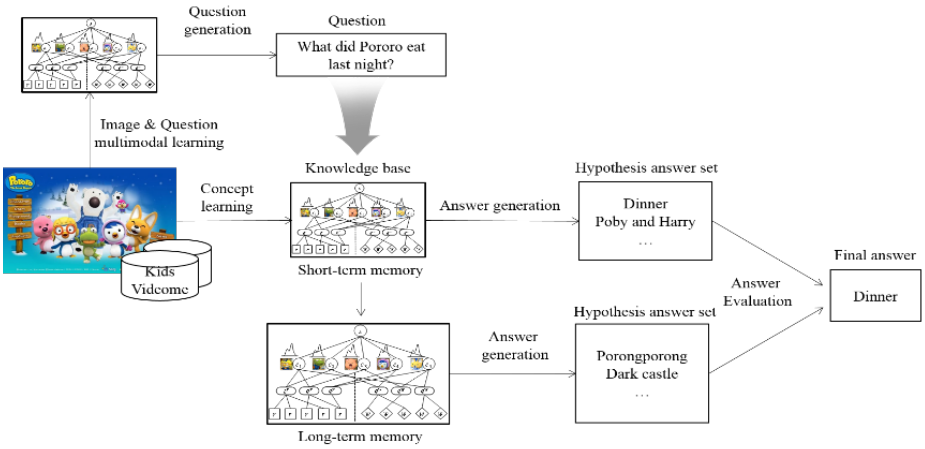

Fig. 4: Overview of video QA system

by employing the model to a mobile robot, it can learn the cartoon video with the child and teach the child what happened in the video. We develop a stochastic method for graph construction, i.e. a graph Monte Carlo algorithm, which enables the graph to maximize the $P(G_t|D)$ from the previous graph G_{t-1} as formulated below,

$$G^* = \arg\max_{G_t} P(G_t|D) = \arg\max_{G_t} P(D|G_t)P(G_{t-1})$$

where $P(D|\theta)$ is the representation of concept using the Spare Population Coding method [4] and G_t is the t^{th} time step of the graph.

This method efficiently searches the vast compositional space of the vision-language concepts thus enabling the DCH model to incrementally build concept hierarchies and handle concept drift which makes the model more suitable for learning in lifelong environments.

Moreover, we have built a video QA system (Fig. 4) that automatically generates questions from the observed video and answers these questions [5].

3.3 Schedule Learning on Robot Platform

A robot needs to understand the environments and ongoing events to operate safely and efficiently in complex real situations such as private homes. To illustrate some of these problems, we built a pseudo-real home environment where family members and a robot can interact with each other based on a daily morning scenario at home.

The robot needs to predict the current and next events to perform appropriate tasks. Event prediction is generally processed based on the environmental understanding. However, we attempted to perform this task in advance using pseudo schedule data, randomly generated by following the morning home scenarios. The schedule data consists of time, location, action, and event labels. To predict an event from the schedule data vectors, we used a Multiplicative-Gaussian Hypernetworks model [6] which is a molecular evolutionary architecture for cognitive learning and memory (Fig. 5). The predicted events, activities, and robot action results are shown on our *website*.

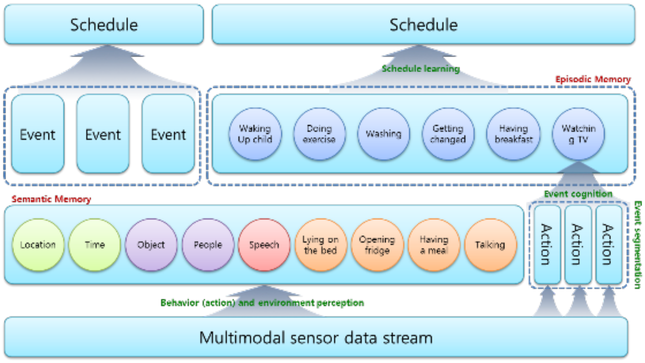

Fig. 5: Schedule learning paradigms

3.4 Distinguishing People in Wild Environment

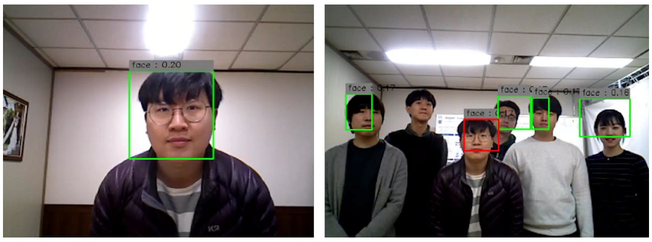

Fig. 6: Left: target person, Right: red box indicating the target found between others

The YOLO [10] module is used to detect human faces in the wild environment. 10 photos of the targeted person to be distinguished is stored in the DB. The face feature extracting model is based on a deep learning network, which was previously trained with the CASIA-WebFace [8]. We extract the feature from the deep network and calculate the Euclidean distance between the peoples' faces detected in the wild environment. Among the tremendous amount of noise present while searching the target person, the most minimal distanced face is selected and finally recognized as the targeted person within an environment containing various people (Fig. 6).

