IIEEJ Transactions on Image Electronics and Visual Computing

Special Issue on Journal Track Papers in IEVC2024

Vol. 12, No. 1 2024

The Institute of Image Electronics Engineers of Japan

Editorial Committee of IIEEJ

Editor in Chief

Vice Editors in Chief

Advisory Board

Editors

Reviewer

IIEEJ Office

Contact Information

The Institute of Image Electronics Engineers of Japan (IIEEJ）
3-35-4-101, Arakawa, Arakawa-ku, Tokyo 116-0002, Japan
Tel : +81-3-5615-2893　Fax : +81-3-5615-2894
E-mail : hensyu@iieej.org
http://www.iieej.org/ (in Japanese)
http://www.iieej.org/en/ (in English)
http://www.facebook.com/IIEEJ (in Japanese)
http://www.facebook.com/IIEEJ.E (in English)

IIEEJ Transactions on
Image Electronics and Visual Computing
Vol.12 No.1　June 2024

CONTENTS

Special Issue on Journal Track Papers in IEVC2024

Announcements

Guide for Authors

Published two times a year by the Institute of Image Electronics Engineers of Japan（IIEEJ）
3-35-4-101，Arakawa，Arakawa-ku，Tokyo 116-0002，Japan
Tel：+81-5615-2893　Fax：+81-5615-2894　E-mail：hensyu@iieej.org　http://www.iieej.org/

Upon the Special Issue on Journal Track Papers in IEVC2024

Editor: Osamu UCHIDA
(Tokai University)

The 8th International Conference on Image Electronics and Visual Computing (IEVC2024) was held in Tainan City, Taiwan, on March 11-14, 2024, as the international academic event of Image Electronics Engineers of Japan (IIEEJ). It was based on the great success of the previous seven workshops/conferences in 2007 (Cairns, Australia), 2010 (Nice, France), 2012 (Kuching, Malaysia), 2014 (Koh Samui, Thailand), 2017 (Da Nang, Vietnam), 2019 (Bali, Indonesia), and 2021(Online). The conference aims to bring together researchers, engineers, developers, and students from various fields in academia and industry to discuss the latest studies, standards, developments, implementations, and application systems in all image electronics and visual computing areas.

As in previous IEVCs, IEVC2024 has two paper categories: general paper and late-breaking paper (LBP). Also, as in the past two IEVCs, the general paper category has two tracks: Journal track (JT) and Conference track (CT). The journal track offers authors the advantage of having their papers publish in the special issue on Journal Track Papers in IEVC2024, scheduled for the June 2024 issue of IIEEJ Transactions on Image Electronics and Visual Computing.

This special issue contains 6 papers that were accepted within the publication schedule of the June 2024 issue, and the second special issue on JT papers in IEVC2024 is also planned for the coming issues. All the papers are of high-level content and will contribute to further developing the field of image electronics and visual computing.

Before concluding, I would like to express my heartfelt appreciation to all the reviewers and editors. Your dedication and expertise have been instrumental in improving the quality of the papers. I would also like to extend my deepest gratitude to the members of the editorial committee of IIEEJ and the staff at the IIEEJ office for their invaluable support.

IIEEJ Paper -- Special Issue on Journal Track Papers in IEVC2024 --

Classification of White Blood Cells Using YOLOv7: Single and Cascade Classification Approaches Applied to Images Segmented by CellaVision^{TM} DM96

Thalita Munique COSTA[†], Yoko USAMI[††], Mai IWAYA[††], Yuka TAKEZAWA[††], Yuika NATORI[††],
Hernan AGUIRRE[‡], Kiyoshi TANAKA[‡] *(Honorary Member)*

[†] Graduate School of Medicine, Science and Technology, Shinshu University, Japan,
[††] Department of Laboratory Medicine, Shinshu University Hospital, Japan,
[‡] Academic Assembly (Institute of Engineering), Shinshu University, Japan

<Summary> A common task executed in the medical routine is the identification, classification, quantification, and analysis of white blood cells from peripheral blood, which is commonly done with the help of automatic counters. Some of the most popular machines present low accuracy and commit relevant mistakes in the classification of the cells. In this work, we propose and discuss the use of the deep learning architecture YOLOv7 in the reclassification of blood cell images segmented by the machine CellaVision TM DM96 into 11 classes, i.e., Band Neutrophil, Segmented Neutrophil, Basophil, Eosinophil, Erythroblast, Thrombocyte, Lymphocyte, Lymphocyte Variant, Metamyelocyte, Monocyte, and Myelocyte, in single and cascade classification methods. The classification made by CellaVision TM DM96 achieved an accuracy of 76.20%, precision of 80.93%, recall of 92.87%, and F1-Score of 86.49%. The single classification method presented a mean accuracy of 93.59%, precision of 96.16%, recall of 97.23%, and F1-Score of 96.69%. The Cascade method resulted in a mean accuracy of 93.85%, precision mean of 96.81%, a recall of 97.23%, and F1-Score of 96.83% for the same evaluation database. Both methods proved effective in increasing the performance in blood images classification and, mainly the cascade method, reduced the rate of more relevant mistakes.

Keywords: digital pathology, white blood cells, YOLOv7, deep learning

1. Introduction

In clinical laboratories, the identification and classification of blood cells, mainly white blood cells (i.e., WBCs) is a common task. The number of WBCs in a sample, the quantity of each type of WBC, and the identification of abnormal cells are important information used in the diagnosis [1]. WBCs are the group of nucleated blood cells that play the most significant role in the defense of the human body against diseases, acting on immunity and eliminating viruses, bacteria, parasites, and fungi[2].

There are three main types of WBCs: monocytes, lymphocytes, and granulocytes. Granulocytes are further divided into neutrophils, eosinophils, and basophils. Neutrophils can be classified, based on maturity level, into band neutrophils, segmented neutrophils, myelocytes, and metamyelocytes. Metamyelocytes and myelocytes are rarely seen in the bloodstream but can be found in cases of severe infections. WBC disorders can be quantitative or qualitative. In qualitative defects, abnormal cells are present in circulation. Variant lymphocytes (i.e., variant) are abnormal lymphocyte cells[1].

Autoimmune diseases and immune deficiencies are examples of diseases diagnosed by the WBC count[2], but each pathological agent modifies the number of each WBC type differently. Because of that, counting the number of cells of each type and identifying abnormal cells is also crucial for diagnosis. Metabolic disorders, bacteria, and fungi are examples that increase the number of neutrophils. Hepatitis, viruses, and leukemia are situations that increase the number of lymphocytes while HIV rubeola, and chickenpox reduce the quantity. Listeriosis, malaria, bacterial and viral infection can change the number of monocytes. Allergic diseases and parasites are examples that change the number of eosinophils and malignant myeloproliferative diseases, and inflammation can change the number of basophils found in the bloodstream[2].

Because of their shape, some other cells are commonly wrongly classified as WBCs by automated counters. Erythroblasts and Giant Thrombocytes are examples of these cells. Even though they are not considered WBCs, the analysis of thrombocytes and erythroblasts is also a step performed in the diagnosis of diseases.

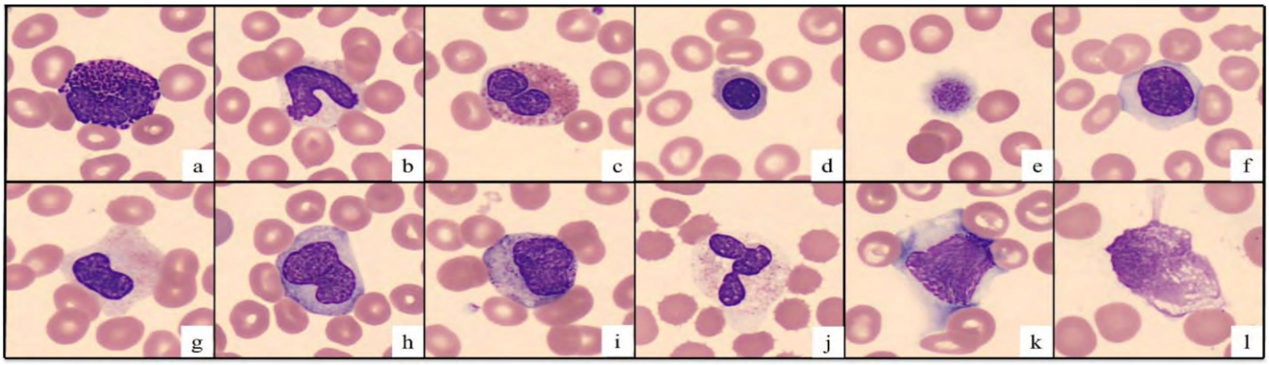


Fig. 1 Cell images obtained from CellaVision: a) Basophil, b) Band Neutrophil, c) Eosinophil, d) Erythroblast e) Thrombocyte, f) Lymphocyte, g) Metamyelocyte, h) Monocyte, i) Myelocyte, j) Segmented Neutrophil, k) Variant Lymphocyte, l) Image outside the proposed classification

The human classification of WBCs is a task that consumes a lot of time and energy for blood-specialized professionals and, even a very well-trained professional, can commit mistakes in a very busy and tiring routine. For these reasons, automated counters are widely used for the identification and classification of WBCs since they bring higher accuracy, consistency in results, and speed over manual techniques but, due to the biologically complex nature of cells, there isn't a single optimal method supporting the diagnosis in the laboratories[3].

CellaVision DM96^{TM} [4] (i.e., CellaVision) is an automated image analysis system for peripheral blood smears, widely used in laboratories, which scan glass slides, identifies potential WBCs, take digital images with high magnification, and, using a neural network algorithm, outputs images of the segmented WBCs and a class label on a customizable computer display for confirmation or reclassification. **Figure 1** shows images obtained by CellaVision. CellaVision can identify not just healthy WBCs but also abnormal WBCs and nucleated Red Blood Cells[5]. As also happens in other automated counters, the classes attributed to the cells by this system are often wrong. The machine offers important support in the blood analysis, segmenting each WBC in a different image, but, due to the high error rate in the classification, blood specialists must reassess all the segmented images and perform the reclassification of these cells, which is a heavy burden in medical routine. To solve this problem, deep learning algorithms can be a useful tool in the reclassification of these images with a higher accuracy so that manual reclassification is no longer necessary.

Neural Networks and Deep Learning merged trying to simulate the human brain neural network process of learning and memorizing. Deep Learning algorithms consist of a multi-layered neural network architecture and, after being trained with a large amount of data, can perform identifications and classifications traditionally performed manually[6]. YOLOv7[7] is a deep learning-based object detection model that stands out, among other reasons, for its accuracy and detection speed.

This work studies the use of the deep learning architecture YOLOv7 as a tool to support the classification of WBC images segmented by CellaVision to increase the accuracy of the system and reduce the number of more relevant mistakes to support the diagnosis of diseases. In this context, we propose and evaluate the single classification method and the cascade classification method to classify the images into 11 classes (i.e., Basophil, Band, Eosinophil, Erythroblast, Thrombocyte, Lymphocyte, Metamyelocyte, Monocyte, Myelocyte, Segmented and Variant) separating the images in respective folders and creating an extra folder called Unclassified for the images that could not be classified with a high class-confidence value.

2. Literature Review

Kratz et al.[5] evaluated the clinical performance of CellaVision against manual microscopy. In the experiments performed, CellaVision achieved 82% accuracy. The correct classification rate was higher for mature than for immature and abnormal cells and the highest coefficients were obtained for segmented neutrophils and lymphocytes while the lowest were obtained for basophils and eosinophils. The motivation of our work is to propose a solution for CellaVision’s low accuracy, so the specialists don’t have to reclassify all the images segmented by the system.

A paper about the WBC classification was written by Hedge et al.[3]. The authors compare the classification made by traditional image processing and using convolutional neural networks trained for 1,000 epochs. The authors used 1,418 cropped images to build tools to classify images into 6 classes, i.e., lymphocytes, monocytes, neutrophils, eosinophils, basophils, and abnormal cells. As pre-processing the images were cropped to segment just the cell. They obtained 99% accuracy with full training CNN and got even better results, 99.8% using the classifier on hand-crafted features.

Kutlu et al.[2)] evaluated the use of neural networks for WBC classification into 5 classes (i.e., Lymphocytes, Monocytes, Eosinophils, Basophils, and Neutrophils). Using the neural network architecture ResNet50 with a transfer learning method, and a database with 5,000 images (1,000 images of each class), an accuracy of 97%, recall of 99%, and precision of 97% was obtained for the threshold of 0.3, from training the neural network 3,000 epochs.

Shahin et al.[8)] used two different transfer learning methods, fine-tuning AlexNet neural network and deep activation features for training 2,551 images, including 5 healthy WBC types. The accuracy obtained was 96.1%.

Some versions of the YOLO series were used in the WBC classification. Abass et al.[9)] used YOLOv2 to identify and classify 3 types of WBC. The goal was to support the leukemia diagnosis. The neural network was used in two steps to first segment and later classify the images, and they obtained an accuracy of 94.3%. The deep learning architecture used, YOLOv2, was launched in 2017.

Praveen at. al. [10)] used YOLOv3 to classify 4 types of WBCs. The process was divided into detection and classification. The method achieved an accuracy of 90%. Their dataset doesn't consider abnormal cases. The deep learning architecture used, YOLOv3, was launched in 2018.

Akalin et al.[11)] used YOLOv5, launched in 2021, to perform the classification of 5 types of WBCs. Only the normal types of WBCs are used in the database and the classification as well. The paper's main discussion is the consequence of using two different neural network architectures combined to make the classification. The approach used by Akalin obtained 98% accuracy in the hybrid approach and 83.3% and 94.66% using YOLOv5 with 2 different types of activation functions. Akalin's paper doesn't use a different evaluation dataset, the metrics discussed are the ones for the images used during the training of the neural network, so the method cannot be precisely evaluated.

To the best of our knowledge, no previous work was done with the goal of classifying WBCs from CellaVision into 11 classes, including abnormal cells. In the literature, it is most common the classification by 5 classes of normal WBCs, however, it is crucial to consider the abnormal cells in the database and classify into more precise classes for aiding the diagnosis in the laboratories. Also, since the deep learning architecture YOLO has been developing day by day, the authors solve this problem using the latest version, YOLOv7, launched in 2022, in this work. Furthermore, we evaluate the obtained results not only in terms of accuracy but also from a clinical point of view analyzing the impact of the mistakes for the diagnosis, while other works don't evaluate the results from this point of view.

3. Deep Learning and YOLOv7

Artificial neural networks simulate the human brain's process of learning. An artificial neural network, like the human neural network, is composed of connected neurons.

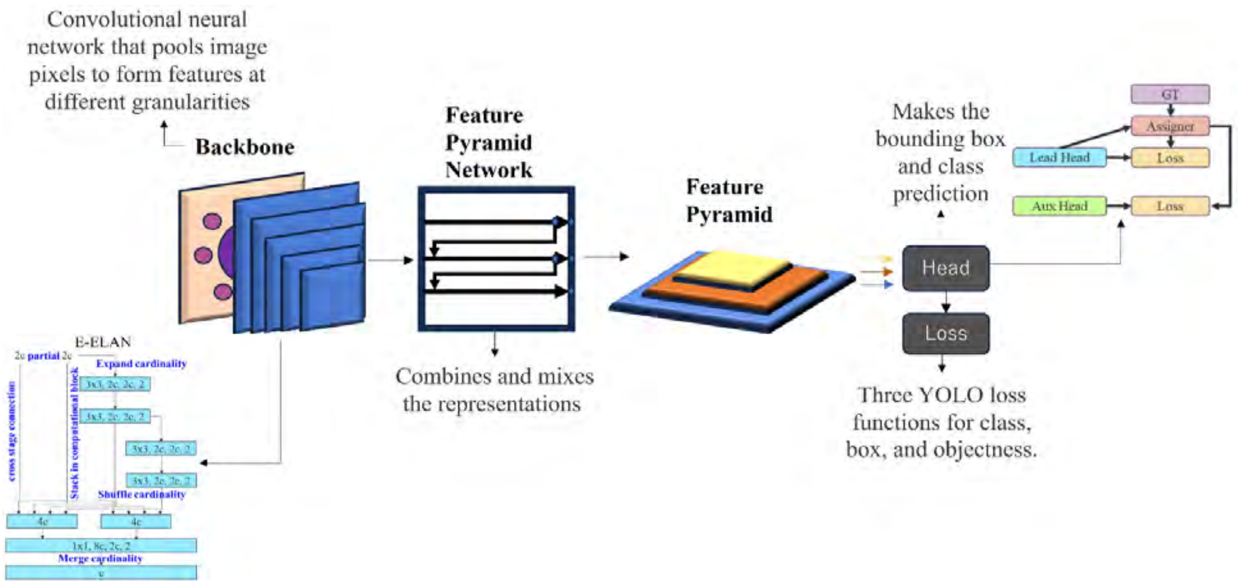

Convolutional neural network that pools image pixels to form features at different granularities
Backbone
Feature Pyramid Network
Combines and mixes the representations
Feature Pyramid
Makes the bounding box and class prediction
Head
Loss
Three YOLO loss functions for class, box, and objectness.
GT
Assigner
Lead Head
Loss
Aux Head
Loss
E-ELAN
2c partial 2c
Expand cardinality
cross stage connection
Stack in computational block
3x3, 2c, 2c, 2
3x3, 2c, 2c, 2
3x3, 2c, 2c, 2
3x3, 2c, 2c, 2
Shuffle cardinality
4c
4c
1x1, 8c, 2c, 2
Merge cardinality
c

Fig. 2 YOLOv7 architecture

The information passed from one neuron to another is transformed according to a parameter called “weight”. These weights are modified in the training process. Deep learning architectures are neural networks with multiple layers[6]. Convolutional neural networks are deep learning architectures inspired by visual perception, capable of object detection and classification in images[12]. The use of convolutional neural networks stands out compared to traditional image-processing methods mainly because of the high performance, but also for the characteristics of the process. For the neural network training it's not necessary any pre-processing or pre-feature extraction and selection. The features are extracted by the convolutional layers[2].

The deep learning architecture used in this work was YOLOv7 that is the 7th version of the YOLO algorithm, released in 2022, and increased accuracy, speed (can achieve 600fps in the detection), and ability to detect a wider range of objects (i.e., high performance detect either small, medium or big objects) compared to the previous versions of YOLO and some of the most used deep learning algorithms. YOLOv7, like some other object detectors, performs image recognition by predicting bounding boxes and class probabilities for the objects in the image, using a convolutional neural network to extract features from the image. YOLOv7 architecture is represented in **Fig. 2**. YOLO frameworks have three main components, the backbone, the head, and the neck. The backbone mainly extracts essential features of an image and feeds them to the head through the neck. The neck collects feature maps extracted by the backbone and creates feature pyramids. Finally, the head consists of output layers that have final detections. The computational block used in YOLOv7 architecture is called Extended Efficient Layer Aggregation Network (i.e., E-ELAN) and allows better learning of the model by expanding, shuffling, and merging cardinality. The training process is optimized with an architecture called “Bag of Freebies”. Another optimization is Deep Supervision Label Assignment which adds auxiliary heads in the middle layers to help the model converge well[7]. YOLOv7 comprehends the desired characteristics for this study as high accuracy and speed compared to other object detectors, prediction of bounding boxes, the possibility of training many classes at the same time, and training images even bigger than the original size (i.e., 360×363pixels).

4. Proposed Methods

An overview of the methodology is shown in the Schematic Diagram in **Fig. 3**. The blood samples are input into the CellaVision machine that outputs WBC images segmented but containing many misclassifications. Considering the low classification accuracy by CellaVision and the importance of classifying normal and abnormal cells, we propose the reclassification of these images using YOLOv7 to increase the accuracy in single and cascade classification methods. A detailed description of both methods is presented in the following subsections.

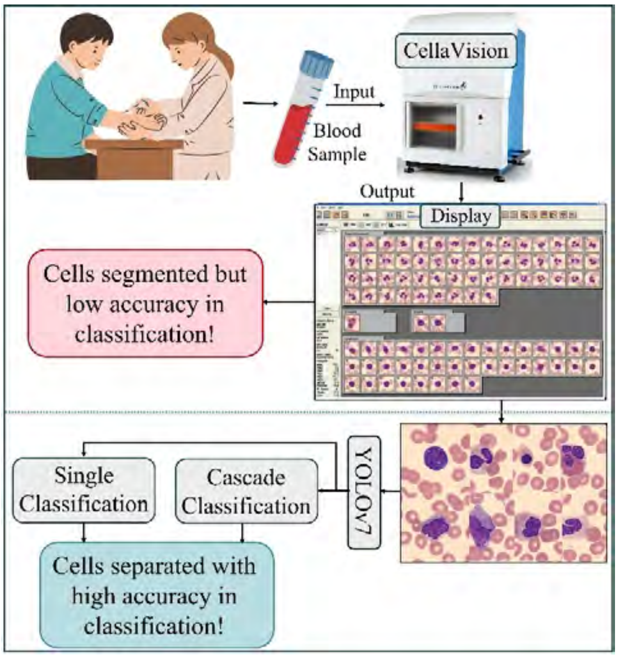

CellaVision
Input
Blood Sample
Output
Display
Cells segmented but low accuracy in classification!
YOLOv7
Single Classification
Cascade Classification
Cells separated with high accuracy in classification!

Fig. 3 Schematic Diagram of the methodology

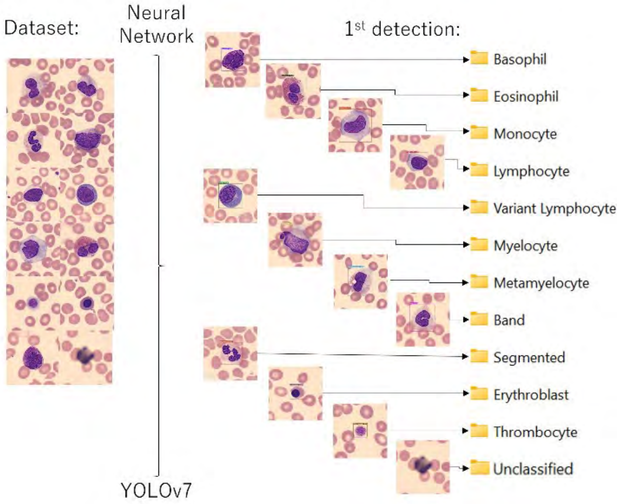


Fig. 4 Single classification method. The images are classified into 11 classes and separated into respective folders. An extra folder called Unclassified is also created for unclassified images

4.1 Single classification method

In the single classification method, represented in **Fig. 4**, only one folder is prepared with all the images for the training dataset, subdivided into train (i.e., 80%) and validation (i.e., 20%). The images in the subfolders are changed for each training in the five-fold cross-validation. The deep learning architecture YOLOv7 was trained for the classification of the images into 11 classes (i.e., band, segmented, basophil, eosinophil, erythroblast, thrombocyte, lymphocyte, variant, metamyelocyte, monocyte, and myelocyte). With the trained weights, the neural network can classify WBCs in images never seen by the neural network before, drawing a bounding box around these objects, and presenting the class with its class confidence value. If there is more than one detection in an image, only the detection with higher class confidence is considered. The classification algorithm creates 11 folders for each class and the images with the cells already classified and tagged, are saved in the respective folders. An extra folder called "Unclassified" is also created to store the images that have no detections with a high class confidence value.

The official training scripts of YOLOv7 were not modified but for the detection and classification, a modified script had to be prepared. In the case of two cells identified in the same image, the scripts consider only the detection with a higher class confidence value. The single classification script creates 12 folders (i.e., one for each class and one for unclassified images), runs the detection, and saves the images into the respective folders, sending the images with no cell detection or detection with class confidence value lower than 0.7 into the folder "Unclassified".

4.2 Cascade Classification Method

For the cascade classification, represented in **Fig. 5**, the images are classified in two phases using 4 different trained weights set. The training dataset for cascade classification is composed of 4 folders, one for the first phase of the classification and 3 for the second phase. First, the cells will be classified between the 5 classes of WBCs (i.e., basophil, eosinophil, lymphocyte, monocyte, and neutrophil) and an extra class called "Not WBCs" that represent images with other types of cells present in the database. The classification algorithm creates separate folders and saves the classified images into different folders. In each one of these folders, the classification of subclasses is performed. The images classified as lymphocyte are reclassified into lymphocyte and variant, the images classified as neutrophils are reclassified into band, metamyelocyte, myelocyte, and segmented, and in the images classified as not WBCs, erythroblast, and thrombocyte are also reclassified. If there is more than one detection in an image, only the detection with higher class confidence is considered. 11 folders are created to store images from each one of the classes, and the "Unclassified" folder is created to store the images that have no detections with high class confidence values.

The flowchart for the cascade classification detection script is shown in **Fig. 6**. The script first creates 6 folders (i.e., one for each WBC type and a not WBC folder), and inside these folders create the respective subfolders. The images are first classified into the 6 groups using the 1st layer trained weights, then all the images stored in the folders with subclassifications are reclassified by the 2nd layer trained weights for their cell type and the images are moved to the subclass folder.

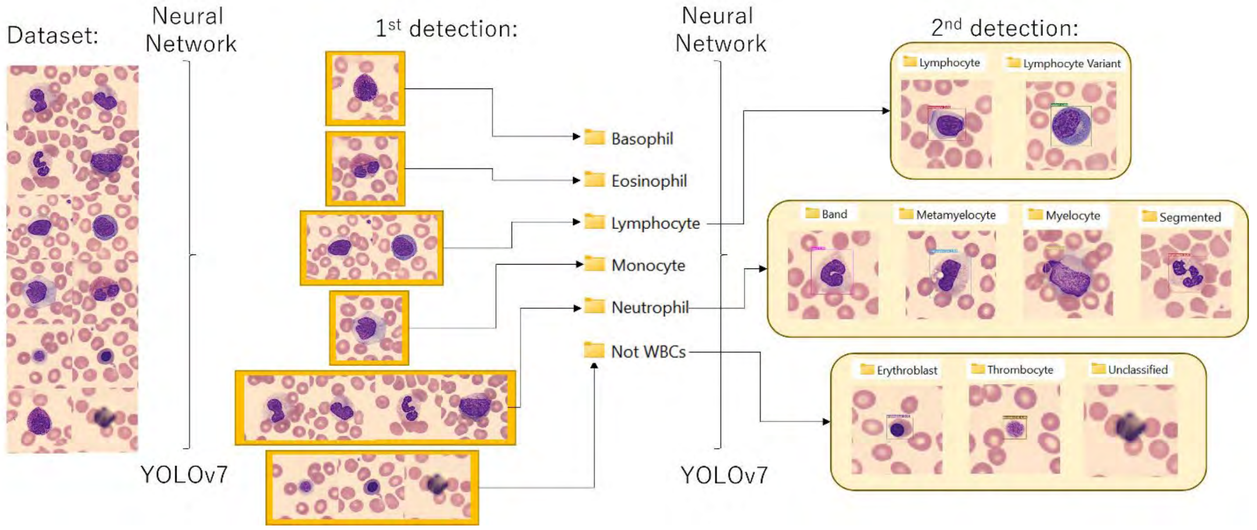


Fig. 5 Cascade method. First, the images are classified into 6 classes representing the 5 types of white blood cells images and another file for other images. In the second phase, images are reclassified into subclasses and separated into other folders, resulting in 11 class folders plus one extra folder for images outside the 11 defined classes.

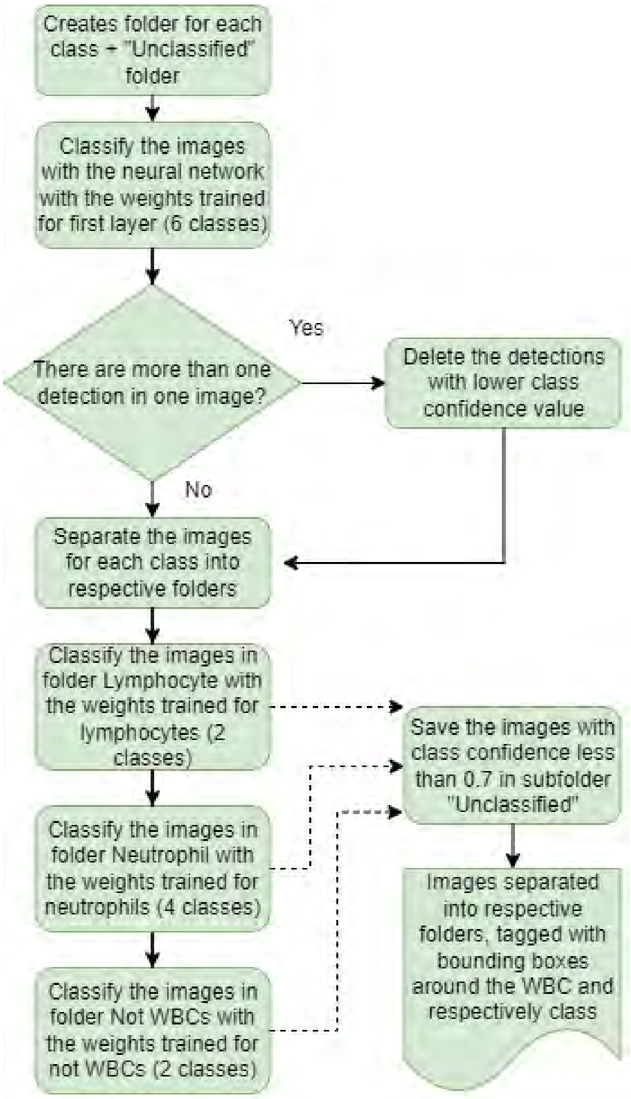

Creates folder for each class + "Unclassified" folder
Classify the images with the neural network with the weights trained for first layer (6 classes)
There are more than one detection in one image?
Yes
Delete the detections with lower class confidence value
No
Separate the images for each class into respective folders
Classify the images in folder Lymphocyte with the weights trained for lymphocytes (2 classes)
Classify the images in folder Neutrophil with the weights trained for neutrophils (4 classes)
Classify the images in folder Not WBCs with the weights trained for not WBCs (2 classes)
Save the images with class confidence less than 0.7 in subfolder "Unclassified"
Images separated into respective folders, tagged with bounding boxes around the WBC and respectively class

Fig. 6 Flowchart of the detection algorithm in cascade classification method

4.3 "Unclassified" folder

Sometimes, in the medical routine, abnormal cells very different from the ones more commonly seen are found in the blood samples. These cells for their different aspects are often not correctly classified by the machines, but their analysis is very important once it can happen in cases of rare diseases. Cases of problems in the sample preparation or the image scanning also can make the classification process by the systems difficult. It is important in the development of the system to consider that only images with high class confidence value are classified so these images don't have to be reassessed by specialists. A folder called "Unclassified" was created to store these images that could not be classified with the minimum class confidence value set for no of the WBC types, so the specialists only must access this folder to identify the unusual abnormal cases. Giant Thrombocytes and Erythroblast cells are not WBCs but have clinical relevance and the number of these images segmented by CellaVision is very high, so it's also necessary to classify and separate these images into different folders inside the group of cells not classified as WBCs so these cells also don't have to be reassessed by the professionals.

