ICIC Express Letters
Part B: Applications
Volume **12**, Number **3**, March **2021**

ICIC International ©2021 ISSN 2185-2766
pp. 283–289

STORAGE ARCHITECTURE AND IMPLEMENTATION OF OBJECT RECOGNITION FOR HUMANOID ROBOT

Emny Harna Yossy, Vincent Andreas and Widodo Budiharto*

Computer Science Department
School of Computer Science
Bina Nusantara University
JL. K. H. Syahdan No. 9, Kemanggisan, Palmerah, Jakarta 11480, Indonesia
*Corresponding author: wbudiharto@binus.edu

Received July 2020; accepted October 2020

Abstract. *One of the robot capabilities that are very important and needed is the ability to recognize objects. There have been many methods for detecting objects, whether using deep learning or not, but there is almost no relevant information, about how to store objects or data that is good, and structured, so that the robot can easily access the information in it. In this journal, we propose a model of how to store information with the appropriate architecture.*

Keywords: Humanoid robot, Storage architecture, Object recognition, Object detection

1. **Introduction.** An intelligent humanoid robot that is able to recognize an object near them and give a proper reaction is our hope for the future. A humanoid robot is generally defined as a programmable machine which can imitate the actions as well as the appearance of human [1]. Kehoe et al. [2] do research on cloud-based robot grasping with an object recognition engine. To recognize objects, there are several commonly used methods, from simple to complex ones using deep learning.

Memory is a crucial part to store data in a humanoid robot. The robot must be able to mimic how human memory works. Proper memory management will make a good robot. In the memory management field, the Soar architecture [3] is one of those that tries to reproduce human-like memory management. The GLAIR memory model [4] also has a concept of long term/short term and episodic/semantic memories. Burghart et al. [5] research cognitive architecture for the humanoid robot. The architecture is a mixture of a hierarchical three-layered form on the one hand and a composition of behavior-specific modules on the other hand. Martín et al. propose the building blocks of the architecture of memory for humanoid robot designed as a finite state machine and organized in an ethological inspired way. However, the need of managing explicit symbolic knowledge in human-robot interaction requires the integration of planning capabilities into the architecture and a symbolic representation of the environment and the internal state of the robot [6].

This paper proposes a simple storage architecture in order the robot getting information and knowledge from user. The contribution is an algorithm and architecture of the memory storage for the robot. Part 1 explains about introduction about memory in humanoid robot, Part 2 is related work, Part 3 is our proposed method, Part 4 is our experimental results and finally conclusion is in Part 5.

2. **Related Work.** There are many methods to do object recognition, but we can classify them into two different approaches: a non-deep learning approach, and a deep learning approach.

DOI: 10.24507/icicelb.12.03.283

2.1. **Non-deep learning approach.** Some methods that are still widely used until now, with non-deep learning approach are SIFT and SURF.

2.1.1. *SIFT.* SIFT (Scale Invariant Feature Transform) [7] is used to detect and describe local features in images. According to Shapiro and Stockman [8], the formula of Gaussian is

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{1}$$

x is the distance from the origin in the horizontal axis, y is the distance from the origin in the vertical axis, and σ is the standard deviation of the Gaussian distribution. We need to ensure these features must not be scale-dependent. So, we will search these features on multiples scales [9]. Then, we will do key point selection; for the low contrast, we can use Taylor expansion to compute for each point. If the result magnitude value is less than 0.03, we will reject the key point. To handle the key points that are close to the edge and have a high edge response but may not be robust to a small amount of noise, we can do second-order Hessian matrix to identify such key points. After that, we will do the orientation assignment, so these key points are invariant to rotation. We can do that by creating a histogram for magnitude and orientation (Φ). To calculate the magnitude, we must find the value of gradient x and y (G_x and G_y).

$$\text{Magnitude} = \sqrt{[(G_x)^2 + (G_y)^2]} \tag{2}$$

$$\Phi = \text{atan}(G_y/G_x) \tag{3}$$

In the next step, we will create a histogram for magnitude and orientation. This histogram would peak at some point. The bin at which we see the peak will be the orientation for the key point.

2.1.2. *SURF.* SURF (Speeded-Up Robust Features) [10] is the improvement of the SIFT algorithm. There are several differences between SIFT and SURF.

There is another approach to do detections, like using a sliding window to run the classifier at even spaced location over the image [12].

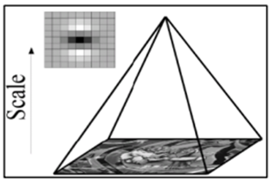
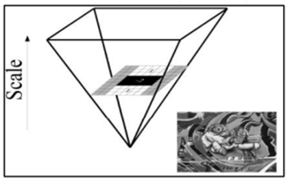
2.2. **Deep learning approach.** Deep learning is a specific subset of machine learning, which is a specific subset of artificial intelligence. Computer vision and natural language processing are a great example of a task that deep learning has transformed into something realistic for robot applications. Using deep learning to classify and label images and text will be better than actual humans. Deep learning methods are proving very good at object recognition, achieving state-of-the-art results on a suite of standard academic benchmark problems. Object recognition that uses deep learning methods, is increasingly developing, along with the higher computer capabilities.

YOLO (You Only Look Once) [13] is a deep learning approach for object detection. YOLO will predict each bounding box object by using a feature from the entire image. First, the image will be divided into $N \times N$ grid. YOLO uses 24 convolutional layers followed by 2 fully connected layers as shown in Figure 1.

YOLO pre-trains the convolutional layers on the ImageNet classification task at half the resolution (224 × 224 input image) and then double the resolution for detection. When detecting an image, YOLO only does one forward propagation pass through the neural network to make predictions [14]. Wang et al. [15] use an adversarial learning strategy to learn invariance in object detection datasets. They propose an adversarial network that generates an example with occlusions and deformation.

3. **Proposed Method.** In our previous work, we successfully proposed the face recognition and speech recognition system using stemming and tokenization for the humanoid

TABLE 1. Comparison between SIFT and SURF [11]

	SIFT	SURF
Scale space	Difference of Gaussian (DoG) is convolved with different size of images with the same size as the filter.  Fix filter is convoluted with down sampling images.	Different size of box filter (Laplacian of Gaussian (LoG)) is convoluted with the integral image.  Fix image is convoluted with upsampling filters.
Key point detection	Using local extrema detection, apply non maxima suppression and eliminate edge response with Hessian matrix.	Determine the key points with Hessian matrix and non maxima suppression.
Orientation	Image gradient magnitude and orientations are sampled around the keypoint location, using the scale of the key point to select the level of Gaussian blur for the image. Orientation of the histogram is used for the same.	A sliding orientation window of size $\pi/3$ detects the dominant orientation of the Gaussian weighted Haar Wavelet responses at every sample point within a circular neighborhood around the interest points.
Size of descriptor	128 bits	64 bits