3.5 Visual Conversation Robot for Interactive Engagement

To achieve human-level artificial intelligence, it is crucial to develop algorithms which handle human-like visual and linguistic information. We propose the Multimodal Residual Networks (MRN) for the multimodal residual learning in assumption of visual question-answering tasks [9]. It extends the idea of the deep residual learning, which learns joint representation from vision and language information effectively. The MRN consists of multiple learning blocks, which are stacked for deep residual learning. It uses the optimal mapping function $H(q, v)$ and the joint residual function $F^{(k)}(q, v)$ for the deep residual learning function,

$$H_L(q, v) = W_q, q + \sum_{I=1}^{L} W_{F^{(I)}} F^{(I)}(H_{I-1}, v)$$

Where L is the number of learning blocks. While the MRN is handling with multidisciplinary problems of vision, language and integrated reasoning, a visual conversation robot can be a bridge to interact with humans. Therefore, we propose Cambot which can be instantiated in any platform including robots, desktops and tablet PCs, which have a camera and microphone, engaging natural environmental situations of visual conversation for human interactions.

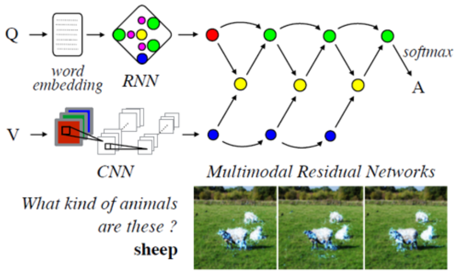
Q
word embedding
RNN
softmax
A
V
CNN
Multimodal Residual Networks
What kind of animals are these ?
sheep

Fig. 7: Inference flow of Multimodal Residual Networks

4 Adopting the Modules to Pepper Robot

We aim our technology for possible application to any platforms using ROS. Every process we built uses ROS topics to communicate between the server and the robot. This means if an anonymous robot is available and using ROS, it is possible to use every module we built by communication via the Internet protocol. Since the Pepper robot is able to publish ROS topics through the Naoqi bridging software, importing the modules to the Pepper robot would simply require changing the topics name of the existing source codes.

4.1 Best View Selection and Positioning

RGB and depth image ROS topics are received from Pepper to the server. The server processes the best view selection and positions as described in the methods section. The resulting information is sent as string topics to Pepper. This allows Pepper to navigate to the best position.

4.2 Online Incremental Concept Building for Visual-linguistic Interaction

The module server receives the RGB image ROS topic from Pepper. As explained in the methods section, it produces a sentence related to the information in the image. The string topic is sent back to Pepper for text-to-speech (TTS).

4.3 Schedule Learning on Robot Platform

The module receives coordination, odometry and an image from Pepper with each ROS topics. Then the server produces the events and predicted coordinates of the robot by running the Multiplicative-Gaussian Hypernetworks model. The topics are integer and float type respectively and sent to the pepper robot.

4.4 Distinguishing People in Wild Environment

As the RGB image ROS topic is sent to the server, it detects the human face and saves the target person image in the DB for subsequent comparison. When the robot is exposed to the wild environment to search for the target person, it receives a stream of RGB image ROS topics and instantly compares the distance between the detected people in the images. Currently, it only sends an integer array to the robot about where the person is, however we will further develop this to provide more information, such as distinguishing genders, number of people etc.

4.5 Visual Conversation Robot for Interactive Engagement

The module server receives an RGB image ROS topic from Pepper and also receives the speech-to-text (STT) from the robot as a string ROS topic. With the information received, the server produces a sentence from the model to correctly answer the STT question. This is then sent back to the robot with TTS by string ROS topic to interact with the human operator.

5 Conclusions and future work

In conclusion, we propose various machine learning architectures for the innovation of mobile home robots in their technologies of navigation, concept building, object recognition and schedule learning. A wider range of related studies can be found on our website, such as dialogue and motion generation. With a suitable robot platform, we anticipate further application of our learning architectures for more technologies in the RoboCup@Home League. We believe with an advanced robot platform, continued progress with our study to design autonomous learning architectures will lead us closer to our ultimate goal of building intelligent mobile cognitive machines that interact and serve humans robustly in home settings.

References

1. Choi, J., Lee, B. J., & Zhang, B. T. (2016). Human Body Orientation Estimation using Convolutional Neural Network. arXiv preprint arXiv:1609.01984.
2. Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems (pp. 1097-1105).
3. Ha, J., Kim, K. M., & Zhang, B. T. (2015, January). Automated Construction of Visual-Linguistic Knowledge via Concept Learning from Cartoon Videos. In AAAI (pp. 522-528).
4. Zhang, B. T., Ha, J., & Kang, M. (2012). Sparse Population Code Models of Word Learning in Concept Drift. In CogSci
5. Kim, K. M., Nan, C. J., Heo, M. O., & Zhang, B. T. (2016). Pororobot: Child Tutoring Robot for English Education. , International Symposium on Perception, Action, and Cognitive Systems (PACS), 2016 (Best Paper Award)
6. Lee, C. Y., Lee, S. W., Lee, C., & Zhang, B. T. (2016). Schedulebot: A Home Robot Learning and Acting Schedule Adaptively via Dynamic Environments. International Symposium on Perception, Action, and Cognitive Systems: Beyond AlphaGo (PACS 2016), pp.69-70, 2016
7. Lee, S. W., Lee, C. Y., Kwak, D. H., Kim, J., Kim, J., & Zhang, B. T. (2016). Dual-memory deep learning architectures for lifelong learning of everyday human behaviors. In Twenty-Fifth International Joint Conference on Artificial Intelligencee (pp. 1669-1675).
8. Yi, D., Lei, Z., Liao, S., & Li, S. Z. (2014). Learning face representation from scratch. arXiv preprint arXiv:1411.7923.
9. Kim, J. H., Lee, S. W., Kwak, D., Heo, M. O., Kim, J., Ha, J. W., & Zhang, B. T. (2016). Multimodal residual learning for visual qa. In Advances in Neural Information Processing Systems (pp. 361-369).
10. Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 779-788).

APENDIX

Team name: AUPAIR
Contact Information: bjlee@bi.snu.ac.kr
Website url: https://bi.snu.ac.kr/Robocup/
Team member name: Beom-Jin Lee, Jin-Young Choi, Chung-Yeon Lee, Kyung-Wha Park, Sungjun Choi, Christina Baek, Byoung-Tak Zhang

Hardware

Although we anticipated working with the Pepper robot at the Robocup@home Social Standard Platform League, as we have been unable to receive the robot on time, we have used alternative platforms, the Turtlebot and Silbot for the purpose of preparing this paper.

-Turtlebot

The turtlebot is a **two wheeled** mobile robot. 1 **Xtion camera sensors**, 1 **RPLIDAR 360 laser sensor**, computing note book is **Asus 1215N laptop** with an Intel Atom D525 dual core processor 1.8GHz, 2GB DDR3 RAM using Ubuntu Linux 14.04 and ROS Indigo.

-Silbot

Intel Core i5 3.4Ghz processor, 4GB Memory using Ubuntu Linux 14.04 and ROS Indigo. 1 **Xtion camera sensor**, 1 **WebCam C525**, 1 maximum resolution of 1280x960 panel, 16 **Ultrasonic Sensor HC-SR04**, **omni-wheeled** robot.

-External Device

For deep learning based real-time applications such as object detection, object recognition, navigation, dialogue modeling and etc, GPU servers are required. We use two servers with the following specifications; Intel i7-6700k processor 3.4 GHz, 32GB RAM, 12GB Pascal GPU and using Ubuntu Linux 14.04 and ROS Indigo. For communicating with the robot, we use ROS topics which the robot publishes all the sensors she have. (explained in section 4).

Software

The open source software have been used with modification to fulfill our optimization to our needs. Tensorflow (https://www.tensorflow.org/), YOLO Object Detection (You Only Look Once: Unified, Real-Time Object Detection) [1] (https://pjreddie.com/darknet/yolo/), pypi speech recognition API (https://pypi.python.org/pypi/SpeechRecognition/), Imagenet Object Recognition (https://www.tensorflow.org/tutorials/image_recognition), ROS Navigation Stack (http://wiki.ros.org/turtlebot_navigation) and our contributed software in our website (https://bi.snu.ac.kr/Robocup/)