5. Experimental Setup

5.1 Implementation

The computer used for the experiments has an Intel(R) Core (TM) i9-10900F processor, 32GB RAM, and a video card NVIDIA GeForce RTX 3090. The application codes were written in Python 3.9.

The images obtained from CellaVision have 360×363 pixels so, for the training, the image size set in the hyperparameters was 384×384, which was the closest possible value. The weights of the neural network were trained separately for each classification for 3,000 epochs. The weights were saved for each epoch, so the best weights set, i.e., the ones with higher accuracy, precision, mAP@0.5, and mAP@0.5:0.95 scores, were selected.

After the training, an F1-Score graph was plotted with the class confidence values. By the graph, it is possible to analyze the F1-Score result for each possible class confidence value. We set a threshold value of 0.7 for all detections. This value got high results in the F1-Score curve for all the trainings and was chosen also because for the application is important that the cells are classified only if they have a high class confidence to guarantee the reliability of the system. For each method, five-fold cross-validation was performed, resulting in information about average and highest accuracy.

5.2 Database

The database includes healthy WBCs, abnormal WBCs, and other cells with clinical interest commonly mistaken for WBCs. The dataset used comprehends 5850 blood cell images of size 360×363 pixels, from blood samples collected between 2018 and 2019 in Shinshu University Hospital. The data comprehends only blood images and the class of each cell. No information from the patients was collected, respecting the patient's privacy. The images were downloaded from the CellaVision system and classified by two excellent blood analysis specialists. Each one of the specialists analyzed the images alone and on different dates. Only the images classified equally for both specialists were used in this study.

Images of 11 classes were obtained, i.e., band neutrophil, segmented neutrophil, basophil, eosinophil, erythroblast, lymphocyte, variant lymphocyte, metamyelocyte, monocyte, myelocyte, and thrombocyte. From these 5,850 images, 2,000 were separated for the evaluation of the method and 3,850, comprehending a similar number of each class, were used as training dataset.

For each image in all the training datasets, a text file was created, in a specific model for YOLO training, containing a number that corresponds to the class, the position, and the size of the bounding box that involves the cell presented in the image. The same images were used for each neural network training in the single and the cascade classification methods.

The 2,000 images validation dataset comprehends all the classes but not with equal quantities. In this evaluation dataset, specialists classed 269 (i.e., 13.45%) images as neutrophil band, 576 (i.e., 28.8%) images as neutrophil segmented, 101 (i.e., 5.05%) images as basophil, 97 (i.e., 4.85%) images as eosinophil, 106 (i.e., 5.3%) images as erythroblast, 117 (i.e., 5.85%) images as lymphocyte, 94 (i.e., 4.7%) images as lymphocyte variant, 143 (i.e., 7.15%) images as metamyelocyte, 272 (i.e., 13.6%) images as monocyte, 95 (i.e. 4.75%) images as myelocyte, and 130 (i.e., 6.5%) images as thrombocyte. A blood sample contains different amounts of each cell (e.g., the segmented neutrophils amount is higher than the other blood cell types) so the evaluation dataset has an adequate amount of each class but having a greater number of segmented neutrophils represents better the performance that would be obtained in a real situation.

5.3 Metrics for evaluation

For the neural network training evaluation, the metrics used are precision (i.e., P), which represents the percentage of correct classifications, recall (i.e., R), which reflects the capability of the neural network in finding correct items, mAP@0.5, which is the mean average precision and can be calculated as the area under a precision-recall curve, and mAP@0.5:0.95, that considers the IoU (i.e., Area of overlap divided by area of union) values from 50% to 95% at step of 5%. For the evaluation of the proposed method, besides P and R, the accuracy (i.e., A), which represents the proportion of the correct predictions, and F1-Score (i.e., F1), that is the harmonic mean between precision and recall, was also used.

True positives (i.e., TP) represent the images correctly classified, False Positives (i.e., FP) are the cells identified with the wrong classification, and false negatives (i.e., FN) are the cells not classified or classified with a confidence value below 0.7, i.e., images in the file “Unclassified”. As the objective is to classify cell images already segmented by CellaVision, all images used contain a cell and therefore the system does not have true negatives. The system only aims to reclassify and separate the images into their respective folders so the limits of the bounding boxes that identify the cells are not evaluated.

$$A = \frac{TP}{TP+FP+FN} \tag{1}$$

$$P = \frac{TP}{TP+FP} \tag{2}$$

$$R = \frac{TP}{TP+FN} \tag{3}$$

$$F1 = \frac{2*P*R}{P+R} \tag{4}$$

5.4 Five-fold Cross Validation

A five-fold Cross Validation[13] based method was used in the training of the neural network. The method applied consists of 5 trainings of the neural network. The training dataset is randomly divided into 5 parts, containing a similar number of images from each class, and, for each training, 1 different part (i.e., 20% of the dataset) is used for the validation of the training while the other 4 parts, (i.e., 80% of the dataset) is used in the training. With this method, it is possible to obtain the mean error rate of the trainings, an error rate value more consistent than the one obtained in only one training. The database division is represented in **Fig. 7**.

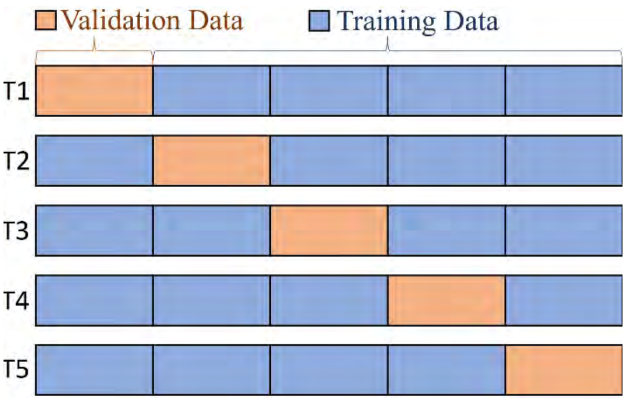


Fig. 7 Database representation for the 5 experiments for each method

5.5 Transfer learning

Transfer learning is a method used in the training of neural networks that consists of using weights trained for different applications as a starting point for training the desired application. This technique reduces training time and can bring better results, as the neural network would already have a basic understanding of relevant objects and characteristics, such as edges, textures, and colors, among others. The pre-trained weights used for Transfer Learning in this work are provided by the YOLOv7 authors in the official Yolov7 repository[14] and were trained to detect 80 different classes (e.g., person, airplane, umbrella, orange, ...).

6. Results and Discussion

The deep learning architecture YOLOv7 showed satisfactory results in WBC classification. **Figure 8** shows an example of a WBC image classified and tagged by YOLOv7. We show the details in the following.

In the training phase, the cascade method showed better results compared to the single classification method, as we can see comparing **Fig. 9** and **Fig. 10** obtained in training 1. While the F1-Score for the single classification was 96%, the F1-Score obtained in the cascade method was 97.7%. Analyzing F1-Score we are, at the same time, analyzing the capability of the neural network in finding correct items and analyzing the classifications made, because of that F1-Score is the most important metric to be analyzed in the training.

Table 1 shows the metrics results in CellaVision classification, the results for each one of the trainings, and the mean results for single and cascade classification. For all the metrics, the proposed methods obtained significantly

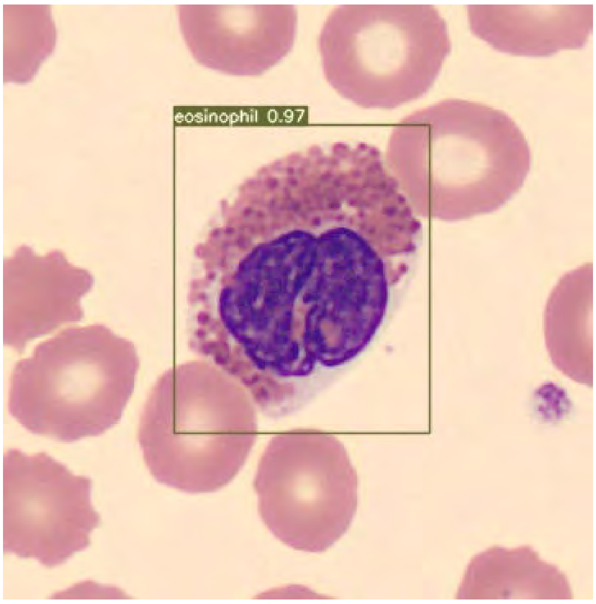


Fig.8 WBC image classified by YOLOv7

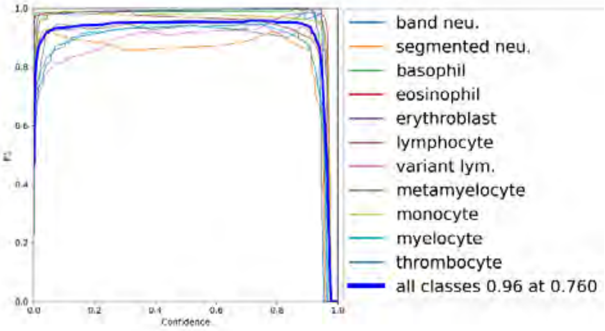


Fig. 9 F1-Score graph for single classification method

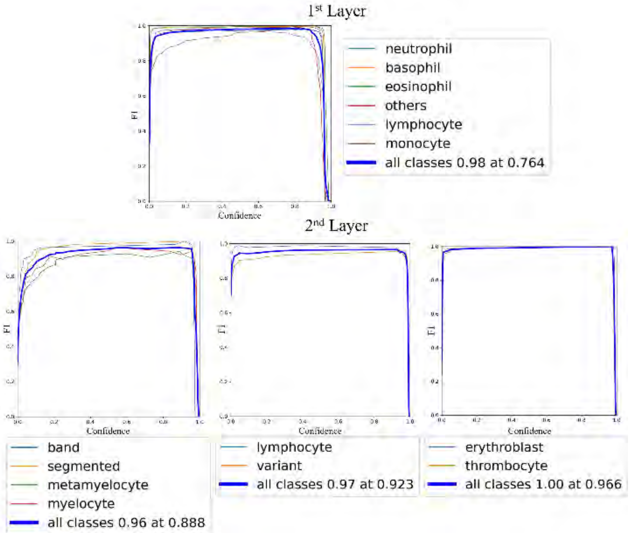


Fig. 10 F1-Score graphs for each classification layer in cascade classification method

Table 1 Results for CellaVision classification (i.e., CV), each one of the trainings and mean for single classification (i.e., SC) and cascade classification (i.e., CC) methods

	Train	A	P	R	F1
CV		76.20%	80.93%	92.87%	86.49%
SC	Train 1	94.7%	96.63%	97.93%	97.28%
SC	Train 2	93.85%	96.45%	97.2%	96.83%
SC	Train 3	94%	97.41%	96.41%	96.91%
SC	Train 4	91.85%	95.18%	96.33%	95.75%
SC	Train 5	93.55%	95.12%	98.27%	96.67%
SC	**Mean**	**93.59%**	**96.16%**	**97.23%**	**96.69%**
CC	Train 1	94.55%	95.55%	98.90%	97.20%
CC	Train 2	94.20%	97.01%	97.01%	97.01%
CC	Train 3	93.20%	97.13%	95.83%	96.48%
CC	Train 4	93.80%	97.96%	95.67%	96.80%
CC	Train 5	93.5%	96.40%	96.89%	96.64%
CC	**Mean**	**93.85%**	**96.81%**	**96.86%**	**96.84%**

better results compared to the CellaVision ones, and the cascade classification obtained the mean and maximum best results. The accuracy of the CellaVision classification for the validation database is 76.20%, the precision is 80.93%, the recall is 92.87% and the F1-Score is 86.49%. In CellaVision classification, 5.85% of the images were wrongly classified outside the proposed classes as “Artefact” and “Smudge cell”. In single classification, we obtained a mean accuracy of 93.59%, a mean precision of 96.16%, a mean recall of 97.23%, and a mean F1-Score of 96.69%.

The maximum values obtained in five-fold cross-validation were accuracy of 94.7 % in 1st training, precision of 97.4% in the 3rd training, recall of 98.27% in the 5th training, and F1-Score of 97.28% in 1s training. A mean of 2.67% of the images were classified as “Unclassified” in this method. In the cascade classification method, we obtained a mean accuracy of 93.85%, mean precision of 96.81%, mean recall of 96.86%, and mean F1-Score of 96.83%. The best values obtained in five-fold cross-validation were an accuracy of 94.55% in 1st training, a precision of 97.96% in the 4th training, recall of 98.90% in the 1st training, and F1-Score of 97.20% in the 1st training. A mean of 3.05% of the images was classified as “Unclassified”.

Compared to the CellaVision results, the average accuracy increased by 17.39 percentage points using the single classification method and 17.65 percentage points using the cascade method, the average precision increased by 15.23 percentage points using the single classification method and 15.88 percentage points using the cascade method, the mean recall increased 4.36 percentage points using the single classification method and 3.99 percentage points using the cascade method, and the average F1-Score increased 10.79 percentage points using the single classification method and 10.35 percentage points using the cascade method. The metrics for the two methods have approximate results, accuracy increased by 0.26, precision increased by 0.65, recall decreased by 0.37 and F1-Score increased by 0.15 percentage points from single to cascade classification methods.

Table 2 shows the accuracy results, for each class and the mean of the accuracies, for CellaVision classification, single classification, first layer of cascade classification, and second layer of cascade classification. The biggest difference between the accuracy values obtained between single classification and cascade classification occurred in the "Monocyte" class where cascade classification had a better accuracy by 10.01 percentage points. If we consider the mean of accuracies by class, which represents the accuracy if the sample has the same number of cells for each class, accuracy for single classification is 0.48 percentage points higher than cascade classification, but it is important to analyze not just the mistakes but also the kind of these mistakes.

Table 2 Accuracy results by class and mean of accuracies for CellaVision (i.e., CV) classification, single classification (i.e., SC), first layer of cascade classification (i.e., CC l1), and second layer of cascade classification (i.e., CC l2).

Class	CV	SC	CC l1	CC l2
Basophil	92.07%	100%	99.41%	99.41%
Eosinophil	98.96%	100%	100%	100%
Lymphocyte	85.47%	92.99%	97.27%	90.28%
Variant	31.91%	96.80%	92.98%	91.63%
Monocyte	92.27%	83.45%	93.46%	93.46%
Band	38.88%	95.85%	99.85%	95.20%
Meta	50%	75.63%	99.30%	71.76%
Myelocyte	38.94%	96.42%	99.16%	95.81%
Segmented	91.84%	96.21%	99.76%	95.15%
Erythroblast	77.35%	99.05%	99.43%	99.06%
Thrombocyte	100%	100%	100%	99.38%
Mean	72.52%	94.22%	98.24%	93.74%

Table 3, Table 4 and **Table 5** shows the error rate between the classes and **Fig.11** shows the most common mistakes (i.e., higher than 3% error rate) in CellaVision, single classification, and cascade classification. We notice that in CellaVision classification, different WBC types (e.g., Monocyte and Myelocyte), cells that would be easily identified by human classification (e.g., Eosinophil and Erythroblast), and WBC cells with not WBC (e.g., Erythroblast and Lymphocyte) are commonly mistaken. We also notice that in single classification there are still common mistakes between different types of WBC (i.e., Lymphocyte Variant and Monocyte), and in the cascade classification method, mistakes between different classes are avoided and the images are sent to the “Unclassified” folder instead of being mistake with another wrong class.

Examples of misclassifications are shown in **Fig. 12**. Here, images a) to c), were misclassified by CellaVision but correctly classified by both single and cascade classification methods, while image d) was misclassified by CellaVision and single classification but correctly classified by cascade classification. In the image a) there is an image misclassified with a different type of WBC, which compromises the analysis of quantity of each WBC type. Images in b) and c) show erythroblasts. In both cases, it was mistaken by WBCs

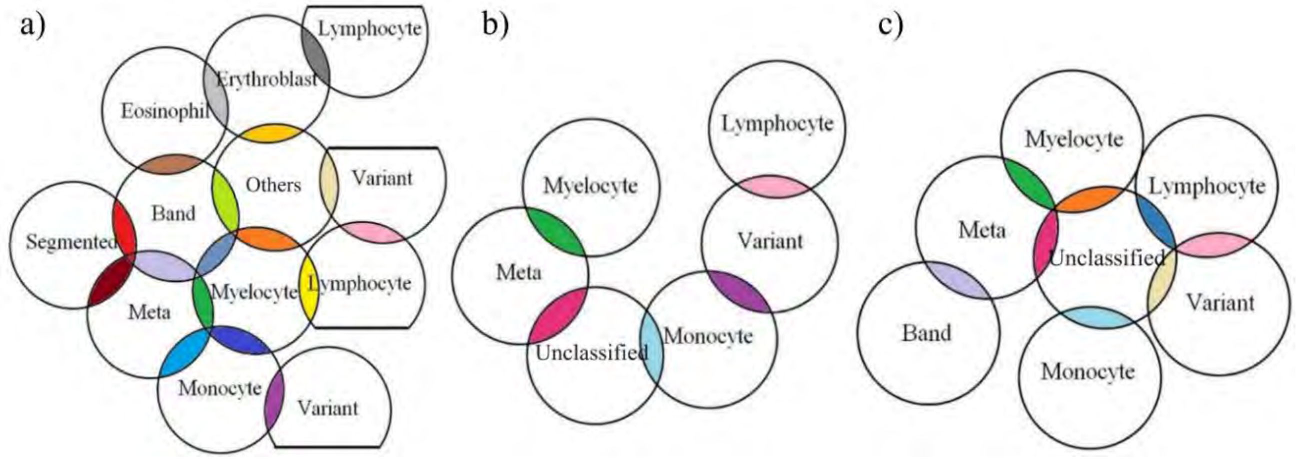


Fig.11 Common mistakes between classes for a) CellaVision, b) Single classification and c) Cascade classification. Intersections represents more than 3% of error rate between the classes.

Table 3 Error rates between the classes in CellaVision for Fig. 11 a)

CV	Bas	Met	Mye	Ban	Seg	Eos	Lym	Var	Mon	Ery	Thr	Oth
Bas	92	0.99	1.98	0	0.99	1.98	0	0	0	0	0	1.98
Met	0	50	3.52	31.7	4.23	0.70	0	0	7.04	0	0	2.82
Mye	2.11	15.8	38.9	5.26	0	1.05	7.37	0	18.9	0	1.05	9.47
Ban	0	0	0	38.8	48.52	3.70	0	0	0	0	0	8.89
Seg	0.35	0	0	3.47	91.8	2.78	0	0	0	0	0	1.56
Eos	0	0	0	0	0	98.9	0	0	0	0	0	1.03
Lym	0	0	0	0	0	0	85.4	13.7	0	0	0	0.85
Var	0	0	0	0	0	0	3.19	31.9	1.06	0	0	63.8
Mon	0	0	0	0	0	0	0	6.62	92.2	0	0	1.10
Ery	1.89	0	0	2.83	0.94	4.72	7.55	0	0.94	77.3	0	3.77
Thr	0	0	0	0	0	0	0	0	0	0	100	0

Table 4 Errors rate between the classes in single classification for Fig. 11 b).

SC	Bas	Met	Mye	Ban	Seg	Eos	Lym	Var	Mon	Ery	Thr	Unc
Bas	100	0	0	0	0	0	0	0	0	0	0	0
Met	0	75.6	13.2	1.97	0.28	0	0	0	0	0.28	0	8.59
Mye	0	1.47	96.4	0	0	0	0	0.21	0	0	0	1.89
Ban	0	0.30	0	95.8	1.93	0	0	0	0	0	0	1.93
Seg	0.14	0	0	1.35	96.2	0	0	0	0	0	0	2.29
Eos	0	0	0	0	0	100	0	0	0	0	0	0
Lym	0	0	0.34	0	0	0	92.9	3.76	0	0.17	0	2.74
Var	0	0	0	0	0	0	1.28	96.8	0	0.21	0	1.70
Mon	0.29	0.07	0.29	0	0	0	0.37	9.56	83.4	0	0	5.96
Ery	0	0	0	0	0	0	0.19	0.75	0	99.0	0	0
Thr	0	0	0	0	0	0	0	0	0	0	100	0

Table 5 Errors rate between classes in Cascade classification for Fig. 11 c).

CC	Bas	Met	Mye	Ban	Seg	Eos	Lym	Var	Mon	Ery	Thr	Unc
Bas	99.4	0	0	0	0	0	0	0	0	0	0	0.59
Met	0	71.7	14.5	3.36	0.70	0	0	0	0.42	0	0	9.25
Mye	0	0.62	95.8	0	0	0	0.21	0	0	0	0	3.35
Ban	0	0.67	0	95.2	1.77	0	0	0	0	0	0	2.36
Seg	0	0	0	1.94	95.1	0	0	0	0	0	0	2.91
Eos	0	0	0	0	0	100	0	0	0	0	0	0
Lym	0	0	0	0.51	0	0	90.2	4.82	0.17	0	0	4.21
Var	0	0	0	0	0	0	0.99	91.6	1.49	0	0	5.89
Mon	0	0	0	0.37	0	0	2.06	0	93.4	0	0	4.12
Ery	0	0	0	0	0	0	0.57	0	0	99.0	0	0.37
Thr	0	0	0	0	0	0	0	0	0	0	99.3	0.62

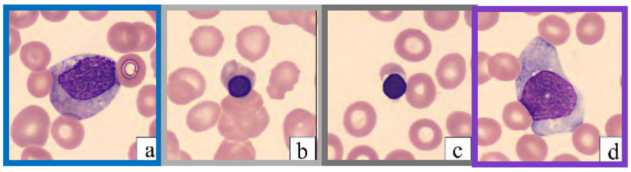


Fig. 12 Example of images misclassified: a) the class is Myelocyte. It was misclassified as Monocyte by Cellavision and correctly classified by single and cascade classification, b) the class is Erythroblast. It was misclassified as Eosinophil by CellaVision and Correctly classified by Single and cascade classification, c) the correct class is Erythroblast but it was misclassified as Lymphocyte by CellaVision and correctly classified by single and cascade classification, d) the class is Monocyte. It was misclassified as Variant by Cellavision and single classification but correctly classified by cascade classification.

Table 6 Comparison with related works

Reference	No. of Classes	Cell types	Deep Learning	Database	Main Features	Accuracy
Shahin et al. [8)]	5	Normal	AlexNet*	3 public databases (2,551 images)	Single classification + 2 Transfer Learning methods	92.9%
Hedge et al. [3)]	6	Normal + Abnormal**	AlexNet	1,418 cropped images (hand crafted features)	Deep Learning + Traditional Image Processing	99.6%
Kutlu et al. [2)]	5	Normal	AlexNet	410 images from BCCD + 242 images from LISC open datasets	Single classification	97%
Praveen et al. [11)]	4	Normal	YOLOv3	364 images from BCCD Dataset	Segmentation and Classification in different steps	90%
Abass et al. [9)]	3	Leukemia	YOLOv2	2,700 images diagnosed with Leukemia from VIN Hospital	Segmentation and Classification in different steps	94.3%
Akalin et al. [10)]	5	Normal	YOLOv5, Detectron2	1,000 images from Raabin Health Database	2 neural networks combined	98%
Proposed method	11	Normal + Abnormal	YOLOv7	5,850 images from Shinshu University Hospital obtained with CellaVision	Single and cascade classification	93.85% (98.24%***)

* AlexNet modified by the authors
** All kinds of abnormal cells are considered just one class
*** Result of the first layer of cascade classification into 6 classes

by CellaVision, which also compromises the total WBC count. Besides, these images would be easily classified by humans. The image d) shows an example where single classification method misclassified an image, compromising the counting of each WBC type, which is a problem in the single classification method.

If we consider that the images present in the Unclassified folder are all correctly classified by an expert, the new accuracy would be 96.26% for single classification and 96.9% for cascade classification and the images rechecked would be, on average, 54 for single classification and 61 for cascade classification what represents, respectively, 2.67% and 3.05% of the images that the professionals check without the methods. Although the methodologies still have errors, the health professionals who worked on this study consider that the error is clinically irrelevant, that is, the errors presented would not make any difference in the medical routine. Finally, **Table 6** compares related works including the proposed methods, in which we compare the number of classes to be distinguished, cell types (normal and abnormal), deep learning architectures, database and images used, main features, and accuracy performed by each method. Note that the figures of accuracy in this table are only for reference since a fair comparison cannot be possible.

7. Conclusions

This work studies the improvement of the quality of the classification, using the deep learning architecture YOLOv7 in single and cascade classification methods, of WBC images segmented by the system CellaVision DM96. Both evaluated methods presented significantly better results than the ones obtained by CellaVision, not just in terms of metrics but also reducing the criticality of the mistakes. The CellaVision detections had an accuracy of 76.20%, precision of 80.93%, recall of 92.87%, and F1-Score of 86.49%. The single classification method showed an mean accuracy of 93.59%, precision of 96.16%, recall of 97.23%, and F1-Score of 96.69%. The cascade method resulted in an mean accuracy of 93.85%, precision of 96.81%, recall of 96.86%, and F1-Score of 96.83% for the 0.7 threshold. The methods separates the images into files representing their respective classes and a different file with the images that could not be classified with the chosen threshold and classes (i.e., "Unclassified"). The methods, mainly the cascade classification, facilitate the medical routine because, without them, all the images obtained had to be reassessed due to the low accuracy of the CellaVision classification, with the discussed methods, with the increase in accuracy and the reduction of the most serious errors (e.g., errors between WBC and non-WBC cells, errors between different WBC tipes and errors that would not be made by human classification), the professionals already have the cells separated and classified and they only need to reassess the images present in the folder “Unclassified”, reducing drastically the heavy burden (i.e., time, energy) in the medical routine.

In future work, we will work on defining the maturity level of neutrophils to support the diagnosis of bacterial infection. We will also implement the proposed method in a medical laboratory and compare the performance using other new neural network architectures.

References

1) Harold Schumacher, William Rock, Sanford Stass, Handbook of Hematologic Pathology, CRC Press; 1 edition (2000).

2) Hüseyin Kutlu, Engin Avci, Fatih Özyurt: "White Blood Cells Detection and Classification Based on Regional Convolutional Neural Networks", Medical Hypotheses, Volume 135, 2020, 109472, ISSN 0306-9877, https://doi.org/10.1016/j.mehy.2019.109472.

3) Roopa B. Hegde, Keerthana Prasad, Harishchandra Hebbar, Brij Mohan Kumar Singh: "Comparison of Traditional Image Processing and Deep Learning Approaches for Classification of White Blood Cells in Peripheral Blood Smear Images", Biocybernetics and Biomedical Engineering. Volume 39, Issue 2, 2019, pp. 382–392, ISSN 0208-5216. https://doi.org/10.1016/j.bbe.2019.01.005.

4) CellaVision DM96 –Automated Digital Cell Morphology–, 2009. Sysmex America Inc. Document Number MKT10-1033 06/2009 2M. https://www.sysmex.com/US/en/products/hematology/cellimage analysis/documents/brochure_dm96.pdf. Accessed 31 August 2023.

5) Alexander Kratz, Hans-Inge Bengtsson, Jeanne E. Casey, Joan M. Keefe, Gail H. Beatrice, Debera Y. Grzybek, Kent B. Lewandrowski, Elizabeth M. Van Cott: “Performance Evaluation of the CellaVision DM96 System: WBC Differentials by Automated Digital Image Analysis Supported by an Artificial Neural Network” , American Journal of Clinical Pathology, Volume 124, Issue 5, November 2005, pp. 770–781, https://doi.org/10.1309/XMB9K0J41LHLATAY.

6) Yu-chen Wu, Jun-wen Feng: “Development and Application of Artificial Neural Network2, Wireless Pers Commun102, pp. 1645–1656 (2018). https://doi.org/10.1007/s11277-017-5224-x.

7) Chien-Yao Wang, Alexey Bochkovskiy, Hong-Yuan Mark Liao: “YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors”, Jul 2022.ArXiv./abs/2207.02696

8) Ahmed Ismail Shahin, Yanhui Guo, Khalid M. Amin, Amr A. Sharawi: “White Blood Cells Identification System Based on Convolutional Deep Neural Learning Networks”, Computer Methods and Programs in Biomedicine, Volume 168, 2019, pp. 69–80, ISSN 0169-2607, https://doi.org/10.1016/j.cmpb.2017.11.015.

9) Shakir Mahmood Abass, Adnan Mohsin Abdulazeez, Diyar Qader Zeebaree. "A YOLO and Convolutional Neural Network for the Detection and Classification of Leukocytes in Leukemia", Indonesian Journal of Electrical Engineering and Computer Science 25.1, 200–213 (2022).

10) Fatma Akalin, Nejat Yumusak. "Detection and Classification of White Blood Cells with an Improved Deep Learning-Based Approach." Turkish Journal of Electrical Engineering and Computer Sciences 30.7, 2725–2739 (2022).

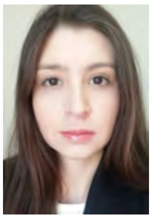
11) Nalla Praveen, et al. "White Blood Cell Subtype Detection and Classification", 2021 8th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI). IEEE (2021).

12) Jiuxiang Gu, Zhenhua Wang, Jason Kuen, Lianyang Ma, Amir Shahroudy, Bing Shuai, Ting Liu, Xingxing Wang, Gang Wang, Jianfei Cai, Tsuhan Chen: “Recent Advances in Convolutional Neural Networks”, Pattern Recognition, Volume 77, 2018, pp. 354–377, ISSN 0031-3203, https://doi.org/10.1016/j.patcog.2017.10.013.

13) Chang-Xue Jack Feng, Zhi-Guang Samuel Yu, Unnati Kingi, M. Pervaiz Baig: “Threefold vs. Fivefold Cross Validation in One-hidden-layer and Two-hidden-layer Predictive Neural Network Modeling of Machining Surface Roughness Data”, Journal of Manufacturing Systems. Volume 24, Issue 2, 2005, pp. 93–107, ISSN 0278-6125, https://doi.org/10.1016/S0278-6125(05)80010-X.

14) Yolov7 repository. Available at https://github.com/WongKinYiu/yolov7 on September 2023.

(Received Oct 15, 2023)
(Revised Dec 20, 2023)



Thalita Munique COSTA

She received her B.S. and M.E. degrees in Electronic Engineering, and in Science, Area of Concentration: Biomedical Engineering, from Federal University of Technology - Paraná (UTFPR), Brazil in 2018 and 2020, respectively. She is now a Dr. Eng. Candidate in Telecommunication System at Shinshu University, Nagano, Japan. Her research interests include Digital Pathology, Artificial Intelligence, and Image Processing.