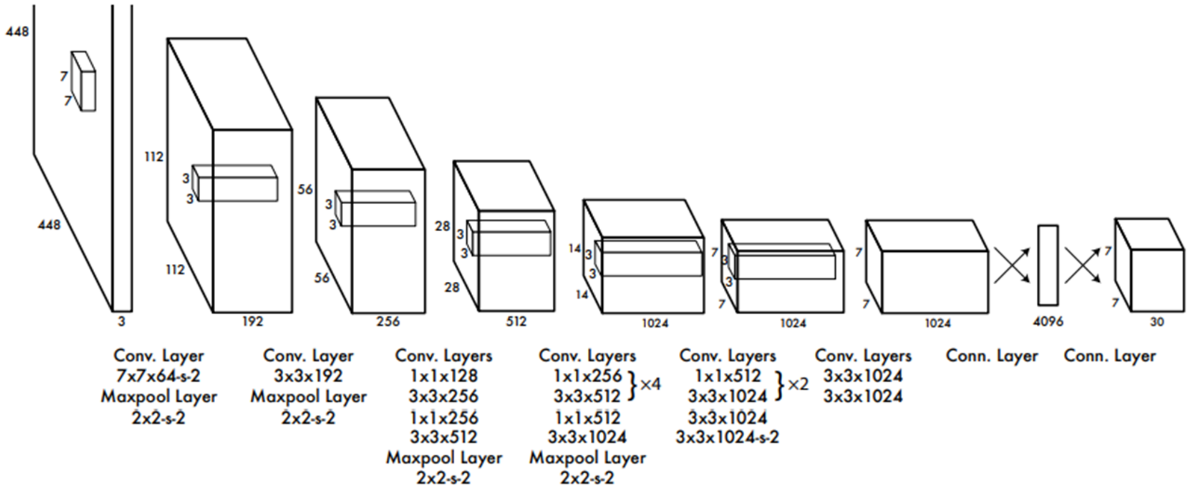


FIGURE 1. The architecture of YOLO [13]

robot for education by using various NLP principles and basic self-learning capability [16-18]. We propose a method of our research as described in Figure 2. This research is commenced from the literature study of object recognition, then deciding the best object recognition method for a humanoid robot, then designing proper database and table for storage architecture, after that testing and evaluation. In the literature study, we find many methods to do object recognition, and we already summarize some methods. After that, we decide on the best object recognition method for a humanoid robot. In this research, we choose YOLO. Then, we try to design the appropriate storage architecture.

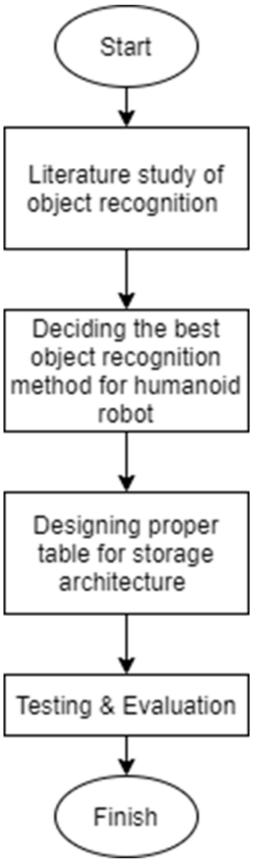

Start
Literature study of object recognition
Deciding the best object recognition method for humanoid robot
Designing proper table for storage architecture
Testing & Evaluation
Finish

Figure 2. Method of our research

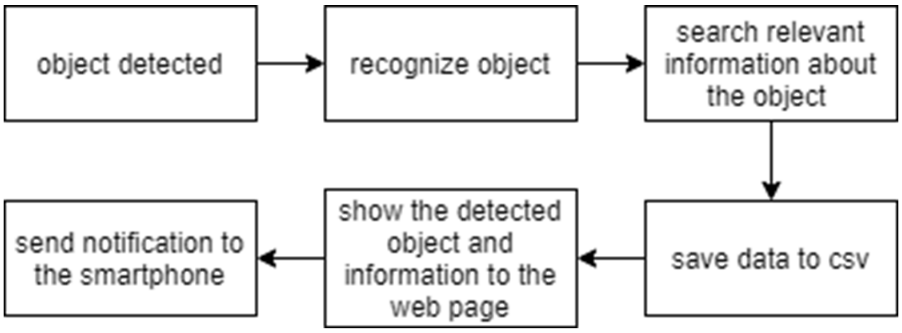

object detected
recognize object
search relevant information about the object
save data to csv
show the detected object and information to the web page
send notification to the smartphone

Figure 3. Diagram of our program

To enhance the ability of a humanoid robot, we add several columns. After designing the table, we try to test and do an evaluation. In this research, we proposed our method for storage architecture, by being stored in a table form in a Comma-Separated Value (CSV) file, which contains the object ID column, object name, object probability, object count, object description, source description, and timestamp.

CSV has advantages in terms of simple use, fast data access, and easy to be transformed into another form; in this case, we want to try to send recognized objects to smartphones using JSON. Previously on smartphones, we had provided an application that would later receive JSON from humanoid robots, to be displayed in the smartphone notification.

When the camera detects pixel changes in front of a humanoid robot, the camera will capture it, and then, the robot will try to recognize the object(s) inside the picture. If the robot successfully recognizes the object(s), the robot will return success status and the object data (object name, total object, object probability). Then the data will be saved in the log file. After that, we will send the data to another platform, such as the smartphone. Below is the snippet of our code.

```
import libraries
open dataset
if __name__ == "__main__":
    is_success, obj_name, obj_prob, obj_count = recognize_obj()
    is_found = False
    obj_data = {}
    if is_success:
        for row in csv_file:
            if obj_name == row[1]:
                is_found = True
                obj_data['objectId'] = row[0]
                obj_data['objectName'] = row[1]
                obj_data['objectDesc'] = row[4]
                obj_data['descSource'] = row[5]
        if is_found == False:
            search_descr(obj_name)
    obj_data['objectProb'] = obj_prob
    obj_data['objectCount'] = obj_count
    obj_data['timestamp'] = dt.datetime.now()
    json_res = json.dumps(obj_data)
    append_to_log_file(json_obj)
    payload = {'json_res': json_res, 'apikey': api_key}
    r = requests.get(url, data = payload)
```

4. **Experimental Results.** We try to do object recognition with several object images. (Bottle, cup, keyboard, phone, dog, cat, tie, suitcase). We will try to examine how accurate the YOLO model recognizes the object, and then how fast it takes time to store, and send data to another platform after the object is already recognized. For this experiment, we use a laptop, with AMD A9 9420, with 8 Gb RAM. The sample of recognized object is shown in Figures 4 and 5.

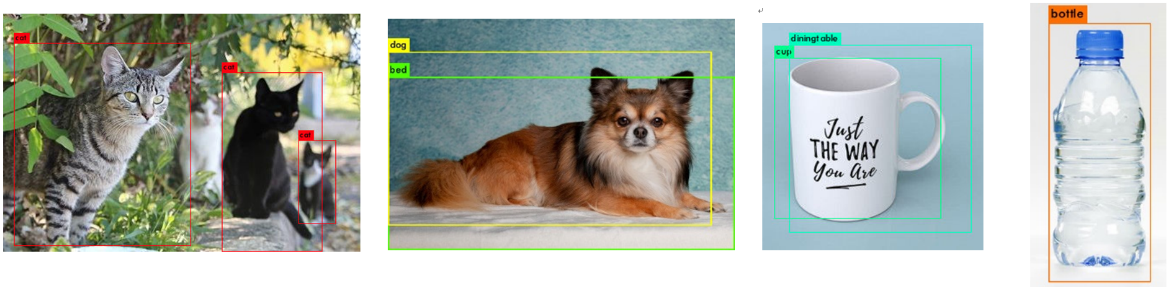
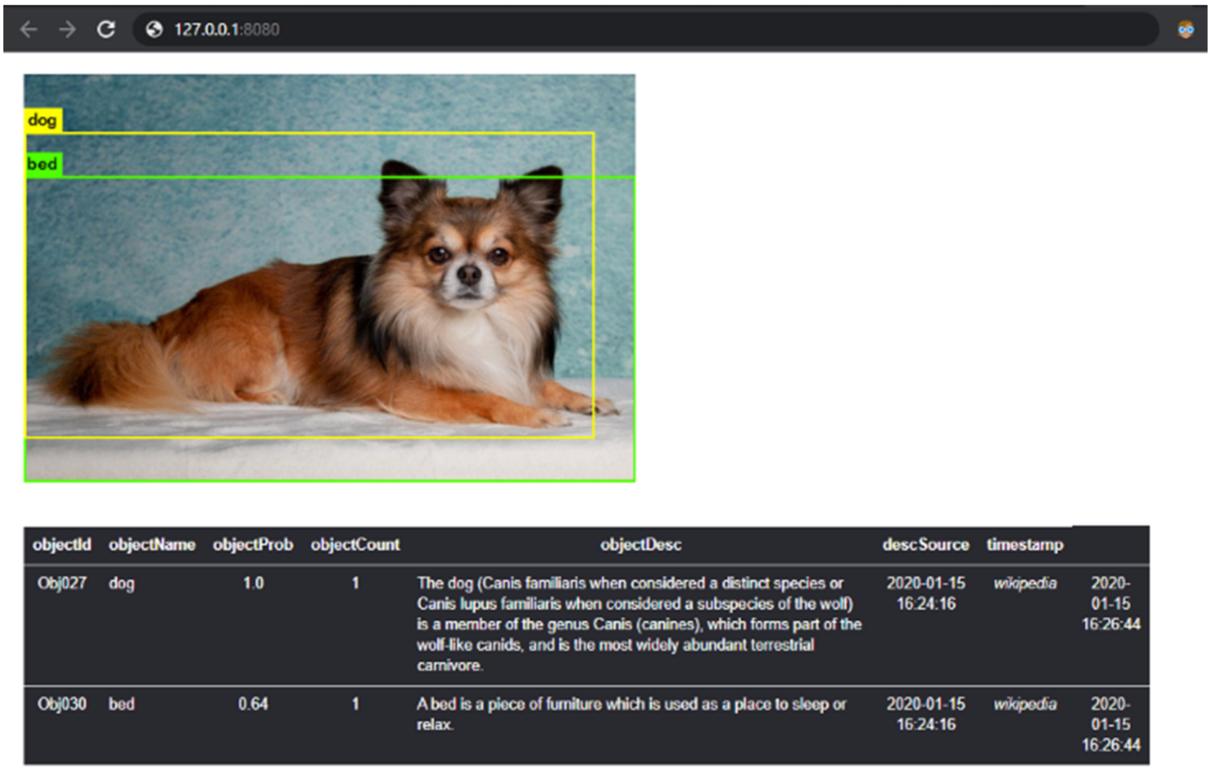


Figure 4. Sample of the recognized object

The result of our experiment is shown in Table 2 with the average time needed for recognizing an object about 22s.

From our experiment, we can conclude that the result of the experiment is quite fast, and we find that from 8 images, we can recognize the image, save the recognized to memory, and send it to the phone with less than one minute. We also improve and implement the object detection for testing 5 DOF arm robot/manipulator for grasping an object.



objectId	objectName	objectProb	objectCount	objectDesc	descSource	timestamp	
Obj027	dog	1.0	1	The dog (Canis familiaris when considered a distinct species or Canis lupus familiaris when considered a subspecies of the wolf) is a member of the genus Canis (canines), which forms part of the wolf-like canids, and is the most widely abundant terrestrial carnivore.	2020-01-15 16:24:16	*wikipedia*	2020-01-15 16:26:44
Obj030	bed	0.64	1	A bed is a piece of furniture which is used as a place to sleep or relax.	2020-01-15 16:24:16	*wikipedia*	2020-01-15 16:26:44

FIGURE 5. Result of object recognition in the web page

TABLE 2. Result of our experiments

Image Id	Object recognized time	Getting Wikipedia description time	Saved to CSV time	Show to web page time	Send notification to phone time	Total time
Img001 (cat)	25s	4s	1s	4s	5s	39s
Img002 (dog)	24s	4s	1s	5s	6s	40s
Img003 (cup)	22s	7s	1.25s	5s	6s	41.25s
Img004 (bottle)	21s	4s	1s	4s	5.5s	35.5s
Img005 (tie)	25s	3s	1s	4.5s	4s	37.5s
Img006 (keyboard)	23s	5s	1s	3s	4s	36s
Img007 (cellphone)	21s	5s	1s	4s	4s	35s
Img008 (suitcase)	22s	4s	1s	4s	5s	36s

5. Conclusion.

The outstanding feature of the humanoid robot is the ability to recognize the object, and interact with that object. To make a faster response and ability, we must have proper memory architecture. In this journal, on the basis of our experiment, we found that it can do faster saving and processing. For future development, we will use the database to save knowledge, so the knowledge can store more data and manage easily. We also will improve our algorithm to make better results to find the answer and improve the face of the robot that is able to show the best emotion. We will add feature of the robot with 5 DOF arm robot (manipulator) for accepting commands from user.

Acknowledgment. This work is supported by Directorate General of Research and Development Strengthening, Indonesian Ministry of Research, Technology, and Higher Education, as a part of Penelitian Hibah Disertasi Doktor titled “Desain Algoritma dan Pemodelan Penyimpanan Representasi Pengetahuan pada Intelligent Humanoid Robot

Berbahasa Indonesia" with SK 26/E1/KPT/2020 and contract number: 183/SP2H/AMD /LT/DRPM/2020, 114/LL3/PG/2020, 060/VR.RTT/VII/2020; and contract date: September 2020.

REFERENCES

[1] V. Graefe and R. Bischoff, Past, present, and future of intelligent robots, *Proc. of the IEEE International Symposium on Computational Intelligence in Robotics and Automation*, Kobe, Japan, pp.801-810, 2003.

[2] B. Kehoe, A. Matsukawa, S. Candido, J. Kuffner and K. Goldberg, Cloud-based robot grasping with the Google object recognition engine, *IEEE International Conference on Robotics and Automation (ICRA)*, Karlsruhe, Germany, 2013.

[3] J. Lehman, J. Laird and P. Rosenbloom, *A Gentle Introduction to Soar, an Architecture for Human Cognition*, Technical Report, University of Michigan, 2006.

[4] S. Shapiro and J. Bona, The GLAIR cognitive architecture, in *AAAI Fall Symposium Series*, 2009.

[5] C. Burghart, R. Mikut, R. Stiefelhagen, T. Asfour, H. Holzapfel, P. Steinhaus and R. Dillmann, A cognitive architecture for a humanoid robot: A first approach, *The 5th IEEE-RAS International Conference on Humanoid Robots*, 2005.

[6] F. Martín, F. J. R. Lera, J. Ginés and V. Matellán, Evolution of a cognitive architecture for social robots: Integrating behaviors and symbolic knowledge, *Applied Science*, vol.10, 2020.

[7] D. G. Lowe, *Object Recognition from Local Scale-Invariant Features*, https://www.cs.ubc.ca/~lowe/papers/iccv99.pdf, 1999, Accessed on 15 April 2020.

[8] L. Shapiro and G. Stockman, *Computer Vision*, Prentice Hall, 2001.

[9] *A Detailed Guide to the Powerful SIFT Technique for Image Matching*, https://www.analyticsvidhya.com/blog/2019/10/detailed-guide-powerful-sift-technique-image-matching-python/, Accessed on 15 April 2020.

[10] H. Bay, A. Ess, T. Tuytelaars and L. V. Gool, Speeded-Up Robust Features (SURF), *Computer Vision, and Image Understanding*, pp.346-359, 2008.

[11] D. Mistry and A. Banerjee, Comparison of feature detection and matching approaches: SIFT and SURF, *GRD Journals – Global Research and Development Journal for Engineering*, vol.2, 2017.

[12] P. F. Felzenszwalb, R. B. Girshick, D. McAllester and D. Ramanan, Object detection with discriminatively trained part-based models, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.32, no.9, pp.1627-1645, 2010.

[13] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, You Only Look Once: Unified, real-time object detection, *arXiv Preprint arXiv:1506.02640v5*, 2016.

[14] J. Redmon and A. Farhadi, YOLOv3: An incremental improvement, *arXiv Preprint arXiv:1804.02767*, 2018.

[15] X. Wang, A. Shrivastava and A. Gupta, A-Fast-RCNN: Hard positive generation via adversary for object detection, *arXiv Preprint arXiv:1704.03414v1*, 2017.

[16] W. Budiharto, A. D. Cahyani, P. C. B. Rumondor and D. Suhartono, EduRobot: Intelligent humanoid robot with natural interaction for education and entertainment, *Procedia Computer Science*, vol.116, pp.564-570, 2017.

[17] W. Budiharto and A. D. Cahyani, Behavior-based humanoid robot for teaching basic mathematics, *Internetwork Indonesia Journal*, vol.9, no.1, pp.33-37, 2017.

[18] W. Budiharto and P. C. B. Rumondor, Modeling of natural interaction and algorithm of intelligent humanoid robot for education, *2017 IEEE International Conference on Mechatronics and Automation (ICMA)*, Takamatsu, Japan, pp.172-176, 2017.