Feature-based Egocentric Grasp Pose Classification for Expanding Human-Object Interactions

Adnan Rachmat Anom Besari[1,2]

[1] Department of Information and Computer Engineering, Politeknik Elektronika Negeri Surabaya, Indonesia.
anom@pens.ac.id, 1904mext@tmu.ac.jp

Azhar Aulia Saputra[2], Wei Hong Chin[2], Naoyuki Kubota[2]

[2] Department of Mechanical Systems Engineering, Faculty of Systems Design, Tokyo Metropolitan University, Japan.
azhar-aulia-saputra@ed.tmu.ac.jp, weihong@tmu.ac.jp, kubota@tmu.ac.jp

Kurnianingsih[3]

[3] Department of Electrical Engineering, Politeknik Negeri Semarang. Semarang, Indonesia.
kurnianingsih@polines.ac.id

***Abstract*— This paper presents a framework for classifying human hand pose, especially in grasping object intuitively. First, we propose a system based on the stereo infra-red image as a sensor that can produce hand coordinates in 3-dimensional space. We use egocentric vision because it can get uniform and natural data with only a single sensor module. Second, we transformed the position to get the angle information for each joint on the finger. Third, we designed an intelligent system based on Multi-Layer Perceptron (MLP) to process angular data to obtain classification results according to the Cutkosky grasp taxonomy. Finally, we compared the results on several similar objects and evaluated their classification accuracy. In the validation phase, the results yielded an accuracy of 16 grasp pose classification is 89,60%. In real-time testing, the results yielded an accuracy of 81.93%. This result shows feature-based learning can reduce the complexity and training time of the MLP. Furthermore, a small amount of training data is sufficient for the training and implementation.**

Keywords—grasp pose classification, human-object interactions, multi-layer perceptron.

I. INTRODUCTION

Research on Human Action Recognition (HAR) is sufficient to recognize human activities from primitive activities to complex activities [1]. Researchers usually use vision-based and body signal-based data. However, HAR is limited to human activities related to movement and gestures that do not involve objects. Until then, researchers began to open the new field of Human-Object Interactions (HOI) detection to recognize human activities together with objects [2]. HOI is a research area that is an extension of object detection involving HAR. However, in its application, HOI faces occlusion problems where it cannot detect the object ultimately. Research in this field also has shortcomings, including being more object-oriented and unable to describe human activities. This problem still becomes a challenge for research of HOI.

In recent years many researchers have begun to be subject-oriented in humans rather than objects [3]. They considered this approach is more comfortable to implement because the object's description from time to time always changes according to product design development, while humans do not change. Moreover, the object specification is less important because some acceptable terms can replace it. For example, weight, large, or thin is relatively more useful to humans than how many millimeters the object is. Meanwhile, how humans interact with objects needs to be known because humans' whole-body have different ways of interacting with many entities [4]. This way is necessary so that the system can provide the most appropriate support to humans.

Humans interact with the world using hands to manipulate objects, machines, tools and socialize with other humans. Currently, we can study human interactions with objects directly by recognizing their grasping behavior [5]. It started with the part of the human body used to show the activity by hand. This approach is needed because grasping is the most basic human interaction with objects. Moreover, the general shape of the current object is not sufficient to detect. We also can get object information according to the geometry of the part held by the hand [6]. For example, opening the nut and bolt using a screwdriver will be easier known through the hand working mechanism, not from the object geometry.

In this study, we are interested in understanding how humans use their hands when carrying out everyday activities, especially in handling objects [7]. We will classify the hand poses on how humans grasp egocentrically based on one of the standard taxonomy models. We use infra-red stereo vision sensors that can take and process hand data into skeletons in real-time. Skeleton data is joint position data in three-dimensional coordinates. We needed the massive data in many variations of poses and possibilities of position. Thus we need to do feature extraction to simplify the process at the next stage [8]. We chose the joint angle because this feature is the easiest to calculate using the joint position. Next, we need to develop a learning system to process the angle features become grasp taxonomy classification in advance.

This paper is structured as follows. Section 2 discusses why the egocentric approach is essential in the grasp pose classification research. Section 3 proposes a methodology in the development of grasp pose classification based on neural networks. Section 4 shows the training results and discusses the effectiveness of the proposed method. Finally, section 5 presents the conclusions and future works of the research.

II. EGOCENTRIC GRASPING

We have encountered much research on hands, especially in gesture classification. Various hand gesture applications require high precision [9], such as communication [10], hand rehabilitation [11], virtual/augmented reality [12], teleoperation[13], and robotic imitation learning [14]. There are several approaches to recognizing hand poses, from muscle signal-based recognition, vision-based recognition, and a combination of the two. Gesture recognition by muscle signal is included in the contact method category because it directly connects with the hand [15]. It was usually using electromyography sensors that are widely available such as Myoarmband® and Bebobsensor®. This sensor takes data from several muscle signals in the arm and then can provide information on hand movements such as first, wave in, wave

2021 IEEE 30th International Symposium on Industrial Electronics (ISIE) | 978-1-7281-9023-5/21/$31.00 ©2021 IEEE | DOI: 10.1109/ISIE45552.2021.9576369

out, open, and pinch. Many researchers usually use this sensor for presentation purposes to mobile robots.

Vision-based recognition is included in the non-contact method because it does not touch hands directly[16]. Image from the camera can be in the 2-dimensional or 3-dimensional format. The 3D image can provide more accurate data and give a hand and each part of the finger possible position in detail. The 3D vision sensor that focuses on hand pose detection is the Leap Motion Controller® (LMC). With the skeleton feature on this device, we can get each joint's coordinates and the hand features. The researcher usually uses this sensor for human-computer interaction and virtual reality.

Researchers divided vision-based hand pose recognition into two parts, namely: hand gesture detection and hand pose estimation. A hand gesture is a symbol of physical behavior or emotional expression and is usually performed by obtaining data directly from the hand image[17]. In comparison, hand pose estimation is the task of finding hand joints from an image. Hand pose estimation is currently in great demand because many researchers predict this approach will give more precision results through hand skeleton position data before determining detection [18]. Meanwhile, recognition based on combining muscle and vision signals [19] or multiplying sensors [20] is rarely developed because it is considered less practical in its application. Researchers who choose to combine the two methods are still optimistic that the results obtained will be better.

The contact method's advantage can be used on a mobile basis, although some people feel uncomfortable using sensors on their hands. Meanwhile, the non-contact method's advantage is that it does not make people use the sensor in their hands because it is placed in one place but cannot be used mobile. For people who want to use the non-contact method but are still mobile, the only option is egocentric vision. Egocentric or first-person vision is a computer vision approach that analyzes the image from a wearable camera [21]. Such cameras are usually worn on the head or the chest and naturally approach the user's visual field. The advantage of egocentric vision is that we can find out a person's attention during activities. Currently, many vendors are producing 3D vision-based glasses for augmented reality applications. Among them is Microsoft HoloLens® or Magic Leap®, which can provide egocentric hand pose data and directly visualize 3D objects on holographic glasses.

After we have chosen egocentric vision as the basis for our approach, we need to determine the classification standard. In the International Classification of Functioning (ICF) on Disability and Health issued by WHO, grasping is an essential part of limb movement activities [22]. WHO included grasping in the "Carrying, Moving and Handling Object" category in the "Hand" Use section, but unfortunately is not described in detail. We need a standard for hierarchical classification of the hand to get a clear structure of the hand and the object. In this study, we chose Cutkosky grasp taxonomy as the first step in data classification.

Mark Cutkosky wrote a paper in 1989 in which he classified a series of manufacturing grasps for evaluating analytical models of grasping and manipulating objects by hand [23]. This taxonomy has been widely used to test hand dexterity. Cutkosky divides the grasp pose into two parts: activities that require power and the other that require precision. Power usually emphasis on security and stability, meanwhile a precision emphasis on dexterity and sensibility. The power section is divided into two parts, namely prehensile, which requires clamping, and non-prehensile, which does not require clamping. Next, the division is carried out hierarchically based on the shape of the object being held, such as prismatic or circular, heavy or light, big or small. Everything is adjusted by the way the hand and each finger touch the object. Fig.1 shows the Cutkosky grasp taxonomy on an egocentric image.

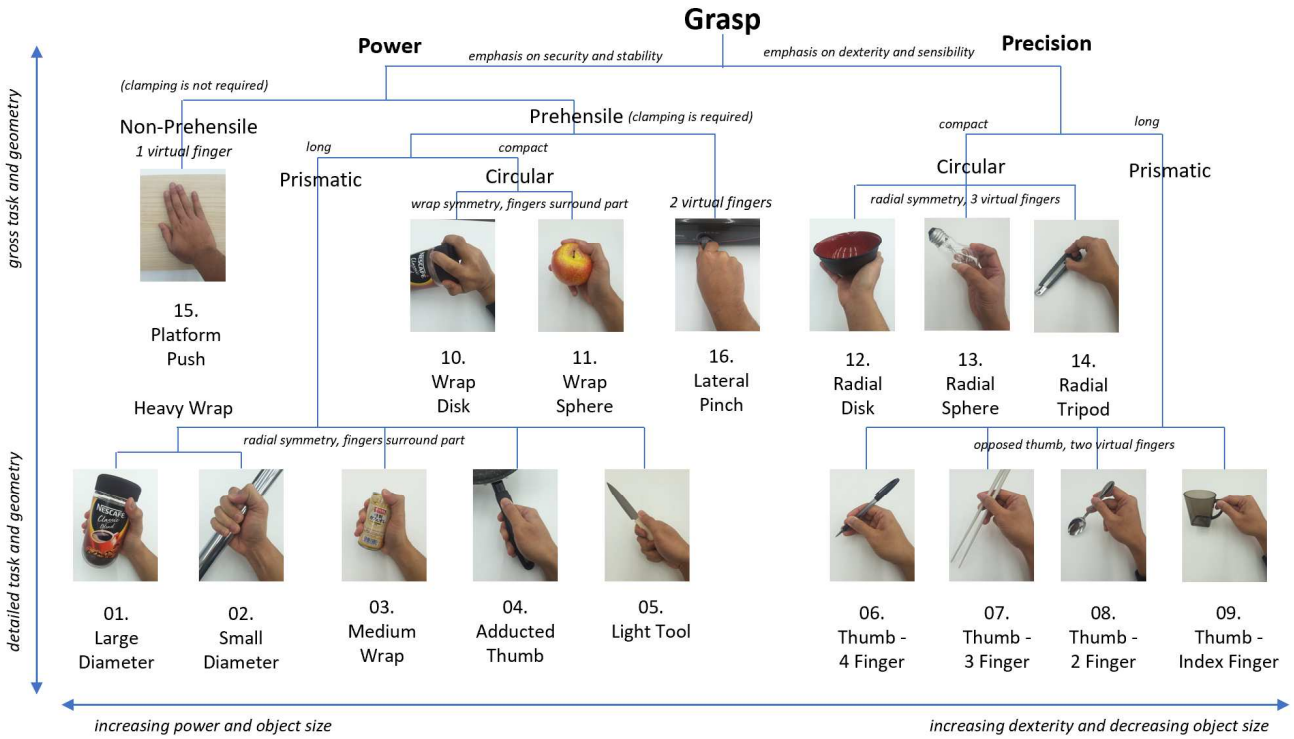

Fig 1. Cutkosky grasp taxonomy on egocentric image.

III. METHODOLOGY

This section will discuss the proposed method used in the egocentric grasp pose classification. The system consists of four parts, namely data acquisition, data transformation, and data training.

A. Data Acquisition

We utilized Ultraleap Stereo Infra-Red 170 (SIR170), the next-generation LMC optical hand tracking sensor, to capture the hand position data in 3D coordinates. The device uses a pair of cameras and an infrared pattern projected by the LEDs to produce an image of the user's hand with depth information. SIR170 has a broader field of view, a more extended tracking range, lower power consumption, and a smaller form factor. It can track hands in a 3D interactive zone extending from 10 cm to 75cm or more, extending from the device in 170x170° (minimum 160x160°) ordinary field of view. Therefore, we categorized SIR170 into optical tracking systems based on stereo vision. We install the sensor on the safety glasses with a 15° angle facing down like in the VR application to get the best results.

Accuracy is one of the essential features of the sensor when measuring human hand poses in 3D objects. The SIR170 has a new motion and position tracking system with an accuracy of up to millimeters. This sensor is available together with an API (Application Programming Interface), which provides positions in the Cartesian space of objects such as fingertips and tooltips. The images obtained by the device are processed on a computer to remove noise and model hands, fingers, tools, and movements. This sensor has a deviation between the desired 3D position and the measured mean position of less than 0.2 mm for the static setting and 1.2 mm for the dynamic environment. The data collection also ignores human hand vibrations that vary in amplitude between 0.4 mm ± 0.2 mm. This device achieves high precision when compared to other gesture-based interfaces (e.g., Microsoft Kinect®). Fig.2 shows the sensor setup with egocentric vision.

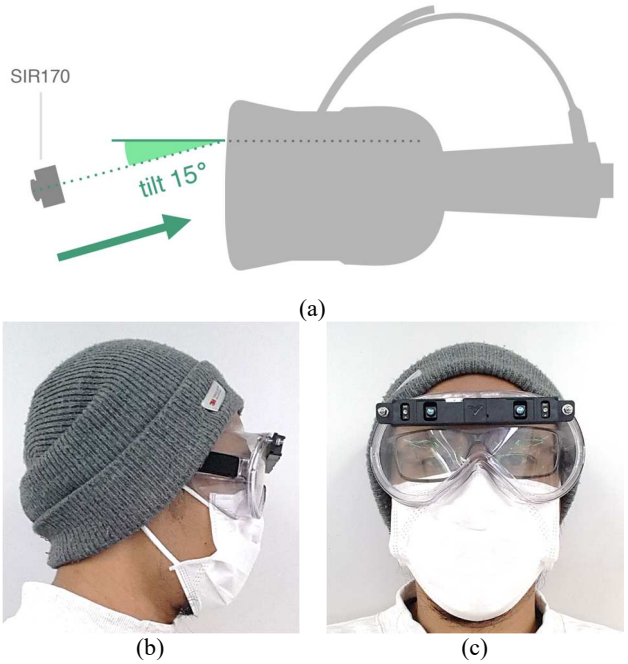

Fig. 2. Sensor setup with egocentric vision : (a) installation of Ultraleap Stereo Infra-Red 170 (SIR170) on glasses; (b) side view; (c) front view.

We connected the SIR170 to a computer using the Unity® application for hand visualization in a 3D environment. We use a computer with an Intel Core i7-10875H CPU @ 2.30GHz (16 CPUs), 16GB RAM, and NVIDIA GEFORCE RTX 2080 (8GB GDDR6 VRAM) specifications. With these specifications, we get an average data resolution of 100 fps. Then we get the data of each hand joint and save it into a CSV file. From this data, we then read and visualize it in 3D coordinates. Fig.3 shows the human hand's taxonomy in skeleton hierarchy, reality, and computer visualization.

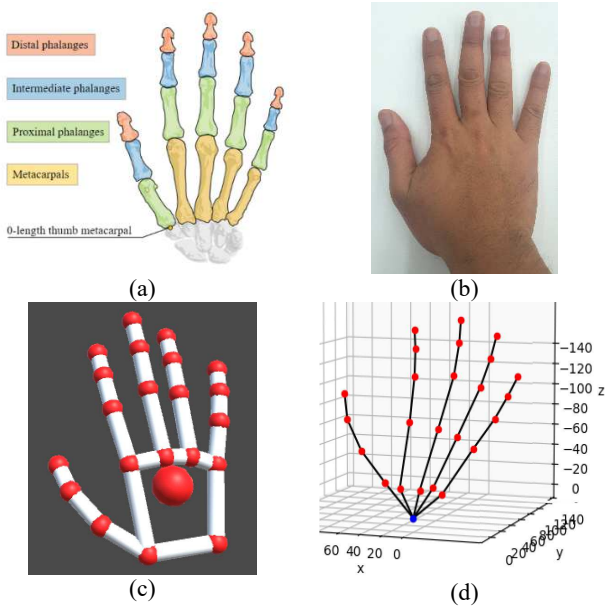

Fig. 3. Taxonomy of human hand : (a) hand skeleton hierarchy; (b) hand image; (c) hand capsule model in Unity®; (d) hand joint 3D coordinate.

B. Data Transformation

Once we have the data, we transform 3D position data from the SIR170 into some features. There are 27 positions of points, consisting of 3 axes (*x, y, z*). Based on Fig.3 (d), the total hand position points are 81 data. We do not use the data directly because we will need many sample data due to the various possible hand poses. This stage will make training data in advance become heavy. The direction towards the hand (Normal Palm) has very many possibilities in the 3D environment. Moreover, not all poses to the hand can be appropriately detected by the sensor because of the occlusion factor and various estimating limitations.

After we get the 3D coordinate for each joint, we will transform them before the training stage [24]. The data obtained is calculated relative to the SIR170 center point, located in the device's center. The first transformation is to move point W (Wrist Position) to point A (0,0,0) as the coordinates at the center of the sensor by making the displacement vector as follows:

$$\overrightarrow{WA} = A - W \tag{1}$$

$$\begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} - \begin{bmatrix} w_x \\ w_y \\ w_z \end{bmatrix} \tag{2}$$

$\overrightarrow{WA}$ is a vector that comes from point W to point A with the value t_x, t_y, t_z . Thus, the points' position on the hand does not come out too far from their center coordinates. Because what we want to recognize is the pose of the hand, not the

movement. The following is an equation for moving the center coordinates of Wrist Position through translation, where x, y, z are the initial coordinates and x', y', z' are the final coordinates.

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (3)$$

To simplify the data, we need another feature placing the position, and that is easiest to obtain is the angle formed by the joints between two bones. To get the angle feature, we perform three points in vector-to-angle transformation. We do this transformation by converting these points in the 3D coordinates into an angle. If we have cartesian coordinates for three points *A*, *B,* and *C,* then we can calculate $\overrightarrow{AB}$ and $\overrightarrow{BC}$. The following are the equations to get two vectors from three points in 3D coordinates:

$$\overrightarrow{AB} = B - A \quad (4)$$

$$\overrightarrow{BC} = C - B \quad (5)$$

Next, we need to find the angle formed by $A \rightarrow B \rightarrow C$ using the *right-hand rule* from *B*. The following is an equation for scalar or dot product :

$$\overrightarrow{AB} \cdot \overrightarrow{BC} = \left\|\overrightarrow{AB}\right\| \left\|\overrightarrow{BC}\right\| cos\, \theta \quad (6)$$

Where $\left\|\overrightarrow{AB}\right\|$ measures the length of $\overrightarrow{AB}$, $\left\|\overrightarrow{BC}\right\|$ measures the length of $\overrightarrow{BC}$. Furthermore, θ (*theta*) is the angle between these two vectors. In this way, we can find the dot product $\overrightarrow{AB} \cdot \overrightarrow{BC}$ and the lengths $\left\|\overrightarrow{AB}\right\|$ and $\left\|\overrightarrow{BC}\right\|$. Finally, we rearrange the formula by substitute the equation. The following is an equation to find θ :

$$\theta = arccos\left(\frac{\overrightarrow{AB} \cdot \overrightarrow{BC}}{\left\|\overrightarrow{AB}\right\| \left\|\overrightarrow{BC}\right\|}\right) \quad (7)$$

A finger has three joints, the first joint connecting the metacarpal and proximal, the second joint connecting the proximal and intermediate phalanges, and the third joint connecting the intermediate phalanges and distal phalanges. Especially for the thumb, it does not have metacarpals. Thus, a hand has 14 joints, as shown in Fig.3(a). Are there any other angle features? If we look at our hand, two fingers that are close together will form an angle. We can determine the angle by connecting the two fingertips' position adjacent to the wrist position, for example, the thumb with the index finger, the index finger with the middle finger, the middle finger with the ring finger, and the ring finger with the little finger. We will get four angles as additional features so that we will have a total of 18 angle features of a hand.

C. Data Training

The data acquisition program has taken grasp poses according to Cutkosky grasp taxonomy. As previously explained, we will predict the classification of 16 grasp poses with supervised learning. The data transformation process has produced 18 angular data, which will be input data in training. We normalize the angular data so that they do not have large deviations. We changed the data so that the distribution would have a mean value of 0 and a standard deviation of 1. We get 130 data for each grasp pose with different directions (Palm Normal). We divided the data into two parts, 100 data for the training process and 30 for the validation process. So that a total of 1600 poses for training data and 480 poses for data validation. The data is saved in tabular format and stored in a CSV file or spreadsheet.

We employed Multi-Layer Perceptron (MLP) as the vanilla architecture of artificial neural networks for the learning system [25]. MLP is a classic type of neural network consisting of one or more layers of neurons (or perceptron). It has the characteristic of a fully connected layer, where this architecture connected each neuron to every other. First, we design the network structure, the number of hidden layers and nodes in each layer. The activation functions for each layer are also are assumed to be known. Weights and biases are the unknown parameters to be estimated. Finding the best MLP network is formulated as a data-fitting problem, and the most well-known is back-propagation algorithms.

Back-propagation is the most widely used method of training the MLP neural network. The system feeds data to the input layer. It is also possible to have a hidden layer that provides a level of abstraction. In the end, the output layer will make predictions according to the learning outcomes. This neural network type is suitable for solving classification prediction problems where the input is assigned a class or label. The total number of parameters in MLP can increase to a very high level. In other words, the number of neurons in the first layer is multiplied by the number of neurons in every next layer. This architecture is inefficient because there is redundancy in such high dimensions. Another disadvantage is that it ignores spatial information. It takes an aligned vector as input. To avoid this problem, we use a lightweight MLP with one hidden layers. Hidden layers with multiple neurons are required to learn non-linear decision boundaries when classifying the output. By learning different functions approximating the output dataset, one hidden layer can reduce the data dimension and identify a complex representation model of the input data. Fig. 4 shows the design of MLP that we employ in the learning system.

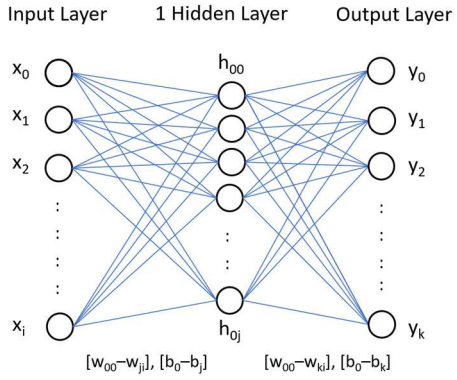

Fig 4. Multilayer perceptron (MLP) with one hidden layer.

We use an input layer containing 18 nodes $(x_0, \ldots, x_{17})$ which features the angles of 14 joints on the fingers and four angles connecting the fingers and wrists' tips. Our model uses one hidden layer, each of which consists some hidden nodes $(h_{00}, \ldots, h_{0j})$. Each node has a weight $(w_{0.0}, \ldots, w_{j.i})$ and bias $(b_0, \ldots, w_j)$ which are always updated during the training and validation phases. While the output layer has 16 nodes

$(y_0, \ldots, y_{16})$ which is a class of grasping poses. We use a training loop that is the same as other models: create an optimizer, feed the inputs to the model, calculate the loss, and use the autograd function to optimize it.

The training data steps as follows: (a) normalize the data structure to be accepted as an MLP input with node and weight properties, (b) applies a linear transformation to the incoming data, (c) uses the Rectifier Linear Unit (ReLU) function element-wire during the learning process. (d) The softmax layer produces the final classification and makes a decision. The task is to predict the input data belongs to which class belongs to which 16 categories of grasp pose. The model produced output for each node in the output layer. From the softmax function, we can get the confidence level of the learning result.

IV. Results and Discussion

In this section, we will present the results of learning using MLP to classify 16 grasp poses. At the training stage, the grasp acquisition program reads data, enters it as MLP input, and carries out the learning process. As the number of data is categorized as small, the number of epochs selected is 100. The loss decreases with the rising number of epochs. After the 100th epoch, a slight loos reduction occurred, and the loss value is less than 0.1. The training process is implemented with 100 data in each class. The experiment showed that the MLP learning method succeeded in a supervised classification when 100 epochs were completed. Fig. 5 shows the decrement of MLP training loss until 100 generations for the supervised classification of 16 grasp pose.

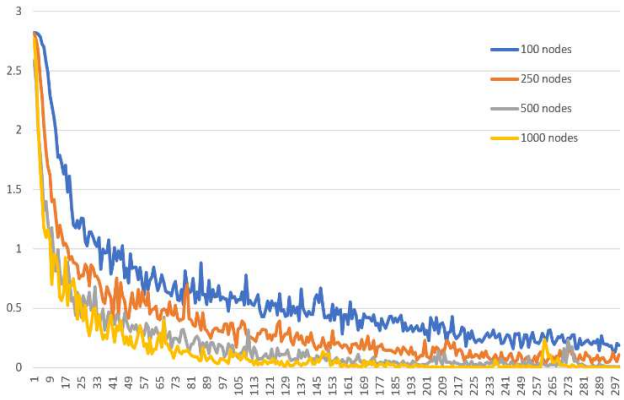

Fig 5. Comparison of the loss of training the MLP with one hidden layers in different node sizes

Next, the performance of the classification has been evaluated at the validation stage. We have performed a validation process on 30 data in each class. The program enters each validation data into the trained model. Then the output of the model is compared with the label in the dataset to get the accuracy. Accuracy is one of the metrics for evaluating classification models. Informally, accuracy is a fraction of the predictions our models make correctly. Accuracy is measured with the following definition: dividing the number of correct predictions by the total number of predictions. The accuracy result from the MLP network above are 83.33% (100 nodes), 84.31% (250 nodes), 86.27% (500 nodes) and 89.60% (1000 nodes). If the number of nodes is enlarged, the accuracy tends to decrease. This happening indicates the overfitting of the model.

This accuracy may seem reasonable at first glance, but we do not know how accurate the model has been training each hand pose. We need to find the success rate of its classification by testing the model using different datasets. The MLP structure has completed training and validation in the testing phase and will be tested with new data. We have tried the model obtained from the learning outcomes to recognize the type of grasp pose in some daily activities. Table 1 shows the result of grasp pose classification using a trained model.

TABLE I. The Results of Real-Time Classification.

No.	Class Name	Illustration	Requirement	Accuracy
1	Large Diameter		Power	82%
2	Small Diameter		Power	81%
3	Medium Wrap		Power	77%
4	Adducted Tumb		Power	82%
5	Light Tool		Power	79%
6	Tumb-4 Finger		Precision	86%
7	Tumb-3 Finger		Precision	81%
8	Tumb-2 Finger		Precision	61%
9	Tumb-Index Finger		Precision	83%
10	Disk-wrap		Power	82%
11	Sphere-wrap		Power	90%
12	Disk-radial		Precision	71%
13	Sphere-radial		Precision	92%
14	Tripod-radial		Precision	87%
15	Platform Push		Precision	88%
16	Lateral Pinch		Precision	89%
	Average			**81.93%**

The results above indicate that the average accuracy obtained from testing is lower than the validation results. However, we can see that the results for each class above differ from one another. Some are quite precise, that some are less precise. This result shows that the grasp poses dataset obtained has several similarities. For example, large, medium, and small diameters have similarities, depending on human inference. Other examples are disk-radial and disk-wrap or sphere-radial and sphere-wrap, having very similar taxonomies. It would be tough to distinguish taxonomies if only looking at the taxonomy grasp alone.

Accuracy does not tell the full story when we are working with a class-imbalanced grasp pose dataset. For example, to distinguish similar grasp poses, we need to know how to choose whether the grasp requires power or precision. These problems have some significant disparity between the

number of identical grasp poses. In future works, the learning system needs additional data for MLP to get better results. There needs to be an approach to improve the accuracy of some grasp poses to be recognized in detail. One method proposed is the contact method, which is knowing the arm muscle signal during grasp activity. Furthermore, this study will face constraints regarding the occlusions and errors sensor readings when data collection. It will be the primary problem for developing a robust system in grasp pose application in the future.

V. Conclusions

This paper discussed grasp pose classification based on angle features. We utilize egocentric vision with only a single sensor module to obtain a uniform and natural skeletal model. The system obtained the data by observing the grasp pose when the hand interacts with the object. We taken the data simultaneously, transformed it into angle features, and then represented it in the 3D environment. We use Multi-Layer Perceptron (MLP) with one hidden layers for supervised classification for the learning system. Results showed the accuracy for the 16 grasps pose is 89.60%. The proposed method was further evaluated with daily life grasp pose datasets. The average grasp pose classification accuracy in real-time is 81.93%. Real-time testing for grasp pose with many possibilities is needed to be supported by distributed personal datasets for real-world applications. For future works, we will apply the proposed system to accurately and efficiently recognize humans-object interactions

Acknowledgment

The authors would like to thank the Japan Ministry of Education, Culture, Sports, Science, and Technology (MEXT) for providing financial support and opportunities to become a doctoral student in Kubota Laboratory, Tokyo Metropolitan University (TMU). This work was (partially) supported by JST [Moonshot R&D][Grant Number JPMJMS2034].

References

[1] Z. Haibin and N. Kubota, "Method on Human Activity Recognition Based on Convolutional Neural Network," in *Intelligent Robotics and Applications*, vol. 11742, H. Yu, J. Liu, L. Liu, Z. Ju, Y. Liu, and D. Zhou, Eds. Cham: Springer International Publishing, 2019, pp. 63–71.

[2] T. Bergstrom and H. Shi, "Human-Object Interaction Detection: A Quick Survey and Examination of Methods," in *Proceedings of the 1st International Workshop on Human-centric Multimedia Analysis*, Seattle WA USA, Oct. 2020, pp. 63–71, doi: 10.1145/3422852.3423481.

[3] A. R. A. Besari, W. H. Chin, N. Kubota, and Kurnianingsih, "Ecological Approach for Object Relationship Extraction in Elderly Care Robot," in *2020 21st International Conference on Research and Education in Mechatronics (REM)*, Dec. 2020, pp. 1–6, doi: 10.1109/REM49740.2020.9313877.

[4] O. Taheri, N. Ghorbani, M. J. Black, and D. Tzionas, "GRAB: A Dataset of Whole-Body Human Grasping of Objects," in *Computer Vision – ECCV 2020*, vol. 12349, A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, Eds. Cham: Springer International Publishing, 2020, pp. 581–600.

[5] J. Liu, F. Feng, Y. C. Nakamura, and N. S. Pollard, "A taxonomy of everyday grasps in action," in *2014 IEEE-RAS International Conference on Humanoid Robots*, Madrid, Spain, Nov. 2014, pp. 573–580, doi: 10.1109/HUMANOIDS.2014.7041420.

[6] P. Song, Z. Fu, and L. Liu, "Grasp planning via hand-object geometric fitting," *Vis. Comput.*, vol. 34, no. 2, pp. 257–270, Feb. 2018, doi: 10.1007/s00371-016-1333-x.

[7] A. R. A. Besari *et al.*, "Preliminary design of interactive visual mobile programming on educational robot ADROIT V1," *2016 International Electronics Symposium (IES)*, Sep. 2016, pp. 499–503, doi: 10.1109/ELECSYM.2016.7861057.

[8] P. B. Shull, S. Jiang, Y. Zhu, and X. Zhu, "Hand Gesture Recognition and Finger Angle Estimation via Wrist-Worn Modified Barometric Pressure Sensing," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 27, no. 4, pp. 724–732, Apr. 2019, doi: 10.1109/TNSRE.2019.2905658.

[9] M. Yasen and S. Jusoh, "A systematic review on hand gesture recognition techniques, challenges and applications," *PeerJ Comput. Sci.*, vol. 5, p. e218, Sep. 2019, doi: 10.7717/peerj-cs.218.

[10] W. B. Dou, W. H. Chin, and N. Kubota, "Hand Gesture Communication using Deep Learning based on Relevance Theory," in *2020 Joint 11th International Conference on Soft Computing and Intelligent Systems and 21st International Symposium on Advanced Intelligent Systems (SCIS-ISIS)*, Dec. 2020, pp. 1–5, doi: 10.1109/SCISISIS50064.2020.9322784.

[11] T. Obo, C. K. Loo, M. Seera, T. Takeda, and N. Kubota, "Arm motion analysis using genetic algorithm for rehabilitation and healthcare," *Appl. Soft Comput.*, vol. 52, pp. 81–92, Mar. 2017, doi: 10.1016/j.asoc.2016.12.025.

[12] Y. Li, J. Huang, F. Tian, H.-A. Wang, and G.-Z. Dai, "Gesture interaction in virtual reality," *Virtual Real. Intell. Hardw.*, vol. 1, no. 1, pp. 84–112, Feb. 2019, doi: 10.3724/SP.J.2096-5796.2018.0006.

[13] W. Zhang, H. Cheng, L. Zhao, L. Hao, M. Tao, and C. Xiang, "A Gesture-Based Teleoperation System for Compliant Robot Motion," *Appl. Sci.*, vol. 9, no. 24, p. 5290, Dec. 2019, doi: 10.3390/app9245290.

[14] T. Obo and N. Kubota, "Imitative motion generation for smart device interlocked robot partner based on neuro-genetic approach," in *2016 55th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE)*, Tsukuba, Japan, Sep. 2016, pp. 1179–1184, doi: 10.1109/SICE.2016.7749258.

[15] F. Stival, S. Michieletto, M. Cognolato, E. Pagello, H. Müller, and M. Atzori, "A quantitative taxonomy of human hand grasps," *J. NeuroEngineering Rehabil.*, vol. 16, no. 1, p. 28, Dec. 2019, doi: 10.1186/s12984-019-0488-x.

[16] Y. Muratov, M. Nikiforov, and O. Melnik, "Hand Gesture Recognition for Non-Contact Control of a Technical System," in *2020 International Russian Automation Conference (RusAutoCon)*, Sochi, Russia, Sep. 2020, pp. 1107–1111, doi: 10.1109/RusAutoCon49822.2020.9208182.

[17] T. Vuletic, A. Duffy, L. Hay, C. McTeague, G. Campbell, and M. Grealy, "Systematic literature review of hand gestures used in human computer interaction interfaces," *Int. J. Hum.-Comput. Stud.*, vol. 129, pp. 74–94, Sep. 2019, doi: 10.1016/j.ijhcs.2019.03.011.

[18] A. Vysocký *et al.*, "Analysis of Precision and Stability of Hand Tracking with Leap Motion Sensor," *Sensors*, vol. 20, no. 15, p. 4088, Jul. 2020, doi: 10.3390/s20154088.

[19] G. A. G. Ricardez *et al.*, "Wearable Device to Record Hand Motions based on EMG and Visual Information," in *2018 14th IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications (MESA)*, Oulu, Jul. 2018, pp. 1–6, doi: 10.1109/MESA.2018.8449178.

[20] V. Kiselev, M. Khlamov, and K. Chuvilin, "Hand Gesture Recognition with Multiple Leap Motion Devices," in *2019 24th Conference of Open Innovations Association (FRUCT)*, Moscow, Russia, Apr. 2019, pp. 163–169, doi: 10.23919/FRUCT.2019.8711887.

[21] A. Bandini and J. Zariffa, "Analysis of the hands in egocentric vision: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1–1, 2020, doi: 10.1109/TPAMI.2020.2986648.

[22] Weltgesundheitsorganisation, Ed., *International classification of functioning, disability and health: ICF*. Geneva: World Health Organization, 2001.

[23] M. R. Cutkosky, "On Grasp Choice, Grasp Models, and the Design of Hands for Manufacturing Tasks," p. 12.

[24] J. E. Gentle, *Matrix Algebra: Theory, Computations, and Applications in Statistics*. Springer Science & Business Media, 2007.

[25] L. C. Vu and B.-J. You, "Hand Pose Detection in HMD Environments by Sensor Fusion using Multi-layer Perceptron," in *2019 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*, Okinawa, Japan, Feb. 2019, pp. 218–223, doi: 10.1109/ICAIIC.2019.8669023.