Yoko USAMI

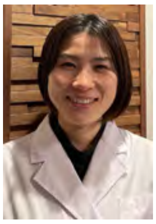
She received her Ph.D. degree from Tokyo Medical and Dental university, Tokyo, Japan in 2013. She joined the Department of Laboratory Medicine at Shinshu University Hospital, Japan where she is currently a vice tech supervisor. Her research interests include Laboratory Medicine, Clinical Chemistry, and Clinical Laboratory Immunology.

Mai IWAYA

She received her M.D. degree from Shinshu University in 2005. She completed anatomical pathology residency at Shinshu University Hospital in 2011 and completed gastrointestinal pathology fellowship at University of Toronto, Canada in 2018. She received the Ph.D. degree from Shinshu University in 2019. She joined the department of laboratory medicine at Shinshu University in 2018 and currently, works as an associate professor/lecturer. Her research interests include artificial intelligence in the pathology field.

Yuka TAKEZAWA

She joined the Department of Laboratory Medicine at Shinshu University Hospital, Japan in 2007. She received her Ph.D. degree from Shinshu University in 2014 and currently, works as a chief. Her research interests include Laboratory Medicine and Clinical Hematology.



Yuika NATORI

She received her bachelor's degree from Teikyo University. She joined the Department of Laboratory Medicine at Shinshu University Hospital in 2014. She is currently in charge of blood and urine tests. Her research interests include laboratory medicine and clinical hematology.

Hernan AGUIRRE

He received an engineering degree in computer systems from Escuela Politécnica Nacional, Ecuador, in 1992 and a Ph.D. from Shinshu University, Japan, in 2003. He is a Professor at Shinshu University. His research interests include evolutionary computation and learning, multidisciplinary design optimization, computational intelligence, and sustainability. He has published over 220 international journal and conference research papers on evolutionary algorithms, focusing on the working principles of single-, multi-, many-, and any-objective evolutionary optimizers, landscape analysis, epistasis, algorithm design, and real-world applications. He received the best paper award at GECCO in 2011, 2015, and 2020. His students have received the best student paper award at EMO 2019 and ECTA 2023. He served as the Editor-in-Chief for the Genetic and Evolutionary Computation Conference 2018 and as a Program Co-Chair for Parallel Problem Solving from Nature 2022. He serves on the ACM SIGEVO Executive Board, is Associate Editor of the ACM Transactions on Evolutionary Computation and Learning, and is a member of the Editorial Board of the Evolutionary Computation Journal, MIT Press. He is a member of ACM, IEEE, IEICE, and IPSJ.

Kiyoshi TANAKA (*Honorary member*)

He received his B.S and M.S. degrees in Electrical Engineering and Operations Research from the National Defense Academy, Yokosuka, Japan, in 1984 and 1989, respectively. In 1992, he received the Dr. Eng. Degree from Keio University, Tokyo, Japan. In 1995, he joined the Department of Electrical and Electronic Engineering, Faculty of Engineering, Shinshu University, Nagano, Japan, and currently, he is a full professor in the Academic Assembly (Institute of Engineering) of Shinshu University. He is the former Vice-president of Shinshu University in charge of international affairs. His research interests include image and video processing, 3D point cloud processing, information hiding, human visual perception, evolutionary computation, multi-objective optimization, smart grid, and their applications. He is a honorary member and a fellow of IIEEJ, a member of IEEE, IEICE, IPSJ, JSEC, and so on.

IIEEJ Paper -- Special Issue on Journal Track Papers in IEVC2024 --

VigNet: Semiautomatic Generation of Vignette Illustrations from Video

Mayu NAMAI†, Issei FUJISHIRO† (*Honorary Member*)

† Keio University

<**Summary**> A variety of summarization techniques has recently been proposed to manage the growing volume of media data, but most are oriented toward homogeneous media conversion, resulting in a limited compression ratio. In this study, we focus on the creation of a vignette illustration that represents the story in an animation or game briefly and that allows the viewer to understand its world perspective at a glance. If video can be converted into vignette illustration, it is expected to provide a more highly compressed summary of the media information. This paper proposes VigNet, a system that semiautomatically converts an input video into vignette illustrations so they reflect users' preferences.

Keywords: video summarization, media conversion, machine learning, world perspective visualization

1. Introduction

A variety of summarization techniques has been proposed to manage the growing volume of media data[1]. However, most approaches are aimed at compressing information of the same modality, resulting in a limited compression rate. To achieve an efficient summarization, it is essential to explore the potential of heterogeneous media conversion. Thus, this study focuses on vignette illustrations, miniature diorama-like illustrations that represent the condensed story content of animated films and games for entertainment purposes. The modality of illustration remains visual, but it clearly differs from video, as shown in **Fig. 1**.

As a result of our own preliminary analysis of approximately 20,000 examples on Pinterest[11] and other related websites, we found that a vignette illustration should consist of the following five elements:

- Characters: The main character and others
- Stage: Base for positioning characters
- Background: Background scene with vague outlines
- Supporters: Objects supporting characters
- Effects: Visual effects that enhance characters

Figure 1 gives an example of a vignette illustration and the five elements. In vignette illustrations, the characters and stage are essential elements, while the other three are ancillary elements that may be combined as needed. Furthermore, a typical characteristic of vignette illustrations is that they often depict the full-body images of characters. In addition, it is a notable fact that some parts of the background may be omitted.

The conversion from an input video to vignette illustrations may be considered one of the most important topics in visual intelligence. Most existing video summarization techniques focus primarily on selecting frames, whereas the synthesis of vignette illustrations requires not only the selection of frames but also the placement of elements, with subtle adjustments to their color tones and more, which is far more challenging. Generative models that convert natural language expressions to images have recently emerged, but only with conventional generative models, making it hard to generate vignette illustrations adaptively that arrange viewers' target contents from the video. While large language models are excellent at generating a variety of images, they are ineffective at generating consistent images robustly because they are susceptible to slight changes in input parameters and training data. Therefore, generating vignette illustrations from video requires a judicious combination of various visual computing methods, including video processing and image synthesis.

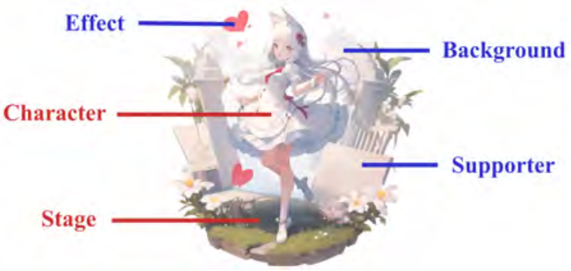


Fig. 1 An example of a vignette illustration with the five elements. This example includes all elements, but some vignette illustrations may not include the Background, Supporters, or Effects.

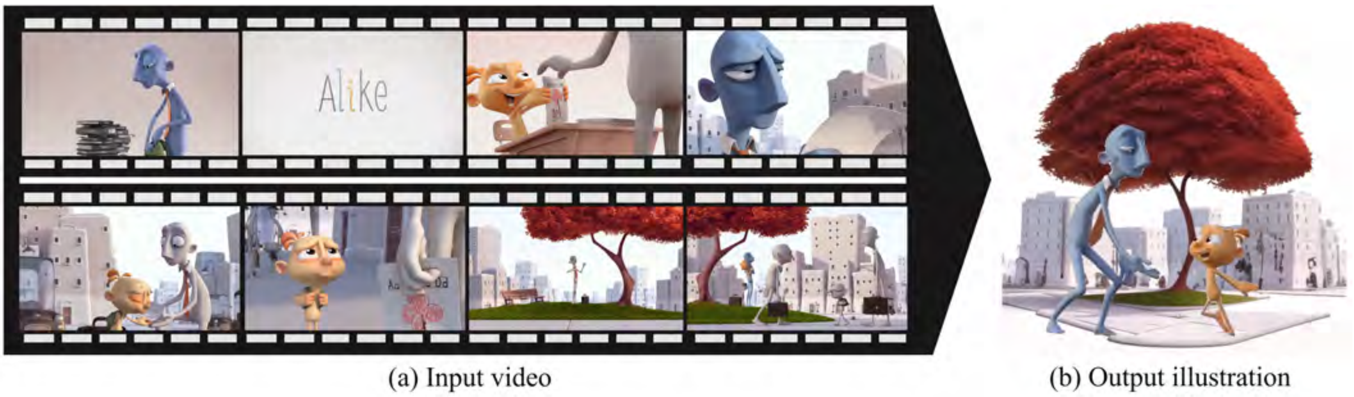


Fig. 2 Example execution of the VigNet system. Input video[6] (a) is semiautomatically converted to an output illustration (b)

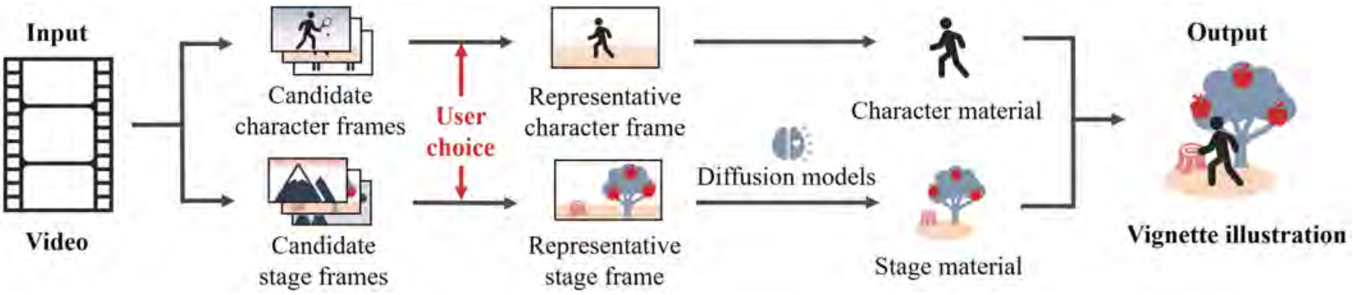


Fig. 3 VigNet processing framework

This paper proposes a system called video image generative network (VigNet), which generates semiautomatically vignette illustrations from a video, reflecting user preferences by means of semantic segmentation and the additional learning of image synthesis models. An example of the system's execution is shown in **Fig. 2**.

2. Related Work

Composable diffusion[14] is one of the latest generative models for heterogeneous media conversion, enabling various combinations of different media formats. By mediating the noise between different input and output media, it can handle many combinations of input and output modalities. However, the paper, which focuses on composable diffusion models, does not consider conversion from video to static images, nor does it provide any examples of results for this specific type of conversion.

Meanwhile, NewsThumbnail[8] is a system that generates automatic thumbnails from news videos by screening contents that are semantically similar to the user's query and combining them. However, its results are displayed only in the form of multiple snapshots arranged with keywords, so NewsThumbnail does not have as high a condensed impression as VigNet.

To the best of our knowledge, there are no known reported studies on the conversion of a video to a single still image that is not merely an enumeration of the contents.

3. System

As shown in **Fig. 3**, the processing framework of VigNet is based on the preliminary analysis mentioned in Chapter 1. Note in the current framework that non-character elements are aggregated into a stage without distinguishing among them. Hereafter, the image area for characters, or the stage, is referred to as the character or stage material, respectively. Both materials are generated independently in the process outlined in Section 3.1 and Section 3.2 and are combined in the process of Section 3.3 to yield a vignette illustration. In addition, **Fig. 4** and **Fig. 5** use the video *Alike*[6] to illustrate the intermediate processes of the system.

3.1 Character material generation

We refer to frames capturing the full bodies of characters as **candidate character frames**, which are extracted using the following procedure. First, for frames extracted at regular intervals from the input video, we use the Grounded Segment Anything Model (Grounded SAM)[4][9] to detect characters in the frames by specifying "character" as the prompt. Then, only frames in which detected bounding boxes do not touch the image boundary are selected, because vignette illustrations typically portray the full bodies of characters. In addition, frames in which characters overlap with other objects are manually excluded, by which process we obtain candidate character frames, as shown in Fig. 4(a).

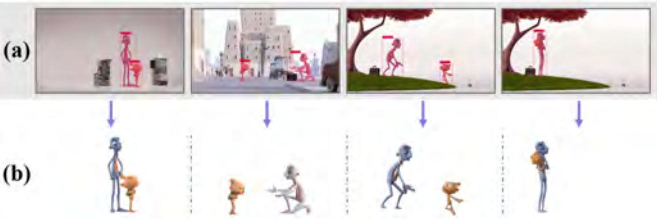


Fig. 4 Character material extraction. (a) Candidate character frame examples and (b) Clipped results

Second, candidate character frames are displayed to users, where each detected character is enclosed in a red frame of the Grounded SAM. Then, the users are prompted to select their favorite frame, referred to hereafter as the **representative character frame**.

Third, the **character material** is extracted from the representative character frame using the mask generated by the Grounded SAM, the actual results of which are shown in Fig. 4(b). If there is more than one character in the frame, the relative positions of the characters in the original frame are also stored and referenced, as will be shown in Section 3.3. This relative positioning effectively reflects the relationships between characters. In many cases, the distance between characters within a frame image represents their psychological distance. When there is a problem in their relationship, characters are often depicted physically apart in the frame. Conversely, when characters harbor intimate feelings for each other, they are typically shown close together. It can be seen in the animation *Alike*, as illustrated in Fig. 4, that characters stand apart in scenes of disagreement (second from the left), and are remarkably close in scenes of agreement (fourth from the left).

3.2 Stage material generation

We refer to frames capturing stages as **candidate stage frames**, and they are extracted by the following procedure. First, by comparing the average absolute error of consecutive frame images, cuts in the input video are detected. Then, a frame located in the middle of each detected cut is extracted. For these frames, the Grounded SAM is once again applied to detect characters and to select only frames in which a relatively small proportion of the scene is occupied by the characters. Next, captions for each frame are obtained using the BILP Image Captioning Large Model[7]. From these captions, frames in which the stage is mentioned are selected using the GPT-3.5-turbo model[10] of the ChatGPT API. Through this process, we obtain candidate stage frames, as shown in Fig. 5(a).

We detail a method for leveraging ChatGPT to determine whether a caption describes the background context. This procedure involves the input of concatenated text, consisting of a specific prompt and the caption to be assessed, into the API. The content of the prompt is, “I will input the text now. Please respond only with the number. In the following texts, answer 1 if the first noun is a character, such as a person or animal, and otherwise answer 2.” Based on the instruction, if the response returned from the ChatGPT API is ‘2,’ it is determined that the caption pertains to the background context. Initially, the accuracy of a single assessment by the ChatGPT API was approximately 80%. To achieve higher accuracy, we conducted 10 rounds of evaluations, designating the final candidate stage frame only if the caption was judged as ‘2’ five or more times. This approach improved the accuracy up to about 90%. However, there remains room for exploration to refine the prompts entered into ChatGPT and improve the accuracy of this method.

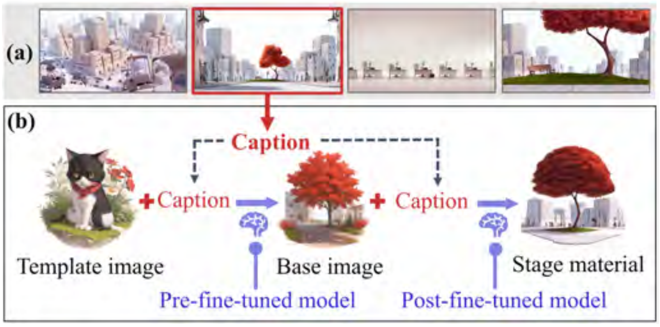
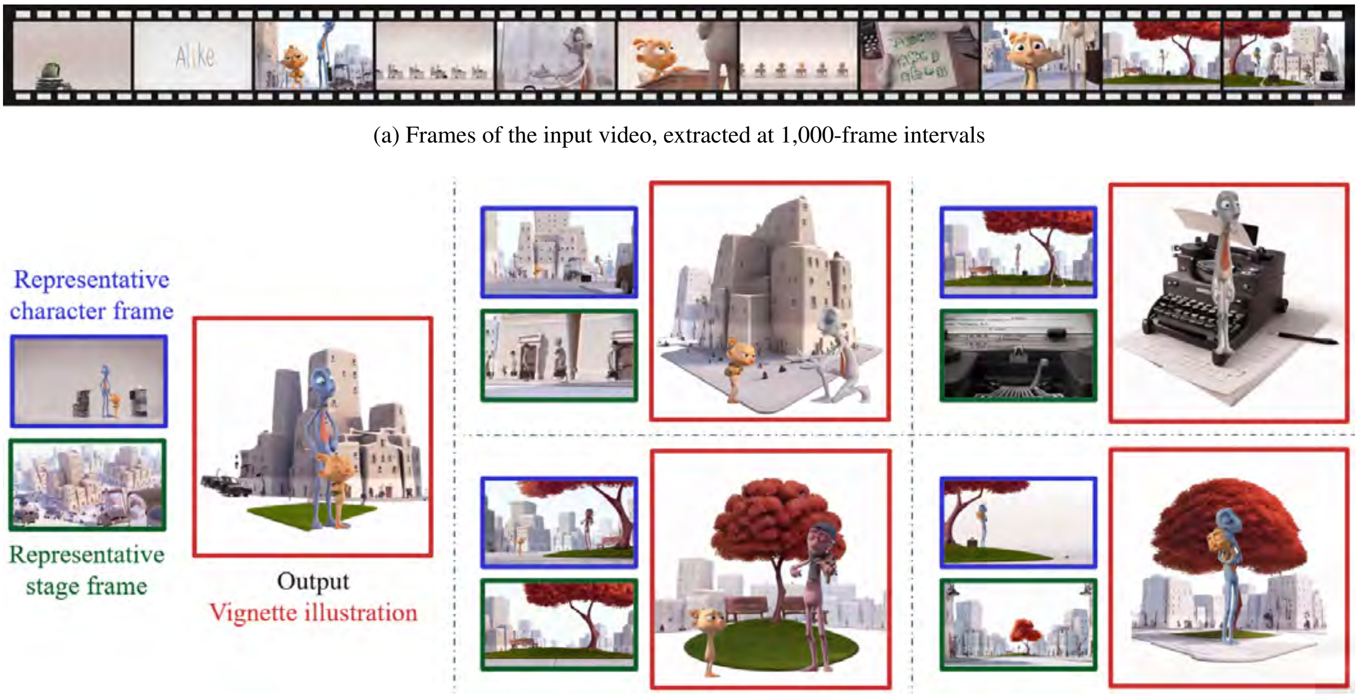


Fig. 5 Stage material generation. (a) Candidate stage frame examples and (b) Process of converting a representative stage frame into stage material

Second, candidate stage frames are presented to users for selection of the **representative stage frame**, defined as the frame the users find most appealing. If the second candidate in Fig. 5(a) is selected as the representative stage frame, the processing described later is executed based on the frame, as shown in Fig. 5(b).

Third, a stable diffusion model[12] is additionally trained by Low-Rank Adaptation (LoRA)[3] using training data frames extracted from an input video at regular intervals. We also employed the Counterfeit-V2.5 model[2], which was tailored to produce anime-style images, as the pre-fine-tuned model.

Fourth, with the following stable diffusion models, **stage material** is generated from the representative stage frames via two-step img2img, as shown in Fig. 5(b). In the first step, using the pre-fine-tuned model, an image is generated by taking as inputs the caption of a representative stage frame and an existing vignette illustration to use as a template image. The output image is currently defined as the base image. In the second step, the same caption and base image are used as inputs to generate the stage material using the post-fine-tuned model. In the initial phase of generation, the base image may fail to



(a) Frames of the input video, extracted at 1,000-frame intervals

(b) Representative frames and output results. The representative character frames are enclosed in blue, the representative stage frames in green, and the vignette illustrations generated by their combination are encompassed in red

Fig. 6 Frame images and examples of output results with *Alike*[6] as the input video

capture the color scheme and ambience of the video. However, in the subsequent phase, the stage materials accurately reflect the video's color palette and atmosphere.

3.3 Image synthesis

Initially, the character material is automatically resized to ensure appropriate scaling within the vignette illustrations. As described in Chapter 1, the preliminary analysis revealed that the number of characters typically present in a single vignette illustration ranges from one to two. Furthermore, it was empirically determined that the total area occupied by the characters in an illustration should be approximately 0.1 × the number of characters. Based on this empirical rule, the character assets were resized accordingly.

Second, the Grounded SAM was used to detect the characters' foothold on the stage material and to place the characters by specifying "floor surface" as the prompt. If one character is present, they are placed at the center; otherwise, they are superimposed on the stage material, reflecting their position in the representative character frame. In this way, the character(s) and stage material are merged into a vignette illustration.

4. Results

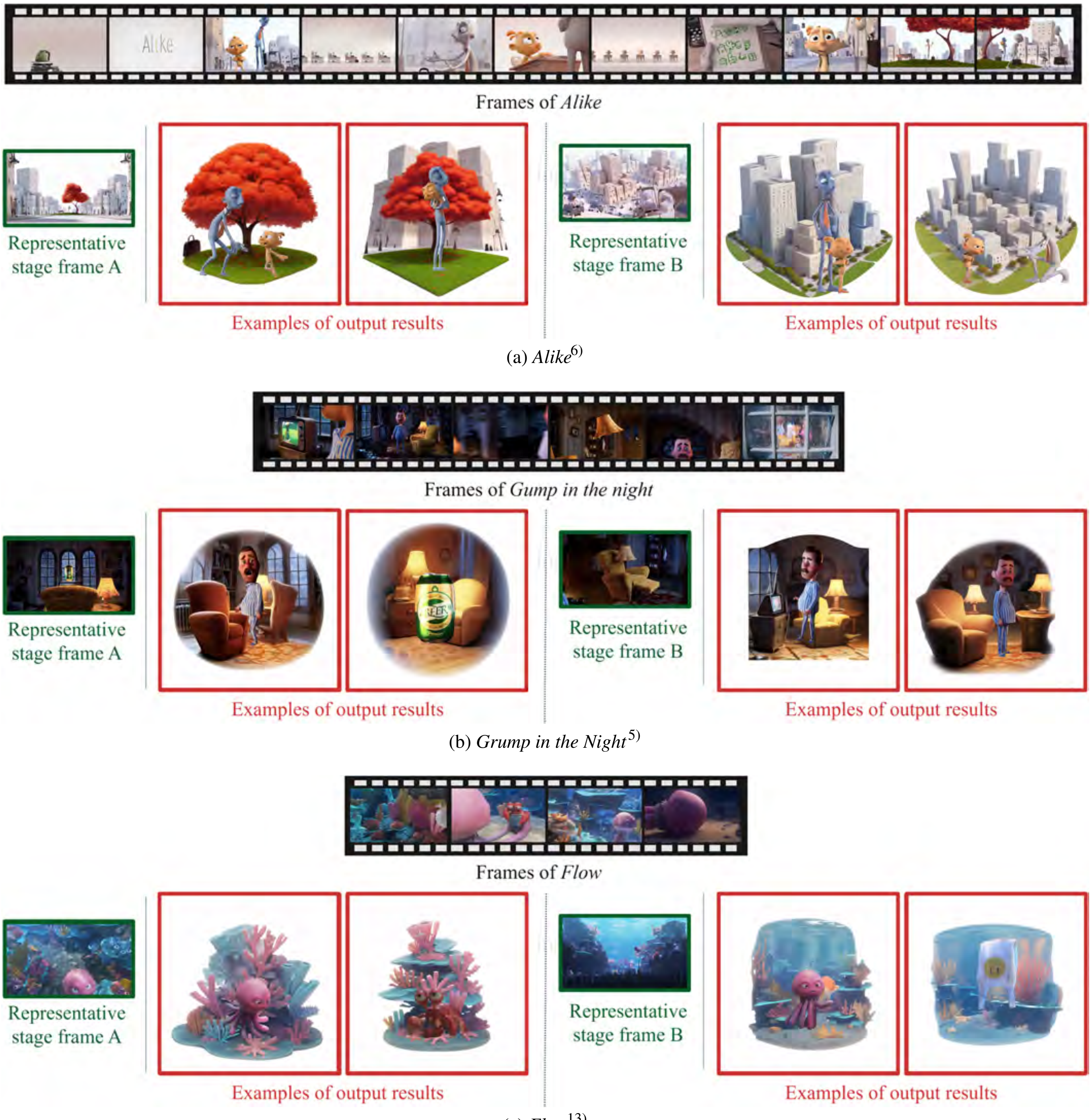
We used a personal computer with CPU: 12th Gen Intel(R) Core(TM) i9-12900H @2.30GHz, RAM: 64 GB, GPU: NVIDIA GeForce RTX 3080 Ti, and Python 3.8.16 to implement the current prototype system of VigNet. For the experiments, we exclusively used modifiable CG animation videos that are licensed under Creative Commons license to ensure their appropriateness for publication. As vignette illustrations are often derived from anime and game artworks, we evaluated whether VigNet could generate vignette illustrations from actual anime productions.

Table 1 Detailed statistics for additional training. M4.1 is defined as the model used in Section 4.1, and M4.2a, M4.2b, and M4.2c as the models used in Section 4.2. The video (a) is *Alike*[6], (b) *Grump in the Night*[5], and (c) *Flow*[13]

Model	M4.1	M4.2a	M4.2b	M4.2c
Video	(a)	(a)	(b)	(c)
#learning frames	1,021	103	61	50
#total learning steps	204,200	20,600	12,200	10,000
Training time	24h 17m	3h 31m	1h 50m	1h 41m
Loss	0.0851	0.103	0.14	0.146

4.1 Execution example

The short movie *Alike*[6] was used as an input to the system, some frames of which are shown in **Fig. 6**(a). The work consists of 10,208 frames in total, and it is 6 minutes and 48 seconds long. We prepared a model trained with frame images extracted at 10-frame intervals from the input video, which


Alike
Frames of Alike
Representative stage frame A
Examples of output results
Representative stage frame B
Examples of output results
Frames of Gump in the night
Representative stage frame A
Examples of output results
Representative stage frame B
Examples of output results
Frames of Flow
Representative stage frame A
Examples of output results
Representative stage frame B
Examples of output results

(a) *Alike*[6)]

(b) *Grump in the Night*[5)]

(c) *Flow*[13)]

Fig. 7 Frames of input videos, and examples of representative stage frames and their output results. Frames of input videos are extracted at 1,000-frame intervals

took 24 hours and 17 minutes to train. The parameters include a repetition count of 20, epoch number of 10, training batch size of 10, network rank of 8, and no normalized images. The detailed statistics of the additional training are shown in **Table 1**. The procedure in Section 3.1 automatically extracted 41 frame images, 13 of which were manually extracted as candidate character frames, and the procedure in Section 3.2 resulted in 10 candidate stage frames. Meanwhile, candidate character frames were extracted manually, as opposed to the candidate stage frames, which were extracted automatically. Thus, the number of intermediate frames did not confuse the user when making their choice. Several representative frames and output results are shown in Fig. 6(b), and the five output results cover the user's selection of representative frames from the candidate frames in various combinations. In all cases, output illustrations were generated so they would convey the features of the frame images selected by the first author and summarize the content of an input video.

For VigNet to function effectively with input videos, we consider two necessary conditions. The first is the inclusion of frames that display the full body of the character without any other objects overlapping it in the foreground. The second is the inclusion of candidate stage frames in which characters do not appear. At present, the system requires the manual selection of candidate character frames and representative frames by the user, but we have a plan to make it fully automatic in the future.

Even when live-action videos are input into VigNet, if the above two necessary conditions are met, VigNet is expected to function effectively. The object detection feature of Grounded SAM used in VigNet has proven effective not only with animated images but also with live-action images. When using live-action videos as input, it may be necessary to adjust the prompts slightly for object detection, but otherwise, it is anticipated that vignette-style live-action images would be generated. Furthermore, some of the stable diffusion models used for image generation excel at producing photorealistic images. Therefore, using a model specialized in generating realistic images rather than the Counterfeit-V2.5 model used in this study might yield better results.

4.2 Sensitivity analysis

We investigated how changes to an input video affect the output illustrations. In addition to *Alike*, two other videos, i.e., *Grump in the Night*[5)] by Kris Theorin, Somethings Awry Production and *Flow*[13)] by The Animation School, were selected as inputs. **Table 2** compares the three videos in terms of several features. The parameters are consistent across all cases, with a repetition count of 20, epoch number of 10, training batch size of 10, Network Rank of 8, and no normalized images. The detailed statistics of the additional training are shown in Table 1.

Alike contains both indoor and outdoor scenes, while *Grump in the Night* and *Flow* are composed mostly of indoor scenes and outdoor scenes, respectively. As a dataset for the learning model, frame images were commonly extracted from each video at 100-frame intervals. For image generation, denoising strength was set as follows: when using img2img to generate base images, the denoising strength was set to 0.75. Conversely, for the generation of stage materials with img2img, the strength was adjusted to 0.90. These settings were adopted as a consistent criterion for image generation within our sensitivity analysis.

Table 2 Features of input videos. (a) is *Alike*[6)], (b) is *Grump in the Night*[5)], (c) is *Flow*[13)]

Title	(a)	(b)	(c)
Duration	6m 48s	4m 13s	3m 24s
#total frames	10,208	6,096	4,924
#cuts	76	103	16
Main scenes	indoor & outdoor	indoor	outdoor
#characters	3	4	3

Figure 7 gives frames of input videos and examples of two representative stage frames based on which two vignette illustrations were generated. The visual representation of the stage materials varies significantly depending on the chosen representative stage frame. Furthermore, even when using the same representative stage frame, by merely adjusting the seed value in img2img, it becomes evident that stage materials with similar yet distinct appearances can be generated.

In the current version of VigNet, elements other than characters are aggregated into the stage. Indeed, Fig. 7(b) extracts the room as a background, while Fig. 7(c) contains coral as a supporter. Because the input videos used in this experiment did not contain any effects, it follows naturally that the output illustrations also lacked effects. Moreover, it was easier to generate a higher quality stage material from a video such as *Flow*, with sequences of similar supporters, than from another

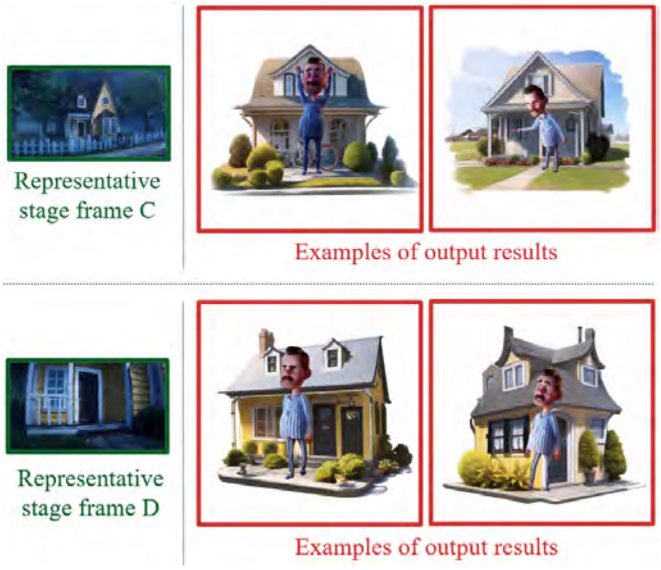

Representative stage frame C
Examples of output results
Representative stage frame D
Examples of output results

Fig. 8 Examples of Failures

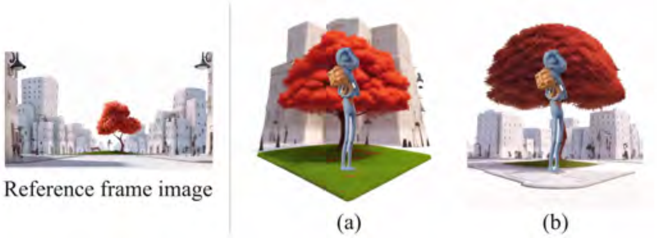


Fig. 9 Examples output by VigNet for different amounts of model training data. (a) was generated using a model trained on 103 frame images, while (b) was generated with a model trained on 1,021 frame images

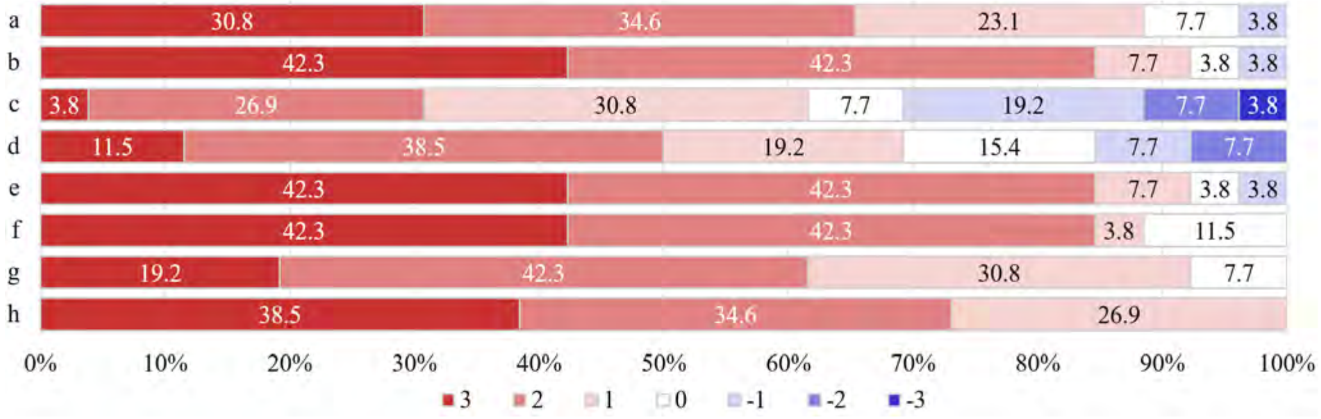


Fig. 10 Results of questionnaire

one, such as *Grump in the Night*, which contains many different supporters. Another reason for the low quality of the generated illustrations in *Grump in the Night* may be the high frequency of cuts.

Moreover, it was found that an insufficient number of images in the training dataset failed to generate high-quality stage materials. **Figure 8** shows two resulting illustrations that use stage materials based on two outdoor frames from *Grump in the Night*. Because most of this video is composed of indoor scenes, when attempting to generate outdoor scenes, the training data were insufficient, resulting in stage materials that hardly reflect the appearance of the house or weather conditions.

Conversely, it was found that as the amount of image data for training increased, VigNet was able to generate high-quality stage materials. **Figure 9** compares the output results of VigNet when the amount of training data for the model varies. Figure 9(a) was generated using a model trained on 103 frame images, while Fig. 9(b) was generated with a model trained on 1,021 frame images. Upon comparison, it is obvious that (b) reproduces the shape and color of the objects in the input video more accurately. In fact, the model that produced (a) exhibited a loss of 0.103, whereas the model that generated (b) demonstrated a lower loss of 0.0851, indicating that the latter achieved a smaller loss value.

4.3 Evaluation experiments

We conducted an evaluation experiment concerning the output results for *Alike* as the input video. For a rewarded viewer evaluation, we collected 26 non-professional participants (50–50 split between genders). The participants ranged in age from their late teens to late twenties. As a reward, each was offered snacks and beverages. The procedure for the evaluation experiment is as follows.

1. We let them freely watch the movie, *Alike*, and then asked them to imagine their own vignette illustration.
2. After briefly introducing the process flow of the VigNet system, we asked each to choose two representative frames from the candidate frames.
3. We presented each with VigNet's output illustrations corresponding to their selected combinations.
4. We asked them to answer a questionnaire to evaluate the output illustrations while comparing with the vignette illustrations they imagined.

Table 3 Questionnaire items

a	Did any of the candidate character frames include the characters you imagined?
b	Did any of the candidate stage frames include the background you imagined?
c	How much of a difference is there between your imagined vignette illustration and the output illustration?
d	Under the assumption that the quality of your imagined illustration is 0, what is the quality of the output illustration?
e	Do you think the output image is a vignette illustration?
f	Which reminds you more of the content of the video, *Alike*'s YouTube thumbnail, as shown in **Fig. 11**, or the output image?
g	Does the output image condense the essence of what you consider important in the video?
h	Does the output image express the world perspective of the video?

Fig. 11 *Alike*'s YouTube thumbnail

Table 3 lists eight questions asked in the questionnaire, and **Fig. 10** shows the results of each. The participants in the evaluation experiment were asked to answer the questions on a 7-point scale from +3 to -3. Positive (+3 to +1) and negative (-3 to -1) results are indicated by the reddish and bluish color gradations, respectively, except for neutral evaluations (0), indicated in white.

As a result, 60% of participants answered that the results generated by VigNet were close to their imagined vignette illustrations, more than 90% stated the results summarized the video well, and all participants reported that the results expressed the world perspective of the video.

Many participants commented that they would like various supporters, including the violin, the stacked books, and the character's bag, to appear in an output illustration. Each of the participants had different ideas of what elements should be included in the output image. Several participants specifically mentioned that if such supporters could be added to the VigNet output, the result would be extremely close to their own vignette illustrations. This suggests that VigNet's results reflect common ideas shared among participants.

Based on this feedback, we would like to enable VigNet to reflect users' preferences by placing supporters explicitly in its outputs. We also have a plan to conduct additional experiments to evaluate the system's usability after a careful redesign of the system's user interface. Furthermore, future studies will compare the outputs generated by VigNet with vignette illustration drawn by professional artists.

5. Conclusion

Regardless of the kind of metaverse realized, the value of a still image that provides a bird's-eye view of the world will remain unchanged. In this paper, therefore, we proposed the VigNet system for generating vignette illustrations semiautomatically from computer graphics animation videos by combining semantic segmentation and fine-tuned image synthesis models. The evaluation experiments conducted empirically proved that the prototype system of VigNet can robustly visualize the world perspective of videos on a small scale, while reflecting users' preferences.

Acknowledgments

This work has been in part supported by the JSPS KAKENHI Grant-in-Aid for Challenging Research (Exploratory) Grant Number 23K18468 and the Research Grant of Keio Leading-edge Laboratory of Science & Technology.

References

1) T. Baltrušaitis *et al.*, "Multimodal machine learning: A survey and taxonomy," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 41, No. 2, pp. 423–443, 2019. DOI:10.1109/TPAMI.2018.2798607

2) gdgsfsfs, *Hugging Face, Counterfeit-V2.5*, https://huggingface.co/gsdf/Counterfeit-V2.5, latest access 31/03/2024.

3) E. J. Hu *et al.*, "LoRA: Low-rank adaptation of large language models," 2021. https://arxiv.org/abs/2106.09685

4) A. Kirillov *et al.*, "Segment anything," 2023. https://arxiv.org/abs/2304.02643

5) Somethings Awry Production Kris Theorin, *Grump in the Night*, 2022, https://www.youtube.com/watch?v=ystQWn3K2GY, License: https://creativecommons.org/licenses/by/4.0/, latest access 02/04/2024.

6) Daniel Martínez Lara and Rafa Cano Méndez, *Alike*, 2017, https://www.youtube.com/watch?v=PDHIyrfMl_U, License: https://creativecommons.org/licenses/by/4.0/, latest access 02/04/2024.

7) J. Li *et al.*, "BLIP: Bootstrapping language-image pre-training for unified vision-language understanding and generation," in *Proceedings of the 39th International Conference on Machine Learning, PMLR*, Vol. 162, pp. 12,888–12,900, 2022.

8) J. Li *et al.*, "Newsthumbnail: Automatic generation of news video thumbnail," in *2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 1383–1388, 2022. DOI: 10.1109/SMC53654.2022.9945444

9) S. Liu *et al.*, "Grounding DINO: Marrying DINO with grounded pre-training for open-set object detection," 2023. https://arxiv.org/abs/2303.05499

10) OpenAI, *ChatGPT*, https://chat.openai.com/, latest access 02/04/2024.

11) Pinterest, Inc., *Pinterest, Vignette Illustration*, https://www.pinterest.jp/search/pins/?rs=ac&len=2&q=vignette%20illustration&eq=vignette%20illu&etslf=11462, latest access 02/04/2024.

12) R. Rombach *et al.*, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10:684–10:695, June 2022.

13) The Animation School, *Flow*, 2023, https://www.youtube.com/watch?v=rO87HFsmfis, License: https://creativecommons.org/licenses/by/4.0/, latest access 02/04/2024.

14) Z. Tang *et al.*, "Any-to-any generation via composable diffusion," 2023. https://arxiv.org/abs/2305.11846

(Received November. 27, 2023)

Mayu NAMAI
She is currently a master's student in the School of Science for Open and Environmental Systems, Graduate School of Science and Technology, Keio University. She received her bachelor's degree in Information and Computer Science from Keio University in 2023. Her major research interest includes semiautomatic generation of vignette illustrations from videos.

Issei FUJISHIRO (*Honorary Member*)
He is currently head professor of the Center for Information and Computer Science at Keio University, Yokohama, Japan. He received his Doctor of Science in information sciences from the University of Tokyo in 1988. His research interests include modeling paradigms and shape representations, applied visualization design and lifecycle management, and smart ambient media with multimodal displays. He is an associate member of Science Council of Japan, a fellow of JFES and IPSJ, and a senior member of IEEE and ACM. He is a 2021 inductee to the IEEE Visualization Academy.

IIEEJ Paper -- Special Issue on Journal Track Papers in IEVC2024 --

Adaptive Selection of Auxiliary Tasks Using Deep Reinforcement Learning for Video Game Strategy

Hidenori ITAYA† (*Student Member*), Tsubasa HIRAKAWA†, Takayoshi YAMASHITA†, Hironobu FUJIYOSHI†

† Chubu University

<**Summary**> Multitask learning can be utilized to efficiently acquire common factors and useful features among several different tasks. This learning method has been applied in various fields because it can improve the performance of a model by solving related tasks with a single model. One type of multitask learning utilizes auxiliary tasks, which improves the performance of the target task by learning auxiliary tasks simultaneously. In the video game strategy task, unsupervised reinforcement learning and auxiliary learning (UNREAL) has achieved a high performance in a maze game by introducing an auxiliary task. However, in this method, the auxiliary task must be appropriate for the target task, which is very difficult to determine in advance because the most effective auxiliary task will change dynamically in accordance with the learning status of the target task. Therefore, we propose an adaptive selection mechanism called auxiliary selection for auxiliary tasks based on deep reinforcement learning. We applied our method to UNREAL and experimentally confirmed its effectiveness in a variety of video games.

Keywords: deep learning, auxiliary task, reinforcement learning

1. Introduction

Real-world problems are complex mixtures of various elements, and even seemingly disparate tasks can be closely related. Solving these different tasks simultaneously with a single model can improve the model performance and reduce training time. This learning method, known as multitask learning, can efficiently acquire factors and useful features that are common to several different tasks. Research in the field of image recognition has shown that model performance can be improved by learning class classification, object detection, segmentation, *etc.* simultaneously[1)–4)]. In the field of natural language, training part-of-speech classification, sentence segmentation, and sentence relation values simultaneously is also known to be effective[5),6)].

Auxiliary learning is a type of multitask learning that improves the performance of the main task as a kind of normalization by adding auxiliary tasks unrelated to the main task to be solved to the learning target task and by learning the auxiliary tasks at the same time. In automated driving, the accuracy of the main task can be improved by utilizing depth estimation and semantic segmentation from in-vehicle camera images as the main task and introducing time and weather estimation as the auxiliary tasks[7)]. In video games, game scores have been improved by introducing three different auxiliary tasks to the main task, which is a game strategy based on deep reinforcement learning (DRL)[8)]. The auxiliary tasks in these studies were designed to be tasks that were unrelated to the main task but that were intuitively thought to contribute to the main task. For example, the domain of an in-vehicle camera image is highly dependent on the time of day and the weather conditions. Since auxiliary tasks are designed manually, they do not necessarily contribute to solving various main tasks. On the contrary, some of them may actually interfere with learning, depending on the main task. While this problem could potentially be solved by carefully designing auxiliary tasks that depend on the main task, it is expensive to verify the effectiveness of auxiliary tasks. In learning methods that introduce these auxiliary tasks, the loss function is a weighted sum of each task, including the main task. Therefore, the weight of each task, i.e., the ratio of the mixing gradients for each task, is important for learning. However, it is difficult to set the optimal weights in advance because they depend on the nature of the main task and the training stage.

In response to these challenges, the purpose of our current study is to adaptively select an auxiliary task suitable for the main task. In this paper, we propose a new auxiliary task selection mechanism, called auxiliary selection, which adaptively selects auxiliary tasks according to the main task. Specifically, it is designed as a DRL agent that outputs weights for each auxiliary loss and adaptively selects auxiliary tasks for network updates on the basis of the main task. The auxiliary selection model is trained simultaneously with other models and shares the reward signal with the main task to find the appropriate auxiliary task according to the training stage of the main task. We apply auxiliary selection to UNREAL, a method that introduces auxiliary tasks into video game strategy, and analyze the selected auxiliary tasks and game scores to determine the selection of suitable auxiliary tasks for the main task.

The contributions of our study are as follows.

- Our method efficiently improves the performance of the main task by utilizing deep reinforcement learning to select the optimal auxiliary task according to the training stage and scene.
- Our method automatically suppresses unnecessary auxiliary tasks and prevents the main task from losing accuracy, thus reducing the cost of designing auxiliary tasks.

2. Related Works

2.1 Improving the performance of the main task by introducing auxiliary tasks

In the field of multitask learning, many investigated have studied how to improve the performance of the main task by introducing auxiliary tasks. These studies are described below, with a focus on supervised learning and deep reinforcement learning.

In the field of supervised learning, Liebel *et al.* introduced semantic segmentation and depth estimation as main tasks in automated driving and auxiliary tasks for time and weather estimation[7]. They reported that the auxiliary tasks of time and weather estimation contributed to improving the accuracy of the main task. Zhang *et al.* achieved more robust face landmark detection by simultaneously optimizing the main task of face landmark detection and the auxiliary tasks of face pose and attribute estimation[9].

In the field of deep reinforcement learning, Mirowski *et al.* focused on navigation tasks in 3D environments and proposed an auxiliary task, depth prediction, which predicts depth information from RGB images[10]. A comparison under various conditions, such as the use of depth information as input and the position of the auxiliary task in the network structure, verified the effectiveness of the depth prediction. Kartal *et al.* focused on the terminal state of the environment and proposed an auxiliary task called terminal prediction (TP) to predict how close the current state is to the terminal state of the environment[11]. Hernandez-Leal *et al.* focused on the behavior of other agents in multi-agent problems and proposed agent-modeling, an auxiliary task to predict the behavior of other agents[12]. Experiments on cooperative and competitive tasks of multi-agent problems showed that the agent-modeling contributed to improving the stability of learning, thereby improving the accuracy in each task.

In the DRL field, there has been extensive research on video game strategy[13], and the improvement of game scores by using auxiliary tasks has been reported. Jaderberg *et al.* proposed unsupervised reinforcement learning and auxiliary learning (UNREAL), in which an unsupervised auxiliary task is learned simultaneously with the main task in deep reinforcement learning[8]. The main task of this method is a video game strategy from images (game screen frames) using asynchronous advantage actor-critic (A3C)[14]. Thanks to solving the auxiliary task by using part of the main task model and learning with a weighted sum of the main and auxiliary losses, the auxiliary task improves the video game score. The authors used three different auxiliary tasks in their study. The first is pixel control, which learns the agent's behavior such that pixels in the input image are changed. This task is based on the idea that visual changes in the environment are associated with important events. In this auxiliary task, the input image is divided into $n \times n$ grids, and the agent learns an action value $Q^{(c)}$ to maximize the pixel value change in each grid. Here, c denotes the number of grids into which the input image is divided. The second auxiliary task is value function replay, which shuffles past experiences to learn the state value function $V(s)$. This task learns state values using experience replay[15], an effective mechanism for improving both the data efficiency and stability of the DRL algorithm. In other words, it is equivalent to performing state value function regression with an auxiliary task in addition to state value function regression with A3C, which promotes optimal state value estimation. The third auxiliary task is reward prediction, which predicts the reward to be obtained in an unknown state. In DRL, learning an agent's behavior requires accurate recognition of the states that give high rewards. However, reward acquisition is very

Table 1 Comparison to proposed and conventional methods

Method	Target	How to select
Tel *et al.*[16]	Policy	Obtain common policy through distillation
Riedmiller *et al.*[17]	Policy	Adopt hierarchical policies
Du *et al.*[18] Lin *et al.*[19]	Task	Adopt gradient similarity between tasks
Ours	Task	Select auxiliary tasks by DRL

sparse in many environments, and it takes a long time to train a model that represents reward acquisition well. Therefore, by solving the task of predicting the reward that can be acquired in the unknown state that follows from sequential states, the task efficiently learns to recognize the states that contribute to reward acquisition. The learning of this auxiliary task is achieved by multi-class cross-entropy classification loss over three classes (zero, positive, negative) of rewards for the next state. By introducing these auxiliary tasks, UNREAL achieved high scores on the main task, which was the maze capture at DeepMind Lab.

2.2 Selection of auxiliary tasks

Auxiliary learning improves the performance of the main task, but if auxiliary tasks that are not suitable for learning the main task are used, they will interfere with the learning of the main task, resulting in a loss of accuracy. Therefore, it is necessary to introduce auxiliary tasks that are suitable for the main task. Several methods have been proposed to solve this problem.

Conventional methods can be divided into those that take an auxiliary policy approach and those that take an auxiliary task approach. A comparison of the proposed and conventional methods is provided in **Table 1**. The auxiliary policy approach constructs a number of policies with auxiliary purposes to the main task and selects the optimal policy from these auxiliary policies. The methods by Tel *et al.* and Riedmiller *et al.* are auxiliary policy approaches. Tel *et al.* use knowledge distillation[20]to obtain a common policy among several auxiliary policies with different roles[16], while Riedmiller *et al.* achieve auxiliary policy selection by adopting a hierarchical structure with several low-level auxiliary policies and a main policy that selects the optimal auxiliary policy[17]. The main policy focuses on a specific task, whereas the auxiliary policies represent actions under other goals and conditions. Therefore, it is easy to obtain feedback from the selection of auxiliary policies, as the agent's behavior changes significantly depending on the selected auxiliary policies. However, these methods are difficult to apply to tasks that cannot be subdivided into smaller policies, and the training cost is high because the construction of auxiliary policies requires individual training on a different target.

The auxiliary task approach solves a task with an auxiliary purpose at the same time as the main task and selects which of these auxiliary tasks to use during learning. The methods by Du *et al.* and Lin *et al.* are auxiliary task approaches. These two methods focus on the gradient of the auxiliary loss, which is the loss of the auxiliary task. Du *et al.* utilize the cosine similarity between the gradients[18], and Lin *et al.* use the Taylor approximation including the gradient of the auxiliary loss[19]to calculate the similarity between tasks. This similarity is then utilized to select which auxiliary losses to be used for training the main task. Since these methods can utilize auxiliary tasks suitable for the main task during learning, they can reduce the training cost compared to the auxiliary policy and contribute to improving the performance of the main task. However, the similarity of the auxiliary losses used to select the auxiliary tasks is very sensitive to the appropriate measure (e.g., how similar is it when to use auxiliay tasks, *etc.*), as it depends on the task and the training situation.

Our method, like those of Du *et al.* and Lin *et al.*, is an auxiliary task approach. We construct a DRL agent that dynamically controls the coefficient for the auxiliary loss, thereby achieving the auxiliary task selection according to the training status of the main task. Therefore, our method optimizes the aforementioned coefficients for auxiliary loss by DRL, which thus enables dynamic selection of auxiliary tasks.

3. Proposed Method

The effectiveness of the auxiliary task depends on the main task, and an inappropriate auxiliary task may prevent the learning of the main task. Therefore, we propose auxiliary selection, which adaptively selects the auxiliary task according to the main task. In this method, we apply our method to UNREAL, which has achieved a high performance score by introducing three auxiliary tasks in video game strategy, and extend it to a method that can learn efficiently on various video games. In other words, this method efficiently strategies video games by dynamically selecting auxiliary tasks according to the video game to be solved.

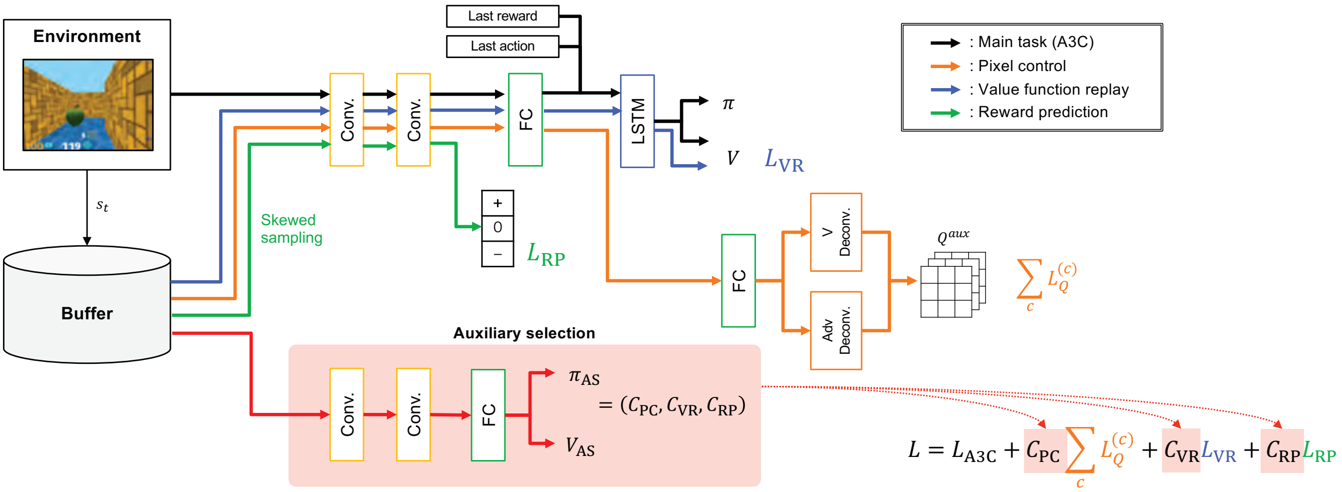


Fig. 1 Overview of proposed method. Here, pixel control, value function replay, and reward prediction are unsupervised auxiliary tasks proposed by Jaderberg *et al.*

3.1 Background

Similar to Jaderberg *et al.*[8], we assume a standard reinforcement learning setting in which the agent interacts with the environment over several discrete time steps. At time t, the agent receives observation o_t with reward r_t and generates action a_t. The agent's state s_t is a function of experience until time t, where $s_t = f(o_1, r_1, a_1, \cdots, o_t, r_t)$. The n-step return $R_{t:t+n}$ at time t is defined as the discounted sum of rewards $R_{t:t+n} = \sum_{i=1}^{n} \gamma^i r_{t+i}$. The value function is the expected return from the state s, $V_\pi(s) = \mathbb{E}[R_{t:\infty} | s_t = s, \pi]$.

In our study, the main task is a video game strategy, and the three UNREAL auxiliary tasks described in Section 2.1 are utilized as auxiliary tasks. The UNREAL auxiliary tasks are unsupervised auxiliary tasks that are not dependent on the main task. This makes it more difficult to manually select suitable auxiliary tasks for the main task in advance compared to supervised auxiliary tasks. In our method, we achieve the dynamic selection of auxiliary tasks by building a DRL agent that selects the most appropriate auxiliary task according to the video game to be solved. The main task is learned using Asynchronous Advantage Actor-Critic (A3C)[14]as well as UNREAL. The A3C algorithm constructs an approximation of the policy $\pi(a|s,\theta)$ and the state value function $V(s,\theta)$ using the model parameter θ. The policy and state value are adjusted toward the n-step lookahead value $R_{t:t+n} + \gamma V(s_{t+n+1}, \theta)$ using an entropy normalization penalty $L_{\mathrm{A3C}} \approx L_{\mathrm{VR}} + L_\pi - E_{s\sim\pi}[\alpha H(\pi(\cdot|s,\theta))]$, where $L_{\mathrm{VR}} = E_{s\sim\pi}\left[(R_{t:t+n} + \gamma^n V(s_{t+n+1}, \theta^-) - V(s_t, \theta))^2\right]$. In this equation, θ^- represents the model parameters before the model update.

3.2 Adaptive selection of auxiliary tasks

Figure 1 shows the network structure of our method. In this Section, we introduce auxiliary selection (AS), which selects the most appropriate auxiliary task for the video game task to be solved. AS takes an image stored in the replay buffer as input and outputs the state value $V_{\mathrm{AS}}(s)$ and the policy π_{AS}. The AS policy π_{AS} determines the binary weights for each auxiliary task, and indicates whether or not each auxiliary task is utilized for training the main task. The binary weights for each auxiliary task are defined as $C_{\mathrm{PC}} = \{0,1\}, C_{\mathrm{VR}} = \{0,1\}, C_{\mathrm{RP}} = \{0,1\}$. The AS policy π_{AS} is calculated as

$$\pi_{\mathrm{AS}} = (C_{\mathrm{PC}}, C_{\mathrm{VR}}, C_{\mathrm{RP}}). \tag{1}$$

AS consists of two convolutional layers and a fully-connected layer. Unlike other auxiliary tasks, the AS network is constructed without sharing weights with the main task network, and AS is trained independently of the main task. AS's policy is learned on the basis of the same reward as the main task. In other words, AS controls the binary weights of the auxiliary tasks to improve the score of the video games, which is the main task.

3.3 Loss function

We formulate the loss function for our method as

$$L = L_{\mathrm{A3C}} + C_{\mathrm{PC}} \sum_c L_Q^{(c)} + C_{\mathrm{VR}} L_{\mathrm{VR}} + C_{\mathrm{RP}} L_{\mathrm{RP}}. \tag{2}$$

where L_{A3C} is the loss of the main task (i.e., A3C) and $\sum_c L_Q^{(c)}$, L_{VR}, and L_{RP} are the losses of each auxiliary task described in Section 2.1. Pixel control learns individual policies to maximally change the pixels of each grid by dividing the input image into $n \times n$ grids. Thus, $L_Q^{(c)}$

Algorithm 1 Proposed method

```
//Assume global shared parameters $\theta$ and $\theta_{\mathrm{AS}}$ and global shared counter $T = 0$
//Assume worker-specific parameters $\theta'$ and $\theta'_{\mathrm{AS}}$
Initialize local step counter $t \leftarrow 1$
repeat
    Reset gradients: $d\theta \leftarrow 0$ and $d\theta_{\mathrm{AS}} \leftarrow 0$
    Synchronize worker-specific parameters $\theta' = \theta$ and $\theta'_{\mathrm{AS}} = \theta_{\mathrm{AS}}$
    $t_{start} = t$
    Get state $s_t$
    repeat
        Perform action $a_t$ according to policy $\pi(a_t|s_t, \theta')$
        Receive reward $r_t$ and new state $s_{t+1}$
        Store experience ($s_{t+1}$, $r_t$, $a_t$) in replay buffer
        $t \leftarrow t + 1$
        $T \leftarrow T + 1$
    until terminal $s_t$ or $t - t_{start} == t_{max}$
    Execute each auxiliary task with the experiences stored in replay buffer
    Execute auxiliary selection with the experiences stored in replay buffer
    Accumulate gradients $d\theta$ w.r.t. $\theta'$
    Accumulate gradients $d\theta_{\mathrm{AS}}$ w.r.t. $\theta'_{\mathrm{AS}}$
    Perform asynchronous update of $\theta$ using $d\theta$ and of $\theta_{\mathrm{AS}}$ using $d\theta_{\mathrm{AS}}$
until $T > T_{max}$
```

is the loss of PC, which is the loss of n-step Q-learning for grid c. Our method achieves auxiliary task selection by multiplying the loss of each auxiliary task used to update the network parameters by the binary weights obtained in AS. As shown in Eq. (2), the binary weights C_{PC}, C_{VR}, and C_{RP} of AS are included in the loss function L of this method. Therefore, the learning of the AS network using the loss function L is performed so that the loss is close to zero, including the AS outputs C_{VR}, C_{PC}, and C_{RP}. That is, AS is learned so that $C_{\mathrm{VR}}, C_{\mathrm{PC}}$, and C_{RP} are zero. Thus, we define different loss functions for learning the AS network, and learn AS independently of the main and auxiliary tasks.

The AS loss function is formulated using the loss function of the state value $V_{\mathrm{AS}}(s, \theta_{\mathrm{AS}})$ and the policy $\pi_{\mathrm{AS}}(a|s, \theta_{\mathrm{AS}})$, as

$$L_{\mathrm{AS}v} = (r + \gamma V_{\mathrm{AS}}(s_{t+1}, \theta^{-}_{\mathrm{AS}}) - V_{\mathrm{AS}}(s_t, \theta_{\mathrm{AS}}))^2. \tag{3}$$

$$\begin{aligned} L_{\mathrm{AS}p} = &- \log(\pi_{\mathrm{AS}}(a|s, \theta_{\mathrm{AS}}))A(s,a) \\ &- \alpha H(\pi_{\mathrm{AS}}(\cdot|s, \theta_{\mathrm{AS}})). \end{aligned} \tag{4}$$

where θ^{-}_{AS} are the network parameters before the AS network update, and r is the reward of the main task (score of the video game). Also, entropy $H(\pi_{\mathrm{AS}}(\cdot|s, \theta_{\mathrm{AS}}))$ is a term to facilitate the search such that the network parameters do not converge to a local solution, and α is the scale parameter of entropy $H(\pi_{\mathrm{AS}}(\cdot|s, \theta_{\mathrm{AS}}))$.

The AS loss function is defined as the sum of the losses in Eqs. (3), (4), as

$$L_{\mathrm{AS}} = L_{\mathrm{AS}v} + L_{\mathrm{AS}p}. \tag{5}$$

Eqs. (3), (4), (5) show that AS is a DRL agent that selects auxiliary tasks. In other words, AS is constructed as a model with different weights than the main task model for solving video games, and is optimized by the same reward as the main task to achieve the selection of auxiliary tasks to improve scores in video games.

3.4 Algorithm

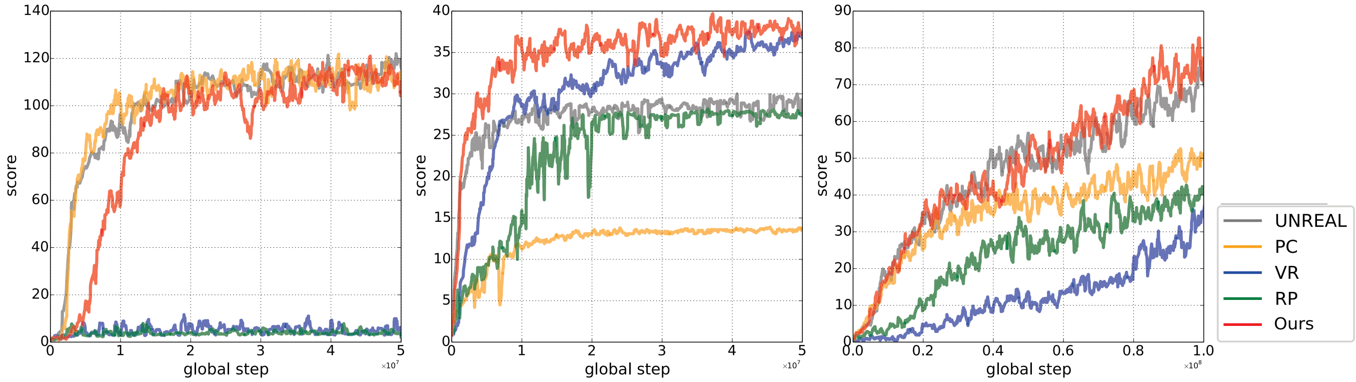
The processing flow of our method is shown in **Algorithm 1**. Here, θ, θ' denotes the main task model parameters and $\theta_{\mathrm{AS}}, \theta'_{\mathrm{AS}}$ denotes the auxiliary selection model parameters. Our method utilizes A3C as the training algorithm, which involves distributed learning by multiple workers and asynchronous updating of model parameters. Therefore, each worker has its own local model parameter $\theta', \theta'_{\mathrm{AS}}$ and local step t. In addition, the global model parameters $\theta, \theta_{\mathrm{AS}}$ and global step T are shared among workers.

First, we synchronize the network parameters of each worker θ' and θ'_{AS} with the shared parameters θ and θ_{AS}, respectively. Then, agents of each worker repeatedly take actions in an environment by following policy $\pi(a_t|s_t, \theta')$ until reaching a termination condition or t_{max} steps. The experiences (s_{t+1}, r_t, a_t) are stored in a replay buffer. Next, we execute the auxiliary selection and the three auxiliary tasks in turn, and we compute the gradients $d\theta$ and $d\theta_{\mathrm{AS}}$ shown in Eqs. (2), (5). Using these gradients, we update the global model parameters θ and θ_{AS}. This update is performed asynchronously on each worker and is repeated until T_{max} steps to adaptively select and learn auxiliary tasks.

4. Experiments

We used the DeepMind Lab[21)]for the evaluation of our method. DeepMind Lab mainly contains three games: i) nav_maze_static_01 (maze), ii) seekavoid_arena_01 (seekavoid), and iii) lt_horseshoe_color (horseshoe).

The maze is a first-person maze exploration game. The agent receives +1 for each apple obtained along the way and +10 for reaching the goal, and competes for the highest score within a given time. There are six actions that can be selected by the agent: move left, move right, move forward, move backward, move parallel to the left, and move parallel to the right. The seekavoid is a game in



(a) nav_naze_static_01 (b) seekavoid_arena_01 (c) lt_horseshoe_color

Fig. 2 Game score transition during training in DeepMind Lab

which agents earn +1 for each apple and −1 for each lemon they acquire, and compete to earn the highest score within a time limit. There are six actions that can be selected by the agent: move left, move right, move forward, move backward, move parallel to the left, and move parallel to the right. The horseshoe is a first-person shooter game. The agent attacks enemies appearing on the stage with a laser and earns +1 by defeating them, and competes for a score within a time limit. The agent can choose from seven actions: move left, move right, move forward, move backward, move parallel to the left, move parallel to the right, and attack.

We compared our method with the following baselines:
UNREAL: Three auxiliary tasks are used for training.
PC, VR, and RP: Each uses its own auxiliary task for training.
The common hyperparameters during training were unified. The training steps were 5.0×10^7 steps for maze and seekavoid, and 1.0×10^8 steps for horseshoe.

4.1 Comparison according to game score

Figure 2 shows the score transition during the training of the comparison methods in DeepMind Lab. Here, the horizontal axis indicates the number of global steps to update the network parameters, and the vertical axis indicates the score for each task. From left to right: maze, seekavoid, and horseshoe.

maze. Figure 2(a) shows the scores for maze. UNREAL and PC achieved higher performances, while the scores of VR and RP were almost zero. This means that VR and RP did not improve the main task. The PC encouraged the agent's action of exploring the maze by changing the pixel values. This enabled an agent to move in every corner of the maze environment. Our method also achieved the same score as UNREAL and PC.

seekavoid. Figure 2(b) shows the scores for seekavoid. PC was inadequate for seekavoid because the pixel values changed significantly even when negative rewards were obtained. RP was also not efficient because this environment offers dense rewards. In contrast, UNREAL and VR achieved higher scores. Surprisingly, VR outperformed UNREAL. In contrast to this result, our method also achieved the same performance as VR and achieved higher performance with fewer training steps than VR.

Table 2 The number of times each auxiliary task was selected in an episode

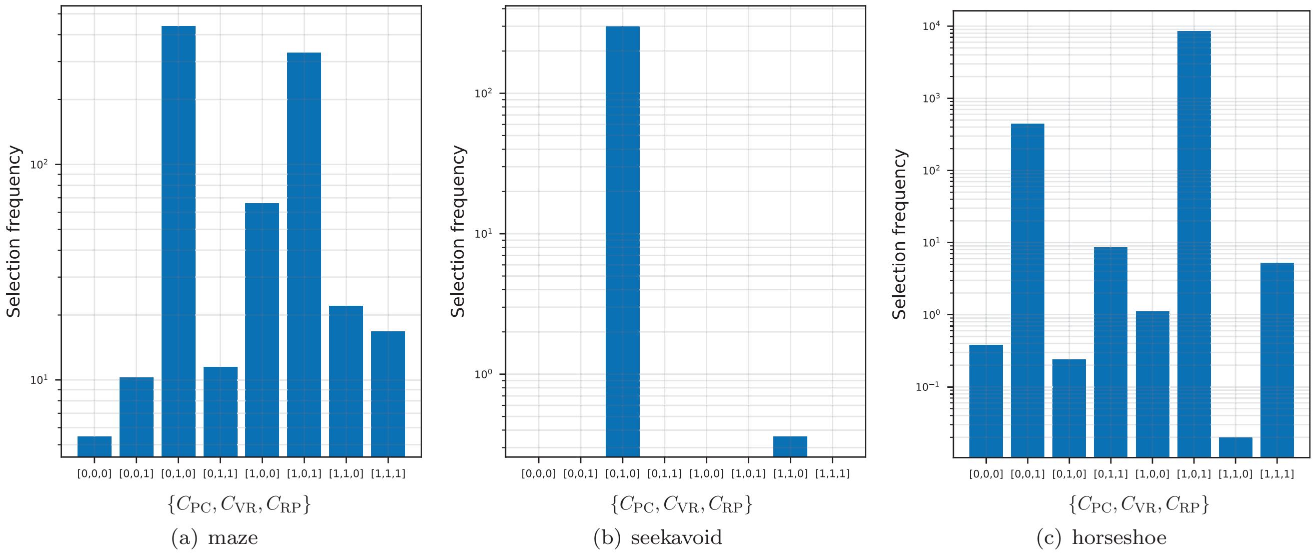
Env.	Auxiliary task		
	PC	VR	RP
maze	435.4	487.8	369.0
seekvoid	0.3	300.0	0.0
horseshoe	8545.1	14.1	8998.2

horseshoe. Figure 2(c) shows the scores for horseshoe. Here, PC had the highest score, since the actions that defeat enemies change pixel values significantly. However, UNREAL outperformed the other methods, and our method achieved the same performance as UNREAL.

4.2 Analysis of the selected auxiliary tasks

The number of actions selected by the auxiliary selection during one episode of each game is shown in **Figure 3**. Here, the horizontal axis represents the actions $\{C_{\mathrm{PC}}, C_{\mathrm{VR}}, C_{\mathrm{RP}}\}$ output from the auxiliary selection, and the vertical axis represents the number of times each action was selected. From left to right: (a) maze with 5.0×10^7 steps, (b) seekavoid with 5.0×10^7 steps, and (c) horseshoe with 1.0×10^8 steps. **Table 2** lists the number of times each auxiliary task was selected in an episode. We calculated the number of times the auxiliary task was selected as the average of 50 episodes. The number of action steps in an episode was 900 for maze, 300 for seekavoid, and 9,000 for horseshoe.

The results for maze show that all auxiliary tasks were equivalently selected. Because the appropriate auxiliary



(a) maze (b) seekavoid (c) horseshoe

Fig. 3 The number of selections per action by auxiliary selection during one episode

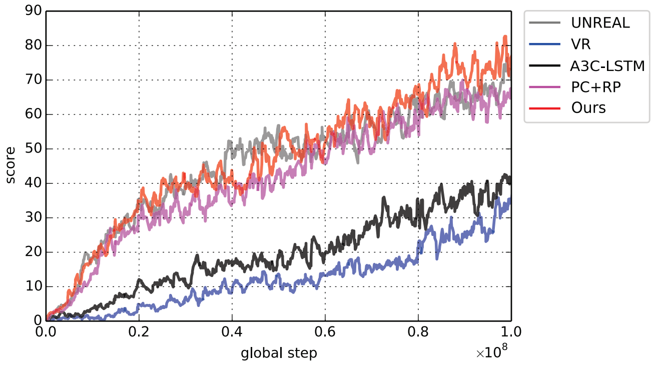


Fig. 4 Game scores in horseshoe with additional auxiliary task combinations

tasks for the maze task were UNREAL or PC, our method equally selected all auxiliary tasks. In seekavoid, our method stably selected the value function replay. Since these results correspond to the results in Figure 2(b), our method only selects auxiliary tasks that contribute to the training of the main task. In horseshoe, pixel control and reward prediction were often selected. Although the best score was achieved by UNREAL, auxiliary selection for horseshoe did not select value function replay. To analyze the reason of this selection, we conducted additional experiments. In addition to the results of the baselines shown in Figure 2(c), we added the following baselines: A3C-LSTM (without auxiliary tasks), and PC+RP (uses pixel control and reward prediction). **Figure 4** shows the scores of each baseline and our method. We can see here that the score of VR was lower than that of A3C, and that PC+RP achieved the same score as UNREAL and our method. Therefore, our method successfully removes the value function replay from the training of horseshoe.

The above findings demonstrate that our approach can select auxiliary tasks that contribute to training the main task.

4.3 Dynamic selection of auxiliary tasks according to training stages

In this Section, we analyze the percentage of auxiliary tasks selected for each training stage in order to confirm whether the optimal selection of auxiliary tasks based on the training stage is achieved. The percentage of auxiliary tasks selected during one episode for each training phase in seekavoid is shown in **Figure 5**. Here, we studied 0.2×10^7, 0.5×10^7, 1.0×10^7, and 5.0×10^7 training steps. The selection rate of the auxiliary task is the average of 50 episodes of the number of selections per episode using the model with each step trained.

From Figure 5, we can see that all auxiliary tasks were selected to the same degree in 0.2×10^7 steps and 0.5×10^7 steps. Here, UNREAL scored about 25 and 30 points higher for the 0.2×10^7 and 0.5×10^7 steps. From these results, we can assume that, in the early learning phase up to 0.5×10^7 steps, our method selects each auxiliary task just as well as UNREAL, which uses all auxiliary tasks to the same extent. On the other hand, we can confirm that our method preferentially selects VR in the 1.0×10^7 step. UNREAL and VR had the highest score of about 30 points in the 1.0×10^7 step, so we can assume that our method is able to select VR during the training stage where VR is effective. Here, only VR was selected for the 5.0×10^7 step. In the 1.0×10^7 step of the score graph, there is a large difference of about 7 points between the scores of UNREAL and VR, with VR obtaining the highest score. Therefore, we consider that the number of

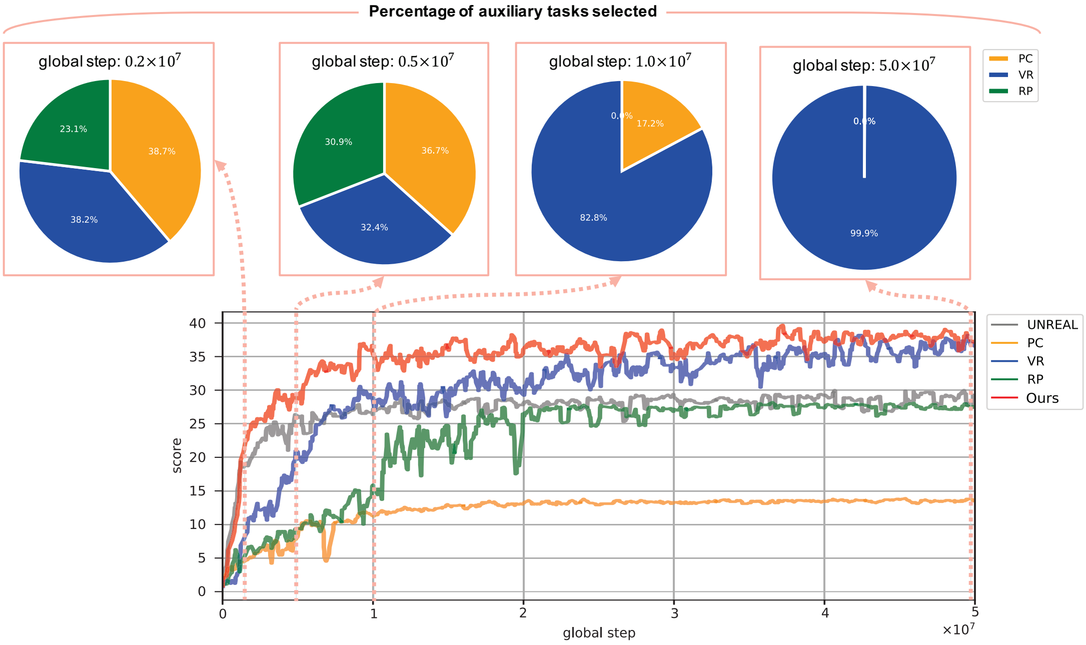


Fig. 5 Percentage of auxiliary task selection during one episode according to training stages (seekavoid)

times VR is selected by the auxiliary selection increased between 1.0×10^7 steps and 5.0×10^7 steps. These results indicate that auxiliary selection is able to select the combination of auxiliary tasks with the highest score in the main task in accordance with the training stage of the main task.

5. Conclusion

In this paper, we proposed auxiliary selection, which selects effective auxiliary tasks to be used during learning of the main task. Auxiliary selection is designed as a deep reinforcement learning agent that controls the binary weights of the auxiliary task in order to improve the accuracy of the main task. We applied our method to UNREAL, which introduces unsupervised auxiliary tasks in video game strategy, to allow dynamic selection of auxiliary tasks. This eliminates the need to manually design auxiliary tasks for each main task and enables more efficient learning of the main task. From experiments using DeepMind Lab, we confirmed that our method achieves a score that is equivalent to the optimal combination of auxiliary tasks for each environment. The analysis of the auxiliary tasks selected by auxiliary selection showed that our method can improve the performance of the main task by selecting the appropriate auxiliary task for the video game to be solved. In addition, by analyzing the auxiliary tasks selected at each training phase, we found that our method selects the most appropriate auxiliary task according to the training phase of the main task. As future work, we plan to apply our method to other auxiliary learning methods and to experiment with various tasks.

Acknowledgements

This work was supported by JST SPRING, Grant Number JPMJSP2158.

References

1) I. Kokkinos: "Ubernet: Training a Universal Convolutional Neural Network for Low-, Mid-, and High-level Vision using Diverse Datasets and Limited Memory", Proc. of the IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR), pp.6129-6138 (2017).

2) M. Teichmann, M. Weber, M. Zoellner, R. Cipolla, and R. Urtasun: "MultiNet: Real-time Joint Semantic Reasoning for Autonomous Driving", Proc. of the IEEE Intelligent Vehicles Symposium (IV), pp.1013-1020 (2018).

3) A. Kendall, Y. Gal, and R. Cipolla: "Multi-task Learning using Uncertainty to Weigh Losses for Scene Geometry and Semantics", Proc. of the IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR), pp.7482-7491 (2018).

4) Y. Qian, J.M. Dolan, and M. Yang: "DLT-Net: Joint Detection of Drivable Areas, Lane Lines, and Traffic Objects", IEEE Trans. on Intelligent Transportation Systems, vol.21, no.11, pp.4670-4679 (2020).

5) R. Collobert and J. Weston: "A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning", Proc. of the International Conference on Machine Learning (ICML), pp.160-167 (2008).

6) X. Liu, P. He, W. Chen, and J. Gao: "Multi-Task Deep Neural Networks for Natural Language Understanding", Proc. of the Annual Meeting of the Association for Computational Linguistics (ACL) (2019).

7) Lukas Liebel and Marco Körner: "Auxiliary Tasks in Multitask Learning", *arXiv preprint, arXiv:1805.06334* (2018).

8) M. Jaderberg, V. Mnih, W.M. Czarnecki, T. Schaul, J.Z. Leibo,

D. Silver, and K. Kavukcuoglu: “Reinforcement Learning with Unsupervised Auxiliary Tasks”, Proc. of the International Conference on Learning Representations (ICLR) (2017).

9) Z. Zhang, P. Luo, C.C. Loy, and X. Tang: “Facial Landmark Detection by Deep Multi-task Learning”, Proc. of the European Conference on Computer Vision (ECCV), pp.94—108 (2014).

10) P.W. Mirowski, R. Pascanu, F. Viola, H. Soyer, A. Ballard, A. Banino, M. Denil, R. Goroshin, L. Sifre, K. Kavukcuoglu, D. Kumaran, and R. Hadsell: “Learning to Navigate in Complex Environments”, Proc. of the International Conference on Learning Representations (ICLR) (2017).

11) B. Kartal, P. Hernandez-Leal, and M.E. Taylor: “Terminal Prediction as an Auxiliary Task for Deep Reinforcement Learning”, Proc. of the AAAI conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE) (2019).

12) P. Hernandez-Leal, B. Kartal, and M.E. Taylor: “Agent Modeling as Auxiliary Task for Deep Reinforcement Learning”, Proc. of the AAAI conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE) (2019).

13) N. Justesen, P. Bontrager, J. Togelius, and S. Risi: “Deep Learning for Video Game Playing”, IEEE Trans. on Games, vol.12, pp.1-20 (2017).

14) V. Mnih, A.P. Badia, M. Mirza, A. Graves, T. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu: “Asynchronous Methods for Deep Reinforcement Learning”, Proc. of International Conference on Machine Learning (ICML), pp.1928-1937 (2016).

15) V. Mnih, K. Kavukcuoglu, D. Silver, A.A. Rusu, J. Veness, M.G. Bellemare, A. Graves, M.A. Riedmiller, A.K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis,: “Human-level Control through Deep Reinforcement Learning”, Nature, vol.518, no.7540, pp.529-533 (2015).

16) Y. Teh, V. Bapst, W.M. Czarnecki, J. Quan, J. Kirkpatrick, R. Hadsell, N. Heess, and R. Pascanu: “Distral: Robust Multitask Reinforcement Learning”, Proc. of the Neural Information Processing Systems (NeurIPS), pp.4496-4506 (2017).

17) M. Riedmiller, R. Hafner, T. Lampe, M. Neunert, J. Degrave, T. Wiele, V. Mnih, N. Heess, and J.T. Springenberg: “Learning by Playing – Solving Sparse Reward Tasks from Scratch”, Proc. of the International Conference on Machine Learning (ICML) (2018).

18) Y. Du, W.M. Czarnecki, S.M. Jayakumar, R. Pascanu, and B. Lakshminarayanan: “Adapting Auxiliary Losses using Gradient Similarity”, *arXiv preprint, arXiv:1812.02224* (2018).

19) X. Lin, H.S. Baweja, G.A. Kantor, and D. Held: “Adaptive Auxiliary Task Weighting for Reinforcement Learning”, Proc. of the Neural Information Processing Systems (NeurIPS), pp.4772-4783 (2019).

20) G. Hinton, O. Vinyals, and J. Dean: “Distilling the Knowledge in a Neural Network”, Proc. of the Neural Information Processing Systems (NeurIPS) deep learning workshop (2015).

21) C. Beattie, J.Z. Leibo, D. Teplyashin, T. Ward, M. Wainwright, H. Küttler, A. Lefrancq, S. Green, V. Valdés, A. Sadik, J. Schrittwieser, K. Anderson, S. York, M. Cant, A. Cain, A. Bolton, S. Gaffney, H. King, D. Hassabis, S. Legg, and S. Petersen: “DeepMind Lab”, *arXiv preprint, arXiv:1612.03801* (2016).

(Received November 9, 2023)
(Revised April 26, 2024)

Hidenori ITAYA (*Student Member*)
He received his B.E. in Computer Science from Chubu University, Japan in 2019. He received his M.S. degree from the same institution in 2021. He has worked at DENSO WAVE incorporated since 2021, and is currently pursuing his Ph.D. from Chubu University. He is a member of the IIEEJ.

Tsubasa HIRAKAWA
He received his Ph.D. in Computer Science from Hiroshima University, Japan in 2017. From 2017 to 2019, he was a researcher fellow at the Chubu University, and he has been a specially appointed associate professor at the Chubu Institute for Advanced Studies, Chubu University, Japan since 2019. He has also been a lecturer in the Department of Computer Science, Chubu University, Japan since 2021. He was a Fellowship of the Japan Society for the Promotion of Science from 2014 to 2017, and he was a visiting researcher at ESIEE Paris, France, in 2014 and 2015.

Takayoshi YAMASHITA
He received his Ph.D, in Computer Science from Chubu University, Japan in 2011. He worked at OMRON Corporation from 2002 to 2014 and was a lecturer in the Department of Computer Science, Chubu University, Japan from 2014 to 2017, where he was also an associate professor from 2017 to 2021. He has been a professor at the same institute since 2021. His current research interests include object detection, object tracking, human activity understanding, pattern recognition, and machine learning. He is a member of the IEEE, the IEICE, and the IPSJ.

Hironobu FUJIYOSHI
He received his Ph.D. in Electrical Engineering from Chubu University, Japan, in 1997. From 1997 to 2000, he was a post-doctoral fellow in the Robotics Institute of Carnegie Mellon University, Pittsburgh, PA, USA, working on the DARPA Video Surveillance and Monitoring (VSAM) effort and the humanoid vision project for the HONDA Humanoid Robot. He is now a professor of the Department of Robotics, Chubu University, Japan. From 2005 to 2006, he was a visiting researcher at the Robotics Institute, Carnegie Mellon University. His research interests include computer vision, video understanding and pattern recognition. He is a member of the IEEE, the IEICE, the IPSJ, and the IEEE.

IIEEJ Paper -- Special Issue on Journal Track Papers in IEVC2024 --

Real-Time Intuitive Interaction and Realistic Illumination for CT Volume Rendering

Kousuke KATAYAMA†, Toru HIGAKI†, Kazufumi KANEDA† (*Member*), Bisser RAYTCHEV†,
Wataru FUKUMOTO††, Hidenori MITANI††

† Graduate School of Advanced Science and Engineering, Hiroshima University ,
†† Graduate School of Biomedical and Health Sciences, Hiroshima University

<**Summary**> We developed a real-time, intuitive interaction and photo-realistic illumination method for CT volume rendering. Our approach involves auto-stereoscopic display and hand-sensor-based gesture control, as well as lightweight and effective illumination, and a fast-sampling algorithm that enables them to be rendered in real-time. Consequently, our rendering method achieved render volume data obtained from general CT examinations at real-time 4K stereo view, allowing intuitive comprehension of the 3D structure, and providing the realism required not only for diagnostics but also for educational materials and forensic evidence.

Keywords: volume rendering, 3D user interaction, photo-realistic rendering

1. Introduction

While CT volume renderings are useful for diagnostic purposes, we focus on the use of CT volume renderings for the visualization of forensic evidence and as a medical educational tool. CT volume rendering for forensic evidence is used to visualize CT scans taken for screening and preservation of evidence before autopsy. CT volume rendering for medical educational purposes is also valuable because it is easier than working with specimens or real objects and can be viewed from a greater degree in any angle than photographs. For these CT volume rendering applications, it is important to be able to represent a realistic appearance and to be intuitively understandable by a non-specialist viewer.

Various methods have been proposed for CT volume rendering in the past[1)]there are many methods for data feature enhancement of CT volumes, and the method can be selected according to the inside and outside of the data space and the purpose. Representing a realistic appearance has also been achieved to some extent[2)]. However, while real-time interaction is important for intuitive comprehensibility, it has not been possible to achieve both realistic appearance representation and real-time interaction. This is because the computational cost of volume rendering is still high. In some of the most realistic-looking, real-time examples to date, Monte Carlo techniques[3)]have been incorporated into volume rendering to ensure real-time rendering and realistic lighting. However, Monte Carlo method rendering has a lag until the image is cleaned up and is not suitable for combination with interactions that are in constant motion.

We propose to combine realistic appearance and intuitive comprehensibility with realistic lighting and new real-time interactions, and to optimize a direct volume rendering method to achieve this combination in a non-Monte Carlo method. The contribution of our proposed methodology can be summarised in three items: (1) we incorporated a real-time realistic rendering technique used in surface rendering to the volume rendering method, (2) we improved the ray-marching acceleration with modified mipmap creation, (3) we proposed the use of a spatial reality display and hand-tracking sensors for superior interaction.

2. Related Works

Omori et al. developed a method for displaying medical CT volume data in 3D using spatial reality display[4)]. Although real-time 3D and 4D displays are achieved with this method, only simplified lighting is applied to volume rendering, resulting in low photorealism.

Photorealistic rendering aims at a realistic representation of the object and the physical light transport results are rendered to be the same as in the real world. Shading techniques such as BRDF are used to make the reflective properties of light photorealistic, as well as global lighting and image-based lighting techniques. These techniques are generally used for surface rendering. Cinematic ren-

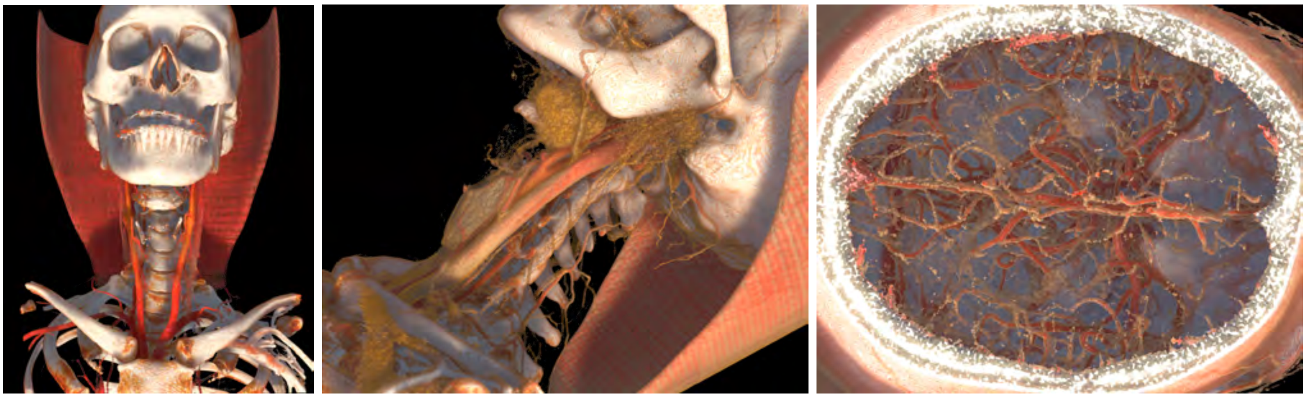


Fig. 1 Real-time realistic illumination for contrast-enhanced neck CT volume rendering using our real-time rendering method

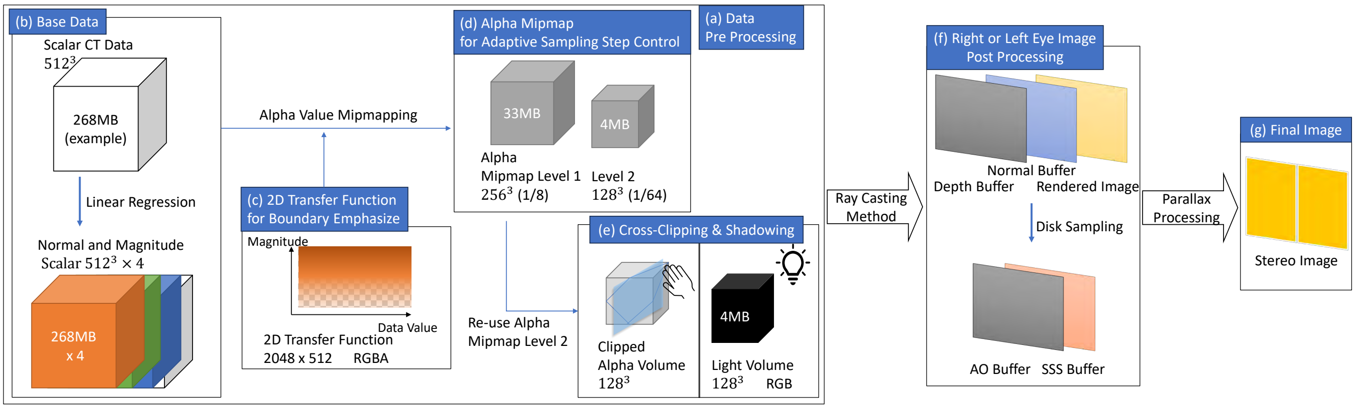


Fig. 2 Rendering pre-processing and post-processing system pipeline overview

dering[5]and Exposure Render[2]have been proposed as realistic volume rendering methods. These methods apply Monte Carlo rendering techniques[3]to volume rendering, which is computationally expensive and unsuitable for real-time rendering.

In order to improve the understandability of volume data, non-photorealistic rendering methods may be useful as well as realism. In volume illustration[6], the normal direction of the volume and boundary surface intensity can be used to enhance contours and boundaries and improve the visibility of the geometry.

Studies on speeding up volume rendering have utilized the hierarchical data structure of volumes[7]and, more recently, image reconstruction using neural networks to reduce computational costs[8].

3. Our Approach

Our volume rendering combines realistic illumination and real-time interaction. Our approach includes the following:

(a) Physically based shading with visibility control that changes transparency according to sight direction, normal, and normal magnitude.

(b) Real-time colored shadowing technique

(c) Screen space-based illumination technique that avoids expensive calculations

(d) Adaptive sampling step control for direct volume rendering based on mipmap volume

(e) Intuitive interaction provided by a free-viewpoint autostereoscopic display and optical tracking sensors

Figure 1 shows a rendering with realistic lighting based on our approach. **Figure 2** shows the rendering system using a 512^3 voxels volume as an example (Fig. 2-(b)). It shows the data and its flow used for pre-processing (Fig. 2-(a)) and direct volume rendering, and the image buffer used for post-processing (Fig. 2-(f)).

3.1 Shading

In making volume rendering realistic, there is a weakness in the representation of volume rendering, which is that the outlines and boundaries of objects represented by volume rendering are unclear. This depends on the data, but if the transparency gradient of the transfer function according to the CT value is steep enough to make it clear, artifacts will be created.

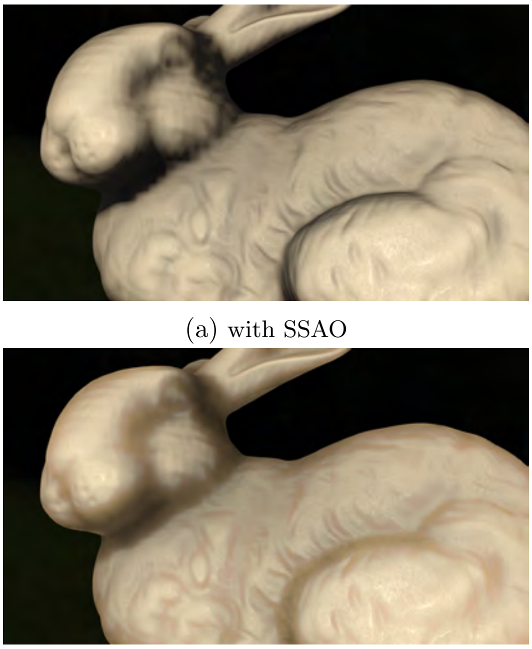
Therefore, to improve visibility, we have incorporated a volume illustration method[6]of controlling transparency by line of sight, normal, and its magnitude, without changing the transparency gradient according to CT value. By increasing the original transparency according to the magnitude, all areas except the boundary are made transparent and the boundary is emphasized. A 2D transfer function is used to consider magnitude, in which the vertical axis of magnitude is added to the horizontal axis of CT value in Fig. 2-(c). Then, an outline clarification process is performed during shading by opacifying the outline by the angle between the line of sight and the normal. The shading alpha and RGB values applied to each sample in direct volume rendering are summarized in Eq. (1) and (2). Where f is a 2-dimensional transfer function of CT-value and its gradient magnitude, s is CT-value, $\vec{n}$ is CT-gradient calculated by linear regression[9] (Fig. 2-(b)), and $\vec{d}$ is ray direction, p_m and p_s are parameters for controlling metallicity and smoothness, respectively. PBS (Physically Based Shader) is a reused Unity Standard Shader[10]function. Equation (1) contains an alpha value increase to enhance the silhouette lines of the object[6]. p_1 and p_2 are the silhouette enhancement parameter (Example: $p_1 = 8.0$, $p_2 = 12$).

$$\alpha = f_\alpha(s, |\vec{n}|)\left(1 + p_1\left(1 - \vec{d}\cdot\frac{\vec{n}}{|\vec{n}|}\right)\right)^{p_2} \tag{1}$$

$$\mathrm{RGB} = \mathrm{PBS}\left(f_{rgb}(s, |\vec{n}|), \vec{d}, \frac{\vec{n}}{|\vec{n}|}, p_m, p_s\right) \tag{2}$$

3.2 Shadowing

As a shadowing method, a volumetric lighting model[11] is used that is capable of real-time processing. As in the light volume light shown in Fig. 2-(e), the light volume has 1/4 resolution on each side and progresses through the entire light one step per frame, so the performance impact is minimal (for a source volume of size 512^3, it can be processed in less than 1 ms per frame). The light volume also illuminates the cross-section by using a cross-clipped alpha volume, which is described later in the interaction section. We have included a process that considers the color of translucent voxels during shadow computation to increase the cognitive effectiveness of the shadow representation. To avoid unnatural results, the process applies voxel color based on brightness values. Equation (3) and (4) represent the calculation of the color shadow. Where L_{in} , L_{mid} and L_{out} are light of each length, before, middle result, and after passing through the substance, respectively. C_α and C_i are substance color, opacity, and each wavelength component of the color, respectively. V is the brightness function of light.



(a) with SSAO

(b) with SSAO and SSSSS

Fig. 3 Screen space post-processings

$$L_{\mathrm{mid}} = L_{\mathrm{in}}\ (1 - C_\alpha(1 - C_i)) \tag{3}$$

$$L_{\mathrm{out}} = L_{\mathrm{mid}}\frac{V(L_{\mathrm{mid}})(1 - C_\alpha)}{V(L_{\mathrm{mid}})} \tag{4}$$

3.3 Screen space based technique

Post-processing performed in screen space is useful because it can be done in constant time regardless of the resolution of the volume. We process ambient occlusion and subsurface scattering in screen space (Fig. 2-(f)), which could not be considered computationally expensive in volume space to compute during direct volume rendering. Screen Space Ambient Occlusion (SSAO) is computed using common method[12]with depth buffer and normal buffer. Screen Space Sub-Surface Scattering (SSSSS) technique is commonly used in deferred rendering. However, since our method is a forward rendering, we used a simplified version of the former SSSSS method[13],[14]. Our method does not use an incident light buffer, but treats the forward rendering image as if it were incident light. The Sub-Surface Scattering (SSS) light is computed with a diffuse profile[13],[14]and disk sampling, and the final image is maxed with the scattered light and the forward rendered image (SSSSS = max(SSS, image)). In this method, only the areas brightened by SSS are pseudo-represented. Since SSS originally shows differences well at

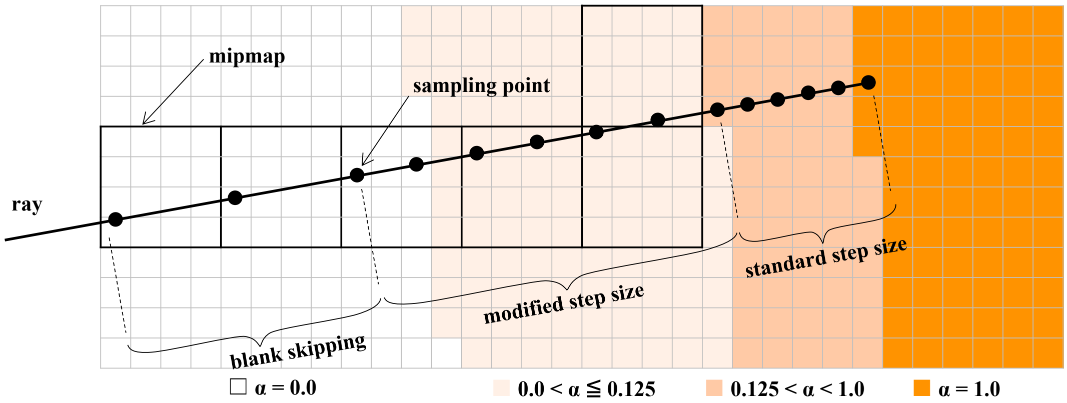


Fig. 4 Adaptive sampling step control

the edges of shadows, this method is also effective for representing translucent materials such as soft tissues when compared with and without SSS as shown in the figure. Finally, parallax processing is added to the stereo rendering as shown in Fig. 2-(g). **Figure 3** shows the effects of SSSSS and SSAO. Small shadows caused by fine surface irregularities are produced by SSAO, and the shadows are colored and softened by SSSSS.

3.4 Adaptive sampling step control

In direct volume rendering, most of the computational cost of a single ray lies in the ray-marching part. The ray-marching parts have two main computational costs: sampling the volume data and shading the samples. Since shading is associated with the valid samples obtained in sampling, we focused on reducing the number of samplings. We developed the adaptive sampling step control. Adaptive sampling step control is a method of determining the sampling step interval using area-opacity values expressed in mipmap alpha volume. By referring to the mipmap values, a wide-area sample can be obtained to determine the need for sampling and to omit sampling. Compared to similar methods that use mipmaps[7], this method is superior in that it can take larger steps due to modifications to the generation of the mipmap, and it can accelerate steps even in semi-transparent regions.

The data structure of the mipmap is the mipmap of the alpha volume when a 2D transfer function is assigned to the CT volume, as depicted in the data flow in Fig. 2-(b) (c) (d). We have made sure that when mipmapping, the lower level mipmap uses the maximum value rather than the average of the upper level 8 voxels. This is to avoid missing small objects. Also, after mipmapping, each voxel has applied a type of morphological processing so that it retains the maximum value of the surrounding 26 voxels. This is to ensure that no over-skipping occurs in the step control described below. These processes must be repeated when the transfer function changes, but since they take less than 10 ms for a base volume of size 512^3, they do not affect rendering unless the transfer function is constantly changing.

Figure 4 shows the correspondence between the sampling width of the adaptive sampling step control and the area-opacity value. There are three main levels of control depending on the area-opacity value. Hereafter, level 1 (1/2 of each side) of the alpha volume is represented as A_1, level 2 (1/4 of each side) as A_2, and the alpha of the original volume as A_0.

3.4.1 Blank skipping

First, when $\alpha = 0$ is sampled at A_2, The sampled area of the A_2 voxel and the surrounding area is completely transparent due to maximum mipmapping and morphological processing. Thus, significant blank skipping is possible. The step size can be set 8 times wider than the usual 1 voxel wide step. This reduces the sampling of transparent regions. This process does not affect image quality at all. We use A_2 instead of A_1 or A_3 because A_2 was found experimentally to be the most efficient. In addition, the use of additional A_1 or A_3 was almost ineffective because it would require additional memory access.

3.4.2 Modified step size

Now, noting the fact that regions with high alpha values ($\alpha \sim 1$) can be rendered faster with standard step size because the opacity accumulation finishes earlier, we find that a speedup is needed in translucent but near-transparent regions ($\alpha \sim 0$). To solve this problem, we change the step size according to the alpha value of the mipmap. Changing the step size requires a change in the ray-marching integration formula for consistency in alpha value integration. We modify the alpha value α

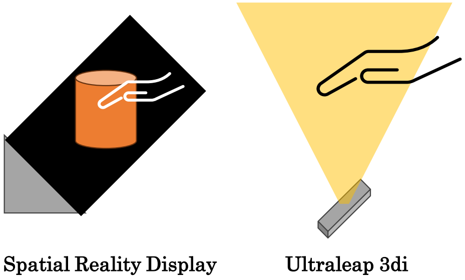


Fig. 5 Interaction devices overview

used in the integration formula according to the step size r as shown in Eq. (5). Where α is sampled substance alpha, and α^* is the substance alpha value used in the integration formula.

$$\alpha^* = 1 - (1 - \alpha)^r \tag{5}$$

In changing the step size, sampling can be skipped for regions with low alpha values because they have less impact on the image and their transfer function settings mean that they are not places to be focused on. However, we need to make sure that they are low alpha areas so that we do not skip sampling important high alpha areas. Our mipmap can guarantee a maximum alpha value in the neighborhood. We change the step size according to the alpha value of our mipmap. We have experimentally determined step size r so that speedup can be achieved with minimal impact on image quality. Equation (6) is the formula for determining the step size r. Where 2 is the maximum step size when the alpha is almost 0, and 0.125 is the maximum alpha value at which the step size change is made.

$$r = 1 + \frac{2}{0.125} \max(0.125 - A_2, 0) \tag{6}$$

3.4.3 Standard step size

Areas with high alpha values ($\alpha \sim 1$) do not require acceleration because the opacity accumulation is completed early and can be rendered faster with standard step size. The standard step size is the length of the shortest side of the voxel, to account for the case where the voxel is not a cube. If you want to further reduce noise or artists, you can further reduce the step size, but it is not recommended because of the high computational cost.

3.5 Interaction

Considering its use as an educational or forensic resource, it is important that interactions be easy to perform. Therefore, complex operations such as using a mouse and keyboard are inappropriate, as are VR and MR, which require the user to mount a device. Therefore, we focused on touchless interactions. For our interactions, we use the Spatial Reality Display[15](SONY Corporation) and the Ultraleap3di[16](Ultraleap Inc.) (shown in **Fig. 5**). This glasses-free stereoscopic display uses a built-in camera and face recognition to estimate the position of the eyes, enabling free-viewpoint autostereoscopic viewing as if it were a real 3D photograph. Since the position of the eyes is estimated, it is possible to actually look into the projected image by looking into the eyes. The Ultraleap3di is an optical hand sensor, which can recognize the detailed position of both fingers in the air above the sensor. It allows for non-contact operation, so if the sensor is placed in front of the display, the user can intuitively manipulate the 3D image by holding his or her hands over it.

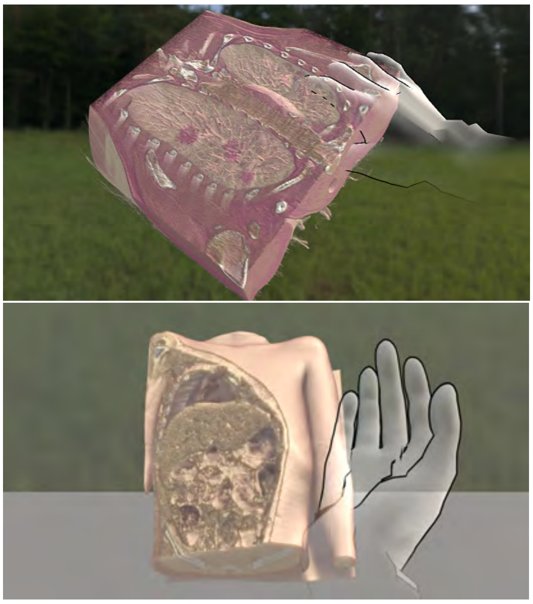
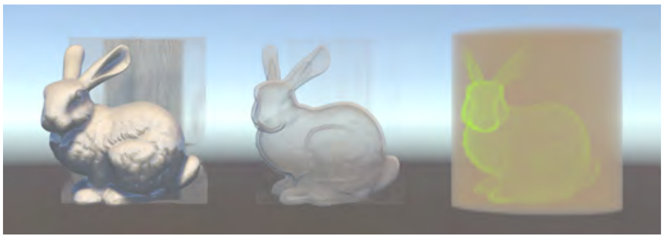


Fig. 6 Interaction of clipping cross-section plane

These two devices are incorporated with the rendering of our approach. Although there is no active interaction with the display, the free-viewpoint autostereoscopic viewing allows intuitive perception of the stereoscopic effect, and the viewpoint is fine-tuned by peering into the display. Real-time, 4K stereo rendering is required, so faster rendering is a must. Hand sensors can be used to determine the rough angles of the volume display. It can also be used to visualize a cross-section of the volume. The cross-section of the volume is the plane that is identical to the palm of the hand (shown in **Fig. 6**). This method of cross-sectional manipulation is more intuitive and easier than using a mouse. This is because the palm of the hand can be determined all at once, whereas manipulating a plane requires six-dimensional manipulation



(a) opaque object, (b) translucent object, (c) translucent air

Fig. 7 Various settings of transfer function

of position and rotation. To accurately reflect lighting on the cross-section as well, a cross-sectional volume is provided in the Fig. 2-(e) data flow that sets the alpha value to 0 for areas that become transparent due to cross-sectioning before the light volume.

4. Implements and Experiments

We implemented our rendering system on Unity2020. Direct volume rendering and post-processing are done by custom shaders processed on the GPU. Normal and magnitude calculations, mipmapping, and pre-processing of cross-sections and light volumes are also handled on the GPU by compute shaders. Free viewpoint autostereoscopic viewing and optical hand sensor control are done by an SDK provided by the manufacturer.

We measured the performance of the rendering system we created and the adaptive sampling step control in four different settings. The input was volume data of the Stanford Bunny[17]($512 \times 512 \times 361$) of ceramics taken by CT, the computer used was CPU Core-i9-11900 CPU, GPU RTX-3090, and we measured the performance with two 4K stereo renderings using Spatial Reality Display. Scene (a) assigns full transparency to air and opacity to ceramic, with the table at a slightly reflective transparency. Scene (b) uses magnitude to make only the material boundary appear translucent. Scene (c) assigns translucency to both air and ceramic, with air and ceramic changing color. Scene (no shading) has the same transfer function settings as Scene (a) and is an emission that displays the material colors as they are without shading. **Figure 7** and **Table 1** show the results of measurements with and without adaptive sampling step control for the four scenes and adaptive sampling step control. In all conditions, the adaptive sampling step control succeeds in increasing the speed, achieving real-time speeds of 30 fps or more.

We measured the computational load for each realistic illumination technique in order to examine the choice of each illumination technique when rendered using a lower performance computer. We used a computer: CPU Core-i7-4770K and GPU GTX-970, rendering a single 4K image. Volume and transfer function settings are identical to Scene (a) in Fig. 7. **Table 2** shows the measurements with 6 different illumination settings. You can see how long each effect took by comparing the resulting frame times with the previous and following settings. The lightest effect was shadowing (1.9 ms) and the heaviest was SSSSS (8.1 ms). Note that the processing costs for shading, shadowing depends on 2D transfer function, volume resolution, and rendering resolution, and post-processing mainly depends on rendering resolution.

Table 1 Performance of 4K stereo rendering

Settings	with Control	without Control
no shading	194 fps (5.2 ms)	88 fps (11.4 ms)
(a)	100 fps (10.0 ms)	62 fps (16.1 ms)
(b)	82 fps (12.2 ms)	44 fps (22.7 ms)
(c)	34 fps (29.4 ms)	14 fps (71.4 ms)

Table 2 Performance of realistic illumination technique with lower performance computer

Settings	Frame Rate
no shading	82 fps (12.2 ms)
Phong shading	67 fps (15.0 ms)
PBS shading	55 fps (17.9 ms)
PBS and Shadowing	50 fps (19.8 ms)
PBS, Shadowing and SSAO	35 fps (28.7 ms)
PBS, Shadowing, SSAO and SSSSS	26 fps (36.8 ms)

Table 3 Performance of adaptive sampling step control components with lower performance computer

Components	Scene (a)	Scene (c)
without Control	16 fps (61.8 ms)	15 fps (66.0 ms)
Blank Skipping	45 fps (22.3 ms)	12 fps (83.4 ms)
Modified Step Size	26 fps (38.4 ms)	28 fps (35.8 ms)
both Controls	49 fps (20.4 ms)	30 fps (33.2 ms)

We measured the performance of each of the adaptive sampling step controls blank skipping and modified step size. To measure the differences, we render single 4K image using a computer with the same lower-performance as in the experiment above. Volume and transfer function settings are the same as in Fig. 7 (a) and (c). **Table 3** shows the measured results. Blank skipping works well in settings with a lot of blanks, and modified step size works well in settings with a lot of high translucent space. In Scene (c), blank skipping rarely works, the result with blank skipping only is slower than without control due to the cost of accessing the mipmap.

The rendering performance for human body CT shown in **Fig. 8** and **Table 4**. A volume data of contrast-enhanced body CT ($512 \times 512 \times 1234$ voxels) was used

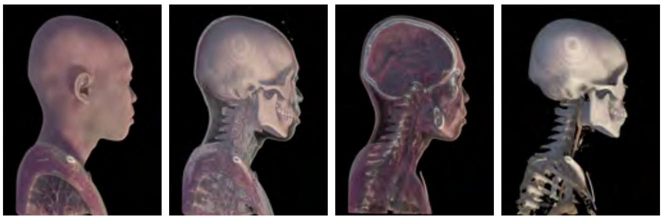

(a) Skin (b) Trans. (c) Clipping (d) Bones

Fig. 8 Rendering results of clinical CT

Table 4 Performance of 4K clinical CT rendering

Settings	Frame Rate
(a) Skin Surface	160 fps (6.3 ms)
(b) Translucent Skin	124 fps (8.1 ms)
(c) Cross Clipping	104 fps (9.6 ms)
(d) Bones and Vessels	200 fps (5.0 ms)

as the input, and single image rendered on the same computer as above. Figure 8 shows images rendered in 4K resolution with different transfer function settings and clipped to the target area (2160 pixels vertically). Real-time and realistic illuminated rendering was achieved with expected clinical CT data and display settings.

To assess the visual usefulness of the proposed method, we carried out a visual assessment experiment on the understandability of the three-dimensional structure of the vessels. The celiac artery in the early phase images of ten abdominal contrast-enhanced CTs was evaluated. Two radiologists (15 - and 14 years experience) specializing in interventional radiology scored the images on the following three-point scale. Score 3: Vascular structures could be identified at a glance without any rotation manipulation of the volume. Score 2: The vascular structures could be identified by slightly rotating the volume. Score 1: The vascular structure could be identified by rotating the volume many times. We compared the scores of the proposed and conventional volume rendering methods. Here, the functions (a), (b), (c), and (e) presented in Chapter 3 were disabled from the proposed method and used as the conventional volume rendering method.

The representative images used in the visual evaluation are shown in **Fig. 9**. The results of the visual evaluation are presented in **Table 5** and **Table 6**. Both readers scored higher with the proposed method (paired t-test, $p < 0.01$).

5. Discussion and Conclusion

We developed a CT volume rendering method that combines realistic illumination with interactions that require fast rendering by developing an accelerated method for direct volume rendering. With this method, even non-Monte Carlo direct volume rendering methods can efficiently handle sub-surface scattering, which is important for soft tissue representation. Our acceleration methods can also be used with a selection of various illumination techniques, allowing for real-time rendering even if you do not have access to a high-performance computer. In terms of interaction, the combination of a free viewpoint stereoscopic view and an optical hand sensor has realized a new intuitive operation in cross-sectional manipulation.

In this paper, we focused on volume data obtained by CT scanning. However, the method will be useful for intuitive understanding of volume data acquired by different methods, such as 3D observational data or simulated data. However, the volume data that can be handled by this method are only grid data, and special implementation or resampling will be required when handling volume data with non-grid structures.

In this study, hand gesture-based interactive manipulation methods were not evaluated and should be carried out in future work. Operation using hand gestures is not necessarily superior to operation using a conventional mouse, and the non-contact feature is more valuable than the operability. It will be necessary to design an evaluation experiment that considers this.

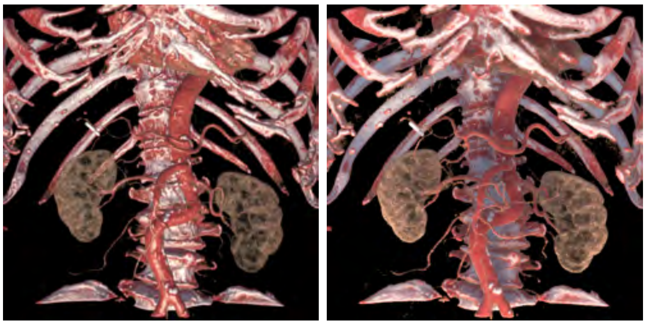

(a) Conventional rendering (b) Proposed rendering

Fig. 9 Representative images used for visual evaluation

Table 5 Visual evaluation of proposed method

Reader	Score 3 count	Score 2 count	Score 1 count	Average score
1	7	3	0	2.8
2	6	3	1	2.5

Table 6 Visual evaluation of conventional method

Reader	Score 3 count	Score 2 count	Score 1 count	Average score
1	0	8	2	1.8
2	0	4	6	1.4

References

1) C. Xu, G. Sun and R. Liang: “A survey of volume visualization techniques for feature enhancement”, Visual Informatics, **5**, 3, pp. 70–81 (2021).

2) T. Kroes, F. H. Post and C. P. Botha: “Exposure render: An interactive photo-realistic volume rendering framework”, PLOS ONE, **7**, 7, pp. 1–10 (2012).

3) D. Lin, C. Wyman and C. Yuksel: “Fast volume rendering with spatiotemporal reservoir resampling”, ACM Trans. Graph., **40**, 6 (2021).

4) A. Omori, R. Inuzuka and Y. Hirata: “Visualization of the complex double-outlet right ventricle anatomy using real-time three-dimensional computer graphics”, JTCVS Tech., **18**, pp. 97–99 (2023).

5) E. Dappa, K. Higashigaito, J. Fornaro, S. Leschka, S. Wildermuth and H. Alkadhi: “Cinematic rendering - an alternative to volume rendering for 3D computed tomography imaging”, Insights Imaging, **7**, 6, pp. 849–856 (2016).

6) P. Rheingans and D. Ebert: “Volume illustration: Nonphotorealistic rendering of volume models”, IEEE Trans. on Visualization and Computer Graphics, **7**, 3, pp. 253–264 (2001).

7) M. Levoy: “Efficient ray tracing of volume data”, ACM Trans. Graph., **9**, 3, pp. 245–261 (1990).

8) D. Bauer, Q. Wu and K. Ma: “Fovolnet: Fast volume rendering using foveated deep neural networks”, IEEE Trans. on Visualization amp; Computer Graphics, **29**, 01, pp. 515–525 (2023).

9) L. Neumann, B. Csébfalvi, A. König and E. Gröller: “Gradient estimation in volume data using 4d linear regression”, Computer Graphics Forum, **19**, 3, pp. 351–358 (2000).

10) Unity: “Standard shader”, https://docs.unity3d.com/2020.3/Documentation/Manual/shader-StandardShader.html (2020).

11) T. Ropinski, C. Doring and C. Rezk-Salama: “Interactive volumetric lighting simulating scattering and shadowing”, 2010 IEEE Pacific Visualization Symposium (PacificVis 2010), Los Alamitos, CA, USA, IEEE Computer Society, pp. 169–176 (2010).

12) M. McGuire, B. Osman, M. Bukowski and P. Hennessy: “The alchemy screen-space ambient obscurance algorithm”, Proc. of the ACM SIGGRAPH Symposium on High Performance Graphics, HPG ’11, New York, NY, USA, Association for Computing Machinery, pp. 25–32 (2011).

13) P. H. Christensen: “An approximate reflectance profile for efficient subsurface scattering”, ACM SIGGRAPH 2015 Talks, SIGGRAPH ’15, New York, NY, USA, Association for Computing Machinery (2015).

14) E. Golubev: “Efficient screen-space subsurface scattering using burley’s normalized diffusion in real-time”, https://advances.realtimerendering.com/s2018/ (2018).

15) SONY: “Spatial reality display white paper: Technical background”, https://www.sony.net/Products/Developer-Spatial-Reality-display/jp/develop/WhitePaper.html (2023).

16) Ultraleap: “Ultraleap 3di”, https://www.ultraleap.com/product/ultraleap-3di/ (2022).

17) M. Levoy: “The stanford 3d scanning repository”, https://graphics.stanford.edu/data/3Dscanrep/ (2000).

(Received November 10, 2023)
(Revised April 7, 2024)

Kousuke KATAYAMA

He received his B.E. and M.E. in Informatics and Data Science from Hiroshima University in 2022 and 2024. His research interests include medical volume rendering. Currently, he is an employee of CAPCOM Corporation, which is in the video game development industry.

Toru HIGAKI

He received his Ph.D. in Engineering from Hiroshima University, Japan, in 2011. Currently, he is an associate professor in Graduate School of Advanced Science and Engineering, Hiroshima University, Japan. His current research interests include medical imaging, visualization, and medical physics.

Kazufumi KANEDA (*Member*)

He received his D.E. in system engineering from Hiroshima University, Japan, in 1991. Currently, he is a professor at Graduate School of Advanced Science and Engineering, Hiroshima University, Japan. His research interests include computer graphics, data visualization, and medical graphics.

Bisser RAYTCHEV

He received his Ph.D. in Informatics from Tsukuba University, Japan, in 2000. Currently, he is an associate professor in Graduate School of Advanced Science and Engineering, Hiroshima University, Japan. His current research interests include machine learning and natural language processing.

Wataru FUKUMOTO

He received his Ph.D. in Medicine from Hiroshima University, Japan, in 2018. Currently, he is an assistant professor in Graduate School of Biomedical and Health Sciences, Hiroshima University, Japan. His specialties include diagnostic radiology, interventional radiology, and postmortem imaging.

Hidenori MITANI

He received his Ph.D. in Medicine from Hiroshima University, Japan, in 2022. Currently, he is an assistant professor in Graduate School of Biomedical and Health Sciences, Hiroshima University, Japan. His specialties include diagnostic radiology interventional radiology, and emergency radiology.

IIEEJ Paper -- Special Issue on Journal Track Papers in IEVC2024 --

An Examination of Motion Analysis During Weight Illusion by Impression Change of the Own Body

Joichiro MURAOKA†, Kosei TOMIOKA† (*Student Member*), Yusei MURAISHI†, Naoki HASHIMOTO††, Mie SATO†

† Utsunomiya University , †† The University of Electro Communications

<**Summary**> The development of virtual reality technology has made it possible to easily change the appearance of a person, and it has been suggested that the impression change of the appearance using virtual reality avatars affects not only self-perception but also weight perception. We have studied the physical effects of the impression change of the own body during the weight illusion based on electromyogram. As a result, a significant correlation was obtained between the electromyogram and the degree of the weight illusion among the subjects, although the correlation was not significant for each subject. In this study, to investigate the change in motion during the weight illusion, we analyze the motion under the impression change of the own body through a comparison experiment of dumbbells' weights. The results showed that subjects who obtained a correlation between the degree of impression of the strength of the avatar and the degree of weight illusion had a positive correlation between the degree of impression of the strength of the avatar and the velocity of their motion. The results suggest that the velocity of motion, the degree of impression of the strength of the avatar, and the degree of the weight illusion are related with each other.

Keywords: virtual reality avatar, self-perception, weight illusion, Proteus effect

1. Introduction

In recent years, research that gives users the illusion of impression by changing visual information has been attracting attention. Among these, the study of the Proteus effect, in which the appearance of an avatar in virtual reality (VR) influences the user's behavior and self-expression, is particularly well-known. Sumida et al.[1]conducted experiments on weight perception using avatars with different muscle masses. The results suggested that changes in cognition in the VR space, represented by the Proteus effect, even affect the weight perception. Okubo et al. conducted an experiment in which dumbbells were lifted (dumbbell lift) using a variety of avatars[2]. The results suggested a negative correlation between the weight perception and the strength impression in a variety of whole-body avatars. That is, the stronger the impression of avatar, the lighter the dumbbell tended to be felt. However, there is insufficient research on the Proteus effect and physical performance, and there is room to examine if there is a possibility for immersion in VR avatars to bring out users' potential power and physical capabilities. We used electromyogram (EMG) to investigate whether the user's physical force output changes during the weight illusion by impression change. As a result, a positive correlation between the degree of the weight illusion and the change in EMG was observed for all subjects, but any significant correlation within each subject was not observed[3].

Therefore, the purpose of this study is to clarify the effects on the body during the weight illusion by the impression change using full-body avatars. Using avatars with different impressions, we investigate if there are relations between the degree of strength impression given to the avatar (hereafter, avatar impression quantity), the degree of weight illusion (hereafter, weight illusion quantity), and the change in movement speed when comparing dumbbell weights.

2. Related research

Stereotypes and assumptions about visual information affect human cognition. This has led to the development of research on the relation between stereotypes and assumptions and cognition, and the relation between the appearance of avatars used by users and their self-perception has been actively investigated.

Yee showed that the appearance of a user's avatar may influence the user's speech, behavior, and sociability when communicating through an avatar in a virtual space[4]. They call the phenomenon of behavior change due to the appearance of avatar the Proteus effect. Banakou et al. showed that the use of Albert Einstein's avatar, which is considered highly intelligent, contributed to improved performance on cognitive tasks and reduced negative prejudice toward the elderly[5]. These reports confirm that stereotypes and assumptions about visual information affect cognition in VR environments as well as in real environments.

Recent studies have suggested that changes in cognition in VR spaces, as typified by the Proteus effect, even affect weight perception[1]. Based on these studies, we investigated the effect of the impression of strength on the body replaced by a 3D model on weight perception[2] . The results showed that the more forceful one feels toward the 3D model of one's own body, the lighter one tends to perceive the lifted object to be[2].

There have also been a few studies on the influence of the Proteus effect on the body, and Marcin et al. investigated the effect of the Proteus effect on the degree of fatigue[6]. The results showed that the use of muscular avatars tended to increase the number of biceps curls.

Thus, the relation between cognition and weight perception and between cognition and fatigue have been studied, but there are no studies on differences in force output and movement.

3. Experiment

The avatar impression quantity and the weight illusion quantity are first obtained through "Avatar evaluation task" followed by "Weight comparison task." During the weight comparison task, we measure the position coordinates of the subject's right forearm during the weight comparison of dumbbells with avatars. The velocity is obtained from the measured position coordinates. The correlations between the avatar impression quantity, weight illusion quantity, and velocity in motion are then examined.

3.1 Experiment environment

In this experiment, a head-mounted display (HMD) (HTC, VIVE Pro) and a motion capture system (OptiTrack, PrimeX13) were used to construct a system to evaluate the avatar impression quantity and weight illusion quantity. During the experiment, the subject's movements were captured by the motion capture system and the avatar's movements were synchronized with the subject's movements in a VR space. The subject could see the avatar's appearance synchronized with his/her own movements using a virtual mirror installed in the VR space. **Figure 1** shows the subject during the experiment and an example of the avatar's appearance in the VR space presented on the HMD.

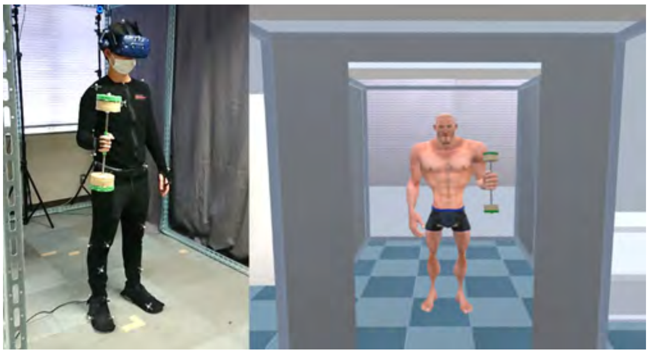


Fig. 1 Experiment scene (subject (left), subject's appearance as seen by the subject through the HMD (right))

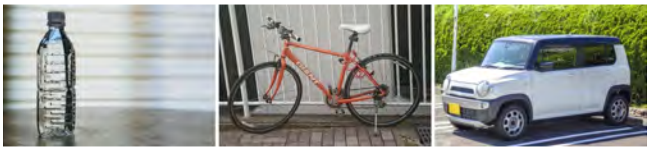


Fig. 2 Example of object images

3.2 Avatar evaluation task

The purpose of this task is to evaluate the basic impression quantity of each avatar.

In the Avatar evaluation task, the subject first selects "the heaviest object that the subject can hold by his/her own strength" from a set of 32 object images as shown in **Fig. 2**. Using this as the reference object, the weight ratio of each of the remaining 31 object images is evaluated in comparison with the reference object. In the experiment, in order to collect the intuitive quantities relative to the reference stimulus as accurately as possible, the subject is asked to answer the weight ratio using a slider that shows continuous values.

Next, this group of the object images are presented to the subject to determine his/her impression of the avatar. The subject sees himself/herself as an avatar in the VR space through the HMD and evaluates his/her impression of the avatar. For each of the 24 avatars, as shown in **Fig. 3**, the subject selects the "heaviest object this avatar can hold" from the set of 32 images. The basic impression quantity of each avatar is defined as Eq. (1).

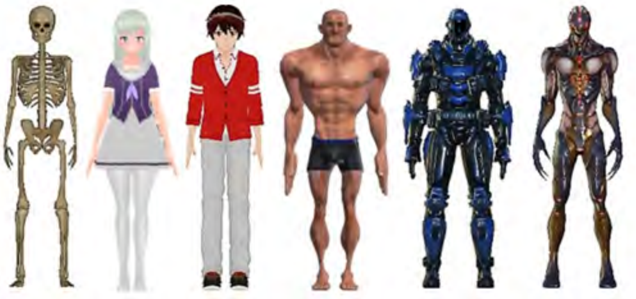


Fig. 3 Example of whole-body avatars used in the experiment

$$\begin{aligned}&Basic\ impression\ quantity\\&=\frac{Weight\ ratio\ of\ the\ heaviest\ object\ selected}{Weight\ ratio\ of\ the\ reference\ object\ (=1.0)}\end{aligned} \tag{1}$$

3.3 Weight comparison task

The purpose of this task is to obtain the avatar impression quantity and the weight illusion quantity.

In the weight comparison task, taking into an account the burden on the subject, the experimenter selects six avatars from the 24 avatars used in the avatar evaluation task. How to select the six avatars is to calculate the avatar impression quantity defined in Eq. (2) and to select six avatars whose avatar impression quantities are 100 or less in any combination of the selected two avatars.

$$\begin{aligned}&Avatar\ impression\ quantity\\&=\frac{Basic\ avatar\ impression\ quantity\ of\ avatar\ B}{Basic\ avatar\ impression\ quantity\ of\ avatar\ A}\end{aligned} \tag{2}$$

On each trial, two avatars are chosen from the six selected avatars. The subject becomes the avatars' appearances (appearance A and B) and compares the weights of the dumbbells. The following procedure is used for the weight comparisons.

(i) Posing while looking at the avatar's appearance of himself/herself in the mirror.

(ii) Dumbbell lift with avatar A (Dumbbell Lift 1).

(iii) Dumbbell lift with avatar B (Dumbbell Lift 2).

(iv) Evaluate the weight ratio of Dumbbell Lift 2 when the weight of Dumbbell Lift 1 is 1.0.

In the step (i), the subject is asked to look at the avatar in a mirror and performs some poses that he/she can imagine from that avatar in order to increase his/her immersion in the avatar. This step contributes to improvement of his/her sense of ownership to the avatar. In steps (ii) and (iii), two of the six avatars are presented at random, and the subject performs a 2 kg dumbbell lift while looking at the avatar synchronized with his/her movements in the mirror. Note that the subject does not know the weight of the dumbbell is 2 kg. In step (iv), a slider that shows continuous values between 0.0 and 2.0 is used to rate the perceived weight of Dumbbell lift 2 compared to the weight of Dumbbell lift 1. The evaluated weight ratio here is the weight illusion quantity of this trial. The weight illusion quantity is defined as Eq. (3).

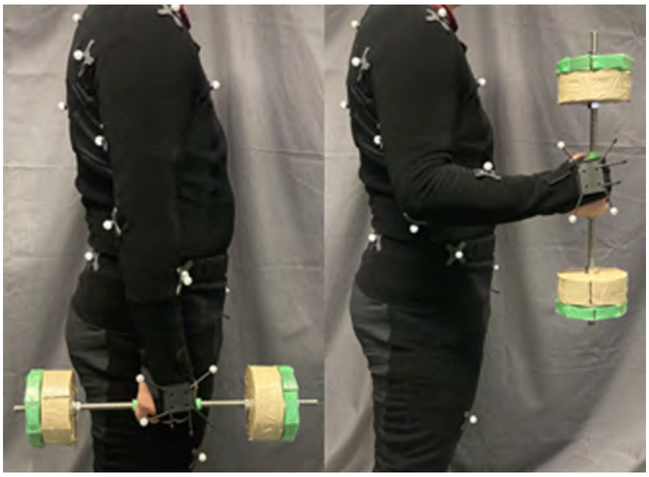


Fig. 4 States of the forearms (the start of the dumbbell lift (left), the end of dumbbell lift (right))

$$\begin{aligned}&Weight\ illusion\ quantity\\&=\frac{Weight\ of\ Dumbbell\ lift\ 2}{Weight\ of\ Dumbbell\ lift\ 1}\end{aligned} \tag{3}$$

While the subject performs the steps (ii) and (iii), the position coordinates of his/her right forearm is obtained. As shown in **Fig. 4**, markers are added to the right forearm of the motion capture suit. In addition, EMG sensors are attached to the brachial bicep and forearm to measure EMG.

The total number of trials is 36, including 30 avatar combinations and six dummy comparisons.

3.4 Post evaluation

At the end of the experiment, the subject is asked to answer questions about his/her immersion and sense of unity for each of the six avatars used in the weight comparison task. To answer, a seven-point Likert scale (−3 for "did not feel at all," 0 for "neutral" and +3 for "felt very strongly") is used. In this evaluation, a sense of agency and a sense of ownership are used as indices to measure the immersion and the sense of unity. The following questions are asked in the post-evaluation.

Q1. Did you feel that the avatar's movements matched your movements?

Q2. Did you feel that your avatar was reflected in the mirror?

Q3. Did you feel as if you were the avatar?

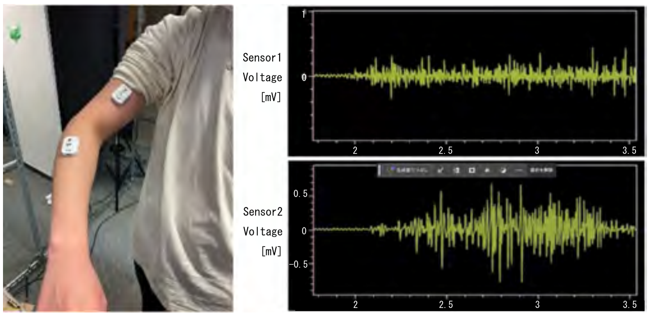


Fig. 5 EMG measurement (sensor position (left), example of resulting graph (right))

4. Results

Five male subjects aged 20-24 years, whose weight perception was judged to be normal by comparing the weight ratio of two dumbbells, participated in the experiment.

4.1 Analysis of the velocity and EMG

The analysis target section is from the start of the dumbbell lift to the end of the dumbbell lift. Figure 4 shows the states of the forearm at the start and end of the dumbbell lift.

The velocity is calculated from the coordinates of the marker of the right forearm obtained by the motion capture system. The velocity data point for each dumbbell lift is the average of the frame-by-frame velocities within the analysis section. The velocity data point for each trial is obtained using Eq. (4).

$$Velocity\ data\ point = \frac{Velocity\ data\ for\ Dumbbell\ lift\ 2}{Velocity\ data\ for\ Dumbbell\ lift\ 1} \tag{4}$$

There are 150 data points from the five subjects. Of these, 142 velocity data points were successfully obtained and analyzed.

EMG data were measured with an EMG sensor (TRUNK SOLUTION CORPOTATION, TS-MYO). The position of the EMG sensor is shown in **Fig. 5**. First, EMG data were full-wave rectification. The EMG data point for each dumbbell lift were averaged over the full-wave rectified data within the analysis section. The EMG data point for each trial is obtained using Eq. (5).

$$EMG\ data\ point = \frac{EMG\ data\ for\ Dumbbell\ lift\ 2}{EMG\ data\ for\ Dumbbell\ lift\ 1} \tag{5}$$

There are 150 data points obtained from the five subjects.

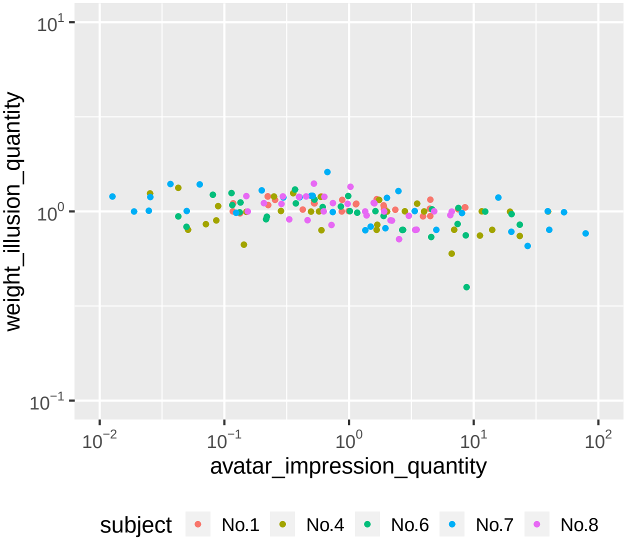


Fig. 6 Scatter plot of the avatar impression quantity and weight illusion quantity data points

4.2 Correlation between the avatar impression quantity and the weight illusion quantity

The purpose of this experiment is to clarify the effects of the illusion on the body. The five subjects showed a negative correlation between the avatar impression quantity and the weight illusion quantity. Spearman's rank correlation analysis was conducted on the avatar impression quantity and the weight illusion quantity. A significant negative correlation was obtained ($\rho = -.44$, $p < .001$). **Figure 6** shows a scatter plot of the avatar impression quantity and weight illusion quantity data points for the subjects with the negative correlation.

4.3 Correlation between the avatar impression quantity and the velocity data point

Spearman's rank correlation analysis was performed on the avatar impression quantity and the velocity data point, taking into an account the influence of outliers. A significant positive correlation was obtained ($\rho = .36$, $p < .001$). There is a relation between the strength impression of the avatar and the lifting speed of the dumbbell. **Figure 7** shows a scatter plot of the avatar impression quantity and velocity data points for the five subjects.

The positive correlation between the avatar impression quantity and velocity data points was confirmed. Therefore, the velocity data was decomposed into X, Y and Z components, and relations between each component and the avatar impression quantity were analyzed. As a result, no significant correlation was found between the X and Z components and the avatar impression quantity

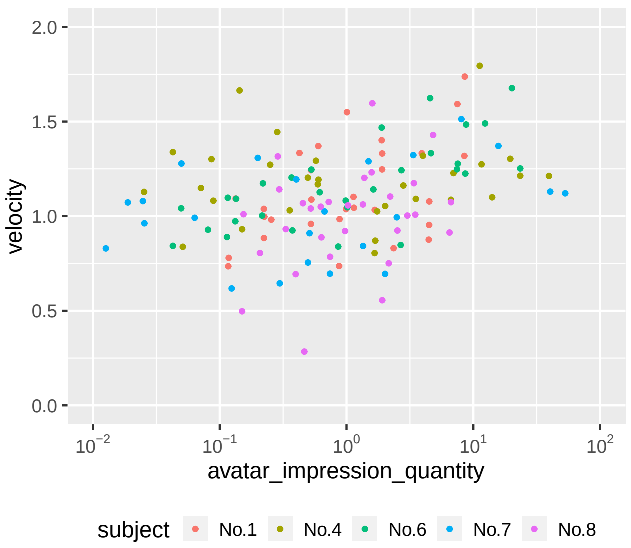


Fig. 7 Scatter plot of the avatar impression quantity and velocity data points

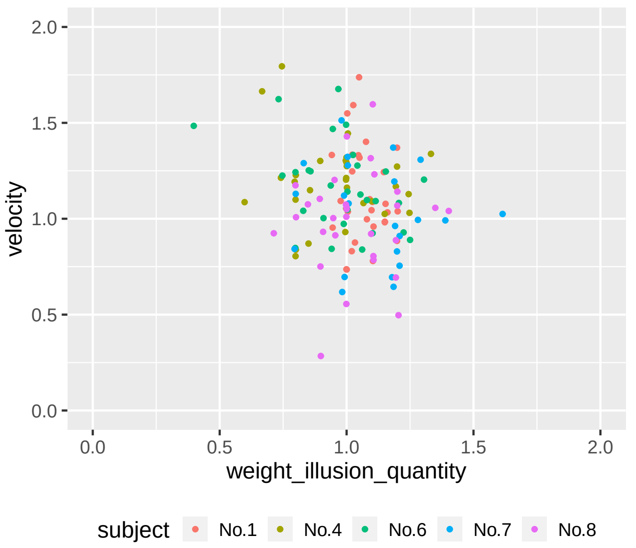


Fig. 9 Scatter plot of the weight illusion quantity and velocity data points

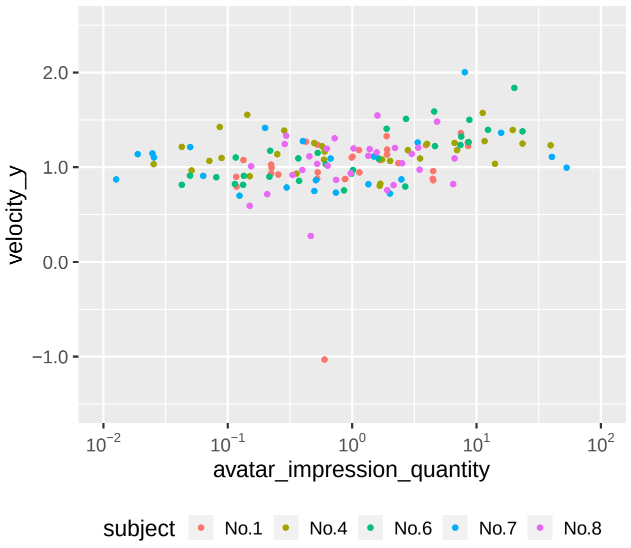


Fig. 8 Scatter plot of the avatar impression quantity and Y component velocity data points

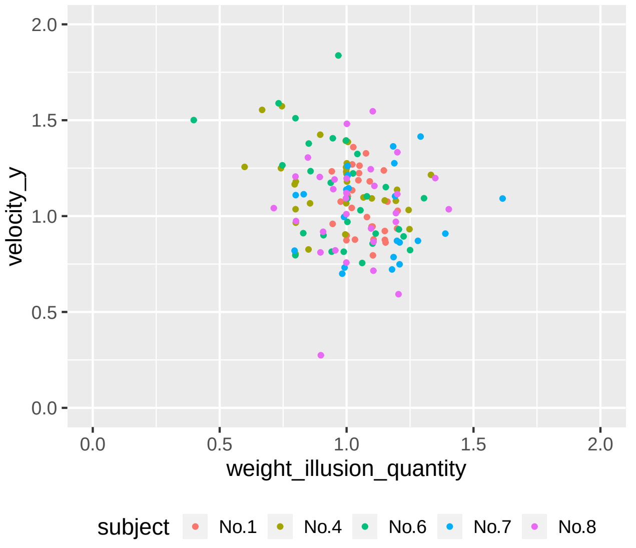


Fig. 10 Scatter plot of the weight illusion quantity and Y component velocity data points

(X: $\rho = .06$, $p = .50$, Z: $\rho = .12$, $p = .14$). A significant positive correlation was found between the Y component and the avatar impression quantity ($\rho = .37$, $p < .001$). The correlation between the avatar impression quantity and the Y component of the velocity data shows the same trend as that between the avatar impression quantity and the velocity data point. Therefore, there is a possibility that there is a relation between the degree of avatar strength impression and the velocity of motion in the Y direction. **Figure 8** shows a scatter plot of the avatar impression quantity and Y component velocity data points for the five subjects.

4.4 Correlation between the weight illusion quantity and the velocity data point

Spearman's rank correlation analysis was performed on the weight illusion quantity and the velocity data point, taking into an account the influence of outliers. No correlation was found ($\rho = -.15$, $p < .01$). However, the five subjects showed negative correlations including a significant negative correlation ($\rho = -.32$, $p < .01$) for the weight illusion quantity and velocity data points. Therefore, it is possible that a relation between the weight illusion quantity and the velocity data point can be obtained. **Figure 9** shows a scatter plot of the weight illusion quantity and velocity data points for the five subjects.

The analysis of the avatar impression quantity and the

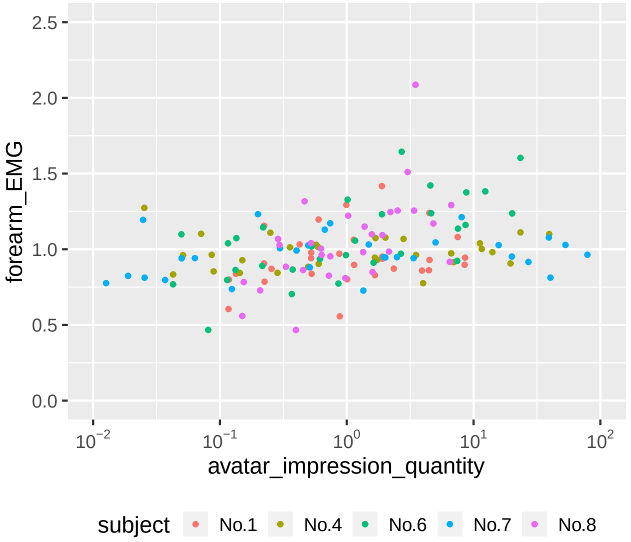


Fig. 11 Scatter plot of the avatar impression quantity and forearm EMG data points

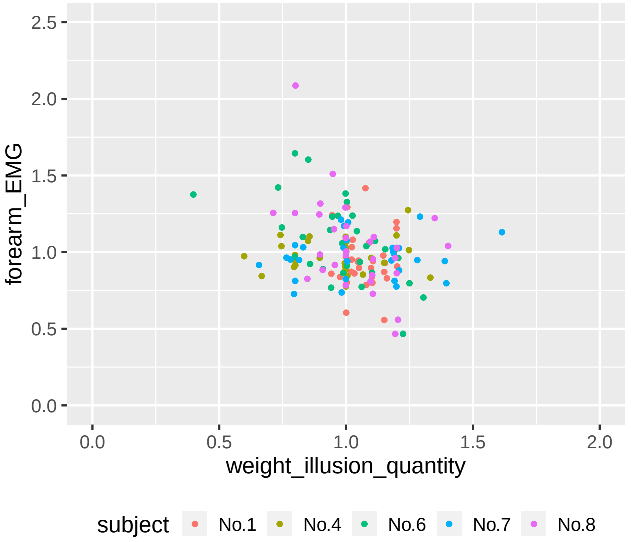


Fig. 13 Scatter plot of the weight illusion quantity and forearm EMG data points

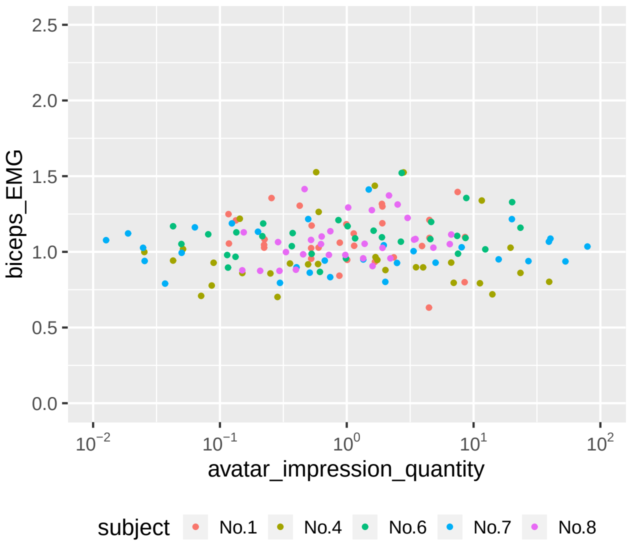


Fig. 12 Scatter plot of the avatar impression quantity and brachial bicep EMG data points

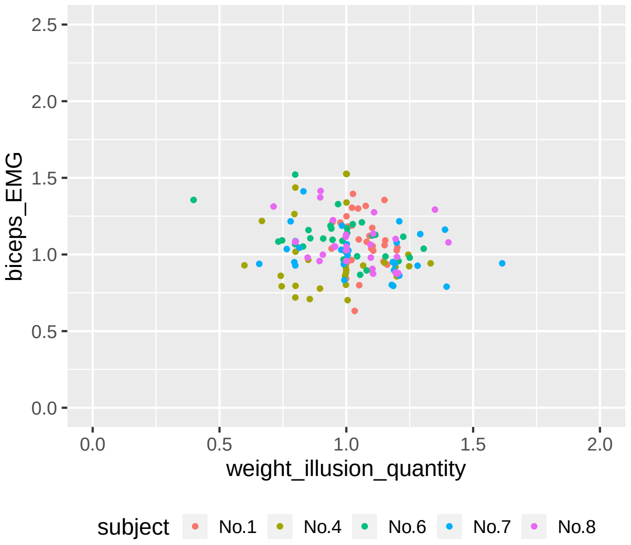


Fig. 14 Scatter plot of the weight illusion quantity and brachial bicep EMG data points

velocity data point confirmed a similar trend both in the avatar impression quantity and velocity data points and in the avatar impression quantity and Y component velocity data points. Therefore, it was thought that the trend could be also appeared in analyzing the weight illusion quantity and velocity data points by decomposing the velocity data into X, Y and Z components. As a result, no significant correlation was found between the X and Z components and the weight illusion quantity (X: $\rho = -.03$, $p = .74$, Z: $\rho = -.09$, $p = .28$). A significant negative correlation was found between the Y component and the weight illusion quantity ($\rho = -.23$, $p < .001$). **Figure 10** shows a scatter plot of the weight illusion quantity and Y component velocity data points for the five subjects.

4.5 Correlation between the avatar impression quantity and the EMG data point

Spearman's rank correlation analysis was performed on the avatar impression quantity and the forearm EMG data point, taking into an account the influence of outliers. A significant positive correlation was obtained ($\rho = .36$, $p < .001$). This is a different trend from a previous study$^{3)}$. **Figure 11** shows a scatter plot of the avatar impression quantity and forearm EMG data points for the five subjects.

Spearman's rank correlation analysis was also performed on the avatar impression quantity and the brachial bicep EMG data point, taking outliers into account. No correlation was found ($\rho = .06$, $p = .48$). This result is similar to a previous study[3]. **Figure 12** shows a scatter plot of the avatar impression quantity and brachial bicep EMG data points for the five subjects.

4.6 Correlation between the weight illusion quantity and the EMG data point

Spearman's rank correlation analysis was performed on the weight illusion quantity and the forearm EMG data point, taking into an account the influence of outliers. A significant negative correlation was obtained ($\rho = -.23$, $p < .01$). This is a different trend from a previous study[3]. **Figure 13** shows a scatter plot of the weight illusion quantity and forearm EMG data points for the five subjects.

Spearman's rank correlation analysis was also performed on the weight illusion quantity and the brachial bicep EMG data point, taking outliers into account. No correlation was found ($\rho = -.17$, $p < .05$). This result is similar to a previous study[3]. **Figure 14** shows a scatter plot of the weight illusion quantity and brachial bicep EMG data points for the five subjects.

5. Discussion

This experiment showed a negative correlation between the avatar impression quantity and the weight illusion quantity, the positive correlation between the avatar impression quantity and the velocity data point, and the negative correlation between the weight illusion quantity and the Y component of the velocity data point. These suggest that the stronger the impression of avatar strength, the lighter the object may be perceived and the faster the object is lifted. The positive correlation was also obtained between the avatar impression quantity and the forearm EMG data point. Mero et al. showed that parameters such as EMG also increased with a running velocity[7]. The positive correlations between the avatar impression quantity and the velocity data point and between the avatar impression quantity and the forearm EMG data point also support this suggestion.

6. Conclusion

In this study, with the aim of clarifying the effects on the body during the weight illusion using a whole-body avatar with different impressions of the avatars, we investigated the relations between the avatar impression quantity, the weight illusion quantity, and the change in motion by measuring the velocity and the EMG during the weight comparison of dumbbells. As a result, it is possible that subjects who have a relation between the avatar impression quantity and the weight illusion quantity also have a relation between the avatar impression quantity and the change in motion. In addition, because the five subjects showed negative correlations for the weight illusion quantity and velocity data points, it is possible that a relation between the weight illusion quantity and the change in motion can be more clearly obtained by increasing the number of subjects.

Acknowledgment

This work was supported by JSPS KAKENHI Grant Number JP23K11195.

References

1) K. SUMIDA, N. OGAWA, T. NARUMI, M. HIROSE: "Proteus Effect of a Muscular Avatar on Weight Perception in Virtual Reality", VRSJ The 25th Annual Conference, 2A2–2 (2020).

2) Y. OKUBO, J. MURAOKA, M. SATO, N. HASHIMOTO: "Influence on Weight Perception by Change of Self-Perception Based on Avatar's Strength", Journal of ITE, Vol. 77, No. 3, pp.394–400 (2023).

3) J. MURAOKA, Y. MURAISHI, K. TOMIOKA, M. SATO, N. HASHIMOTO: "Relation Between Weight Illusion and EMG Signals by Impression Change of the Own Body", ITE Annual Convention 2023, 21C–4 (2023).

4) N. YEE, J. BAILENSON: "The Proteus Effect: The Effect of Transformed Self - Representation on Behavior", Human Communication Research, Vol. 33, pp.271–290 (2007).

5) D. BANAKOU, S. KISHORE. M. SLATER: "Virtually Being Einstein Results in an Improvement in Cognitive Task Performance and a Decrease in Age Bias", Frontiers in Psychology, Vol. 9 (2018).

6) M. CZUB, P. JANETA: "Exercise in Virtual Reality with a Muscular Avatar Influences Performance on a Weightlifting Exercise", Journal of Psychosocial Research on Cyberspace, Vol. 15, No. 3, Article 10 (2022).

7) A. MERO, P. V. KOMI: "Force-, EMG-, and Elasticity-Velocity Relationships at Submaximal, Maximal and Supramaximal Running Speeds in Sprinters", European Journal of Applied Physiology and Occupational Physiology, Vol. 55, pp.553–561 (1986).

(Received November 12, 2023)
(Revised May 28, 2024)

Joichiro MURAOKA
He graduated from the Graduate School of Regional Development and Creativity, Utsunomiya University in 2024. He received a Master of Optical Science and Engineering from Utsunomiya University in 2024.

Kosei TOMIOKA (*Student Member*)
He is a graduate student at the Graduate School of Regional Development and Creativity, Utsunomiya University.

Yusei MURAISHI
He is a graduate student at the Graduate School of Regional Development and Creativity, Utsunomiya University.

Naoki HASHIMOTO
He is a professor at the Graduate School of Informatics and Engineering, the University of Electro-Communications. He received a Doctor of Engineering from Tokyo Institute of Technology in 2001. He is a member of ACM SIGGRAPH, IEICE, ITE and VRSJ.

Mie SATO
She is a professor at the School of Data Science and Management, Utsunomiya University. She received a Doctor of Engineering from Tokyo Institute of Technology in 2001. She is a member of ACM SIGGRAPH and ITE.

IIEEJ Paper -- Special Issue on Journal Track Papers in IEVC2024 --

Smartphone-Based Continuous Authentication Based on Flick Input Features Using Japanese Free Text

Shuto KINOSHITA[†], Yasushi YAMAZAKI[††] (*Member*)

[†] Graduate School of Environmental Engineering, The University of Kitakyushu
[††] Faculty of Environmental Engineering, The University of Kitakyushu

<Summary> With the rapid spread of smartphones, user authentication on smartphones has become essential. However, conventional user authentication methods using PINs, passwords, pattern locks, etc. have a problem in that users are not authenticated continuously after the first successful authentication; therefore, there is a risk that an authenticated smartphone might be used improperly by unauthorized individuals. To address this problem, continuous authentication that verifies the user’s identity without burdening the user by continuously acquiring biometric information from his/her daily smartphone usage has been proposed. Specifically, as smartphones are utilized for various purposes, ensuring the authenticity of the user during text input actions is crucial. Therefore, in this paper, we focus on achieving continuous authentication on smartphones on the basis of flick input behavior, which is a typical text input action by many smartphone users. We aimed to extract user-specific features from Japanese free text input through flick operations in daily usage and evaluated the effectiveness of continuous authentication on the basis of these extracted features. The simulation results indicated that a certain level of authentication accuracy can potentially be maintained by appropriately selecting suitable features.

Keywords: biometrics, smartphones, continuous authentication, behavioral biometrics, flick input, free text

1. Introduction

With the rapid spread of smartphones, there has been an increase in opportunities to handle users’ personal information on smartphones, which makes user authentication technology essential for security and privacy protection. Conventional user authentication methods for smartphones include PINs, passwords, and pattern locks; however, these methods have a problem in that users cannot unlock their smartphones when they forget the passwords and patterns they have set. There is an additional problem in that a malicious third party may impersonate a legitimate user by shoulder hacking or password/pattern guessing from a residual fingerprint on the touch screen. Therefore, biometric authentication technology that utilizes biometric information obtained from sensors installed in smartphones has been attracting attention[1].

However, since smartphones are highly portable and authentication is performed in various environments, changes in the usage environment may affect the reliability of biometric authentication on smartphones. Moreover, in any of the above user authentication methods, the smartphone may be stolen and misused by others after the first authentication success. In fact, one in ten smartphone owners in the U.S. is a victim of theft[2]. As a solution to the first problem, a biometric authentication method was proposed with a function for recognizing the usage environment on the basis of “context awareness,” which is the concept of being able to respond to changes in the situation of people, objects, and environments[3]. It was shown that this method improves the reliability of user authentication by allowing the system to adaptively select an authentication method suitable for the user’s usage environment at the time of authentication. As a solution to the second problem, the implementation of a continuous authentication function[4] was proposed to prevent unauthorized smartphone use after the first authentication success by continuously acquiring biometric information without burdening the user. In one of our previous studies[5], a usage environment-aware continuous authentication system that combines these two functions was proposed, and its application can improve the convenience and security of smartphones. The system runs in the background to continuously authenticate the user.

Continuous authentication has been extensively researched by using various features that can be extracted from smartphone sensors and the devices’ internal information, such as walking patterns, SMS activity, Wi-Fi

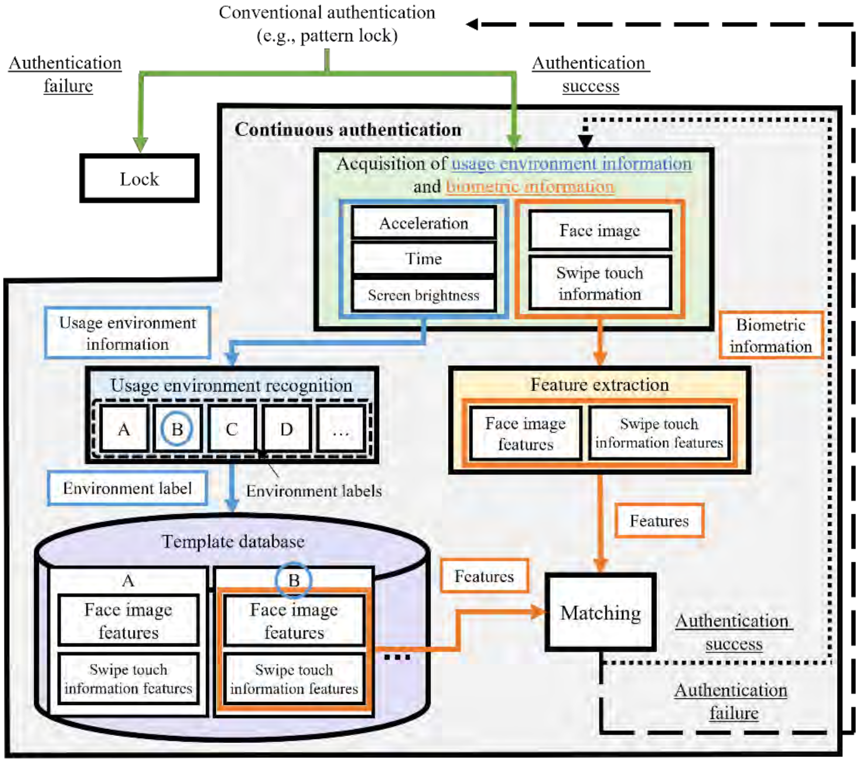


Fig.1 Usage environment-aware continuous authentication system [9)]

usage, application usage history, finger movements during device operation, and changes in the touch area on the screen[6)–9)]. Moreover, as smartphones are utilized for various purposes ranging from SNS and email to password or PIN input, ensuring the authenticity of the user during text input actions is crucial. Therefore, continuous authentication[10)–12)] has been attracting attention, especially for text input actions including flick input, which is used by approximately half of the smartphone users in Japan[13)]. In previous studies, specific text designated by the examiners has been used as verification data. However, requiring users to input specific text for each authentication interrupts their ongoing tasks and is not ideal for user convenience.

Therefore, in this paper, we propose a new continuous authentication method based on flick input behavior. We assume that users input Japanese free text during flick operations in their daily use and that user-specific features, representations of biometric traits unique to each user, are extracted from that input. We utilize these extracted features for continuous authentication and validate the effectiveness of the proposed method.

2. Smartphone-based Continuous Authentication

In this paper, which is a successive study to one of our previous works[5)], we focus on a continuous authentication system with a usage-environment recognition function based on “context awareness”[9)], as shown in **Fig. 1**.

This system utilizes a user authentication function that is commonly installed in smartphones to verify the user’s identity, and when it is confirmed, the system unlocks the device and starts continuous authentication. In the continuous authentication, both the usage environment information and biometric information are acquired at regular intervals because the user’s behavior and the location in which the device is used may change from time to time. As in our previous study[9)], the acquired usage environment information includes acceleration, time, and screen brightness, and the biometric information includes face image and swipe touch information (contact position, contact area, and contact time). After acquiring the usage environment and biometric information, the system recognizes the usage environment by utilizing the usage environment information and selects an environment label corresponding to the recognition result from a template database. Then, the features in the template database included in the environment label are matched with the features extracted from the acquired biometric information. If the authentication succeeds, the system repeats the acquisition of the usage environment and biometric information. If the authentication fails, the system locks the device and goes back to the initial (conventional) user authentication process.

As discussed earlier, ensuring the authenticity of individuals during text input actions on smartphones is an important challenge. Smartphones feature multiple text input methods such as toggle input, free key input, and flick input, and in Japan, flick input is the most widely used. Flick input utilizes a keyboard layout based on the Japanese *Gojuon* table[14] and allows entering the intended characters by tapping each key followed by flicking up, down, left, or right. Since it involves different operations from normal swipe touch input, the individual authentication using swipe touch information introduced in our previous study[9] is not suitable for flick input. Therefore, acquiring flick input features in addition to conventional biometric information needs to be considered for our system.

3. Related Work

Several studies have examined continuous authentication using smartphone flick input features. For example, Izumi et al.[10] conducted feature extraction from flick input in Japanese documents and verified the accuracy of individual identification and user authentication using weighted Euclidean distance and the Array Disorder method. In their experiments, they selected approximately 300 characters per document as keystroke data from a Japanese translation of the novel “Alice’s Adventures in Wonderland” and obtained data from five documents from participants. Subsequently, using the leave-one-out method, they evaluated the accuracy by taking one document from all the profile documents, treating it as an unknown document. Ito et al.[11] categorized each character in the text input data by vowels and performed authentication using five classifiers based on the One Class Support Vector Machine (OCSVM)[15]. In their experiments, they acquired input data from approximately 200 characters in three different texts. Two of the texts were used as training data, while the remaining text was used as test data for authentication. The accuracy was evaluated based on this setup. Motoyama et al.[12] proposed an authentication method intended for short-duration authentication, limiting the number of flick input characters to ten or less. They also compared the identification accuracy using various machine learning algorithms. In their experiments, they obtained the data by inputting two different Japanese text strings of “さとうきびばたけ (sugar cane field)” and “めーるとてにす (mail and tennis)” five times each. The acquired data was randomly divided into training data and evaluation data at a 4:1 ratio for each text string, and machine learning was applied to assess the accuracy.

As described above, previous studies have evaluated authentication accuracy focusing on flick input behavior using various features and training methods for continuous authentication on smartphones. However, Motoyama et al.[12] used the same character texts for both training and validation data, while Izumi et al.[10] and Ito et al.[11] used different character texts for training and validation data, although these texts were still specified by the examiners. In the context of continuous authentication on smartphones, user-specific features should preferably be extracted from normal device interactions without imposing specific authentication actions on users. Therefore, in continuous authentication using flick input features, requiring the user to input specific text when authenticating leads to an interruption of daily-use operation, which is not desirable from the viewpoint of usability.

Considering the above, the main problem with the previous studies on continuous authentication using flick input features on smartphones is the interruption of daily-use operation by requiring users to input specific text each time the authentication is performed.

4. Proposed Method

We address the problem described in the previous section by using a method for extracting user-specific features from users’ flick operations during daily use. In this method, instead of extracting user-specific features from specific text as in the previous studies, we extract them from Japanese free text that users input with flick operations during their daily use and then utilize these features for user authentication. As shown in **Fig. 2**, in the previous methods, the user’s daily-use operation is interrupted by inputting specific text at each authentication timing in continuous authentication. In contrast, the proposed method circumvents the interruption of daily-use operation by performing authentication using user-specific features extracted from the user’s flick operations during daily use such as SNS messages and Web searches.

In this paper, we focus on how to achieve continuous authentication during text input on the basis of flick input behavior in the system shown in Fig. 1. To this end, we added flick input information to the “Biometric information”, and flick input features to the “Feature extraction” in the system. Utilizing these changes, we propose an

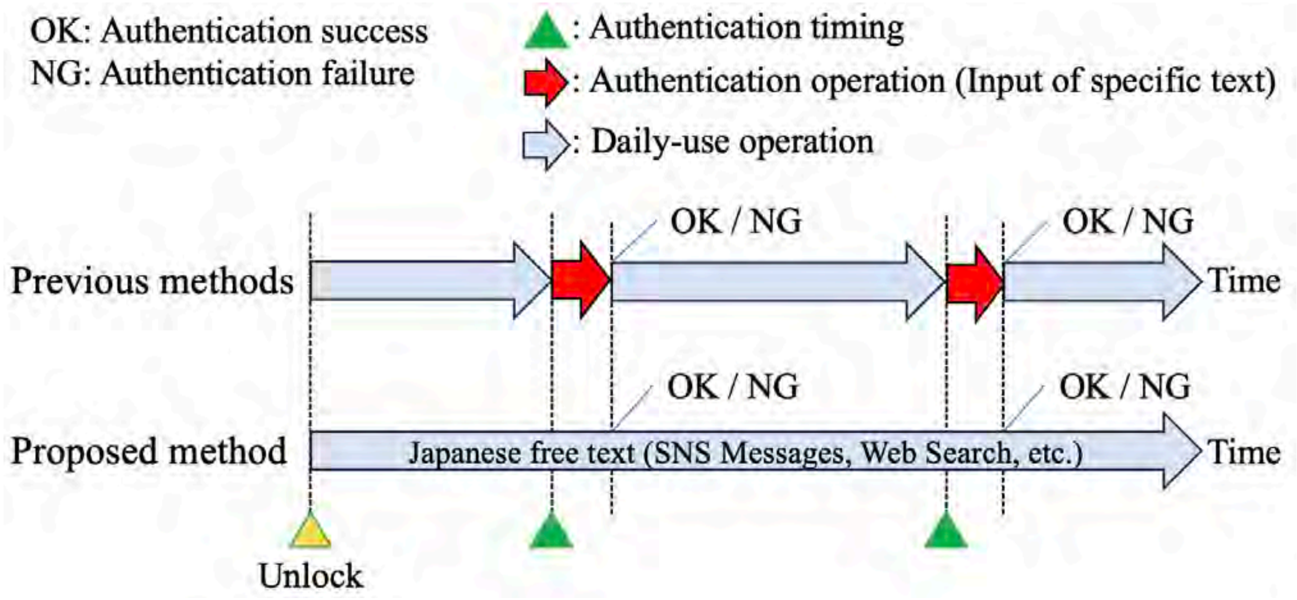

OK: Authentication success
NG: Authentication failure
: Authentication timing
: Authentication operation (Input of specific text)
: Daily-use operation
Previous methods
OK / NG
OK / NG
Time
Proposed method
Japanese free text (SNS Messages, Web Search, etc.)
OK / NG
OK / NG
Time
Unlock

Fig.2 Comparison of continuous authentication between previous and proposed methods

authentication method based on user-specific features extracted from Japanese free text input through daily flick operations.

5. Experiments

We conducted experiments to evaluate the effectiveness of the proposed continuous authentication. All of the experiments were conducted under a protocol approved by the University of Kitakyushu with informed consent from the participants.

5.1 Flick input data collection

We used iPhone 11 and iPhone 12 devices to collect flick input data, targeting a group of 13 participants comprising university students aged 18 to 24 (eight men and five women). As a daily input method, 12 were using flick input and the remaining one was using toggle input. As a daily-use device, 11 were using iOS devices and two were using Android devices for more than 5 years. During text input using flick input, a keyboard (shown in **Fig. 3**(a)) is displayed. By tapping or tapping and then swiping each key (as shown in Fig. 3(b)), specific characters can be entered. First, each participant, while seated, performed flick input to input approximately 300 *hiragana* characters[14] taken from the beginning of the novel “Run, Melos[16]” as a fixed text for training data, which took an average of 4 minutes. Second, they input free text for evaluation data. In this experiment, a self-made application was used to collect the data and both the training and evaluation data were collected on the same day. For the training data, we utilized the fixed text to ensure that enough features were obtained. For the evaluation data, we assumed free text and asked each participant to input five pieces of arbitrary text in *hiragana* characters for each of the following categories I–III.

I. SNS Messages: Recent messages sent on social networking platforms such as LINE, Instagram DM, etc.

II. Web Search: Words or phrases you want to search for.

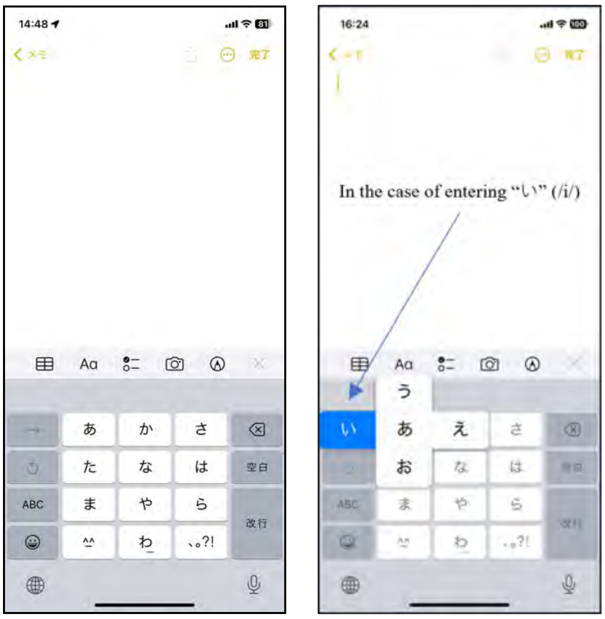
III. To-Do List: Plans and tasks scheduled for today and beyond.

5.2 Feature extraction

We extracted features from the collected flick input data. Referring to previous studies[10)–12)], we used the features listed in **Table 1**.

5.3 Outlier removal

The feature vectors created by arranging the extracted features may contain outliers, which could potentially affect the classification accuracy. We therefore removed outliers using the local outlier factor (LOF)[17] on the basis of past research[11]. LOF is an outlier detection algorithm that identifies anomalies within a dataset by comparing the density of data points to their surroundings. It calculates the local anomaly factor for each data point and is used to identify abnormal points in the dataset. Note that outliers



(a) Initial state **(b)** During flick input

Fig.3 Overview of flick input keyboard

were removed for each type of feature in this experiment. In this case, the parameter k, which specifies the number of data points to be considered as neighbors for each data point, was set to 7, and the LOF score threshold was set to 0.08. These parameters were determined on the basis of the results of preliminary experiments.

5.4 Feature selection

Among the features listed in Table 1, 11 types (indicated by numbers 1, 2, 3, 4, 11, 12, 13, 14, 15, 16, 18) were used for tap actions (corresponding to vowel "a"), and all 18 types were used for flick actions (corresponding to vowel "i," "u," "e," and "o"). We examined feature vectors comprising all combinations of these 11 or 18 types and searched for which combinations of feature vectors were effective for continuous authentication. The combinations of feature vectors resulted in 2,047 variations for tap actions and 262,143 variations for flick actions.

5.5 Training

In the continuous authentication envisioned in this paper, training data from persons other than the user themselves is difficult to obtain. Additionally, creating 46 different classifiers for each *hiragana* character would incur significant training costs. Therefore, with reference to the literature[11], we created five classifiers using OCSVM for each vowel in the input characters (see **Fig. 4**). As a result, we reduced the training cost and conducted user authentication solely on the basis of the user's own data.

Table 1 Feature types

No.	Feature
0	Vowel: Vowel in the input character (used for classifier selection)
1	Current Character: Current input character
2	Previous Character: Character input at the previous time
3	Current Consonant: Consonant of the current input character
4	Previous Consonant: Consonant of the character input at the previous time
5	Flick X Displacement: Displacement of the X-coordinate from the start to the end of the flick
6	Flick Y Displacement: Displacement of the Y-coordinate from the start to the end of the flick
7	Flick Distance: Length of the trajectory from the start to the end of the flick
8	Flick Curvature: Curvature of the flick
9	Flick Speed: Average speed from the start to the end of the flick
10	Flick Angle: Angle between the line connecting the start and end points of the flick and the X (Y) axis
11	Input Time: Time from the start to the end of the flick
12	Time Gap Before: Time from the end of the previous input to the start of the current input
13	Time Gap After: Time from the end of the current input to the start of next input
14	3-Axis Composite Acceleration: Average value of composite acceleration in three axes during input
15	3-Axis Composite Angular Velocity: Average value of composite angular velocity in three axes during input
16	Touch Radius: Radius of the circle when touching the screen at the beginning of the flick
17	Average Touch Radius: Average radius of the touched area from the start to the end of the flick
18	Input Interval: Time duration from pressing one key to releasing the next key

5.6 Evaluation method

As described earlier, we used fixed text comprising approximately 300 characters as training data and free text as evaluation data. In the experiment, we used the training data obtained from each participant to create classifiers for each vowel in the input characters. Subsequently, we applied the created classifiers to the user's evaluation data and evaluation data from individuals other than the user for authentication. Here, the authentication was performed on a character-by-character basis. We evaluated the authentication accuracy by

$$\text{Accuracy} = \frac{\mathrm{TP}+\mathrm{TN}}{\mathrm{TP}+\mathrm{TN}+\mathrm{FP}+\mathrm{FN}} \times 100 \ \ [\%]. \qquad (1)$$

TP (True Positives) represents the number of times the input actions of the user were correctly identified as belonging to the user. TN (True Negatives) represents the number of times the input actions of others were correctly identified as belonging to others. FP (False Positives) is the number of times the input actions of others were incorrectly classified as belonging to the user. FN (False Negatives) is the number of times the input actions of the user were incorrectly classified as belonging to others. Note that the values of TP, TN, FP, and FN were calculated for all participants' data by swapping the training data.

5.7 Experimental results

Authentication accuracy was evaluated on the basis of the method outlined in **5.6**. First, we calculated the authentication accuracy for each feature vector. In this case, the parameters for OCSVM were set with the *nu* value of 0.9 to control the proportion of outliers and the kernel function of *rbf* (radial basis function)[15]. These parameters were determined on the basis of the results of preliminary experiments. **Table 2** presents the top five combinations of feature vectors that showed the highest authentication accuracy for each vowel under these settings. The numbers in the "Feature vector combinations" column in Table 2 correspond to the feature numbers listed in Table 1. Additionally, the results of comparing the authentication accuracy obtained in this experiment with those from previous studies[10)–12)] are presented in **Table 3**. The average False Reject Rate (FRR) and False Accept Rate (FAR) for the feature vector combinations shown in Tables 2 and 3 were approximately 28% and 7%, respectively. In this experiment, we did not consider feature variation over time when there is a time gap between the collection of training and evaluation data, nor the influence of user's proficiency in flick input on authentication accuracy.

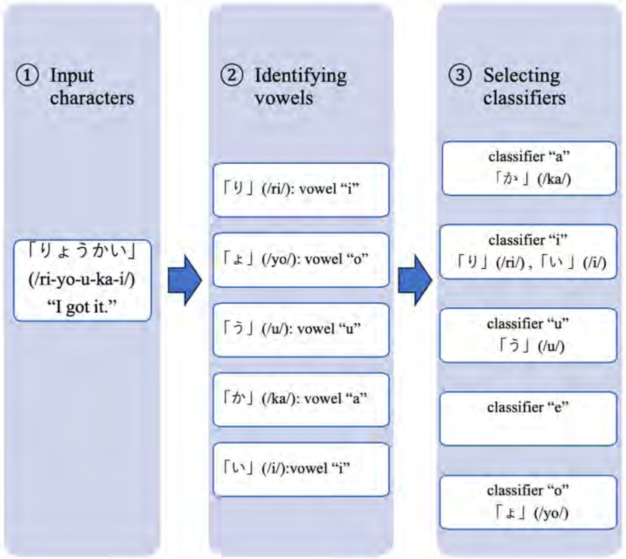


Fig.4 Classifier determination for input characters

5.8 Consideration

From Table 2, among the combinations of feature vectors that exhibited higher authentication accuracy, no. 11 was the most frequently included, followed by nos. 4, 15, and 16. Nos. 11 (Input Time) and 16 (Touch Radius) were also frequently included, which suggests that individual features on the touchscreen during flick input operations are reflected in the input time for each character and the radius of the finger touching the screen at the beginning of the flick. Moreover, the abundance of no. 4 (Previous Consonant) indicates a substantial influence of the character entered prior to the current one, meaning the position of the finger before input. Furthermore, the prevalence of no. 15 (3-Axis Composite Angular Velocity) suggests that the features related to flick input operations manifest not only on the touchscreen but also in the device's motion. This observation is further supported by the relatively high occurrence of no. 14 (3-Axis Composite Acceleration).

According to Table 3, the authentication accuracy is approximately 5% lower in this experiment than in the previous studies. This difference can be attributed to the variations in the data used. In the previous studies, both training and validation data were based on the same dataset or specific texts chosen by the examiners. In contrast, our experiment utilized free-text input. Moreover, our experiment assumed the text that users typically input and

Table 2 Combinations of feature vectors with high authentication accuracy for each vowel and their authentication accuracy

Vowel	Feature vector combinations	Accuracy (%)
“a”	2, 4, 11, 14, 15, 16	91.46
	2, 4, 11, 15, 16, 18	91.46
	2, 4, 11, 14, 15, 16, 18	91.46
	2, 4, 11, 15, 16	91.42
	2, 11, 14, 15, 16	91.42
“i”	1, 2, 11, 13, 15, 17	91.30
	1, 2, 3, 11, 13, 15, 17	91.30
	1, 2, 11, 13, 15, 16, 17	91.30
	1, 2, 3, 4, 11, 13, 15, 17	91.30
	1, 2, 3, 8, 11, 13, 15, 17	91.30
“u”	16, 17	91.89
	1, 4, 7, 10, 11, 16, 17	90.94
	1, 4, 7, 10, 11, 15, 16, 17	90.89
	1, 4, 7, 10, 11, 15, 16, 17, 18	90.89
	1, 4, 7, 10, 11, 17	90.83
“e”	3, 6, 10, 11, 13, 15, 16, 17	92.54
	3, 4, 6, 10, 12, 13, 15, 16, 17	92.54
	3, 6, 8, 10, 11, 13, 15, 16, 17	92.54
	3, 6, 9, 10, 11, 13, 15, 16, 17	92.54
	3, 6, 10, 11, 13, 14, 15, 16, 17	92.54
“o”	4, 6, 10, 11, 12	92.74
	4, 6, 8, 10, 11, 12	92.74
	4, 6, 9, 10, 11, 12	92.74
	4, 6, 10, 11, 12, 14	92.74
	4, 6, 10, 11, 12, 15	92.74

utilized different text for each participant as validation data, resulting in relatively shorter input texts. These factors are likely responsible for the observed decrease in authentication accuracy when comparing this study with previous ones.

On the basis of the above results, in the context of continuous authentication on smartphones using flick input features, authentication accuracy tends to be lower when free text is used than when fixed text is used. However, by appropriately selecting suitable features, a certain level of authentication accuracy can potentially be maintained.

6. Conclusion

In this paper, we focused on continuous authentication on smartphones utilizing flick input features. We aimed to extract user-specific features from Japanese free text input through flick operations in daily use and evaluated the effectiveness of continuous authentication on the basis of these extracted features.

Table 3 Comparison of authentication accuracy

	Accuracy (%)	
Proposed	Vowel “a”	91.46
	Vowel “i”	91.30
	Vowel “u”	91.89
	Vowel “e”	92.54
	Vowel “o”	92.74
Izumi et al.[10)	96	
Ito and Shiraishi[11)	Vowel “a”	98.69
	Vowel “i”	95.04
	Vowel “u”	92.25
	Vowel “e”	92.09
	Vowel “o”	94.72
Motoyama et al.[12)	Multi-class classification	94.1
	2-class classification	97.8

Future work will include an evaluation of the entire performance of the proposed system that includes the flick input features described in this paper. Our principal areas of study will include the selection of text to be used as training data, experimental evaluations with an increased number of participants and variation in features over time, and evaluations considering the balance between resource consumption and authentication accuracy through the appropriate selection of authentication timing and methods to overcome the issue of resource consumption during system operation in practical implementation[18)].

Acknowledgment

This work was supported by JSPS KAKENHI Grant Number JP20K11814.

References

1) P. A. Tresadern, C. McCool, N. Poh, P. Matejka, A. Hadid, C. Levy, T. F. Cootes, S. Marcel: “Mobile Biometrics: Combined Face and Voice Verification for A Mobile Platform”, IEEE Pervasive Computing, Vol.12, No.1, pp. 79–87 (2013).

2) Lookout, “Phone Theft in America: What Really Happens When Your Phone Gets Grabbed”, https://www.lookout.com/blog/phone-theft-in-america (2014).

3) Y. Yamazaki, T. Ohki: “Toward More Secure and Convenient User Authentication in Smart Device Era”, IEICE Trans. on Information & Systems, Vol.E100-D, No.10, pp. 2391–2398 (2017).

4) V. M. Patel, R. Chellappa, D. Chandra, B. Barbello: “Continuous User Authentication on Mobile Devices – Recent Progress and Remaining Challenges”, IEEE Signal Processing Magazine, Vol.33, No.4, pp. 49–61 (2016).

5) T. Agawa, T. Higashi, Y. Yamazaki, T. Ohki: “A Study on Continuous User Authentication for Smart Devices Considering Resource Consumption (in Japanese)”, IEICE technical report, Vol.117, No.236, Biox2017-25, pp. 1–6 (2017).

6) M. Ehatisham-ul-Haq, M.A. Azam, U. Naeem, Y. Amin, J. Loo: “Continuous Authentication of Smartphone Users Based on Activity Pattern Recognition Using Passive Mobile Sensing”, Journal of Network and Computer Applications, Vol.109, pp.24–35 (2018).

7) F. Yao, S. Y. Yerima, B. Kang, S. Sezer: “Continuous Implicit Authentication for Mobile Devices Based on Adaptive Neuro-Fuzzy Inference System”, Proc. of 2017 International Conference on Cyber Security and Protection of Digital Services (Cyber Security), pp.1–7 (2017).

8) T. Karanikiotis, M. D. Papamichail, K. C. Chatzidimitriou, N. -C. I. Oikonomou, A. L. Symeonidis, S. K. Saripalle: “Continuous Implicit Authentication through Touch Traces Modelling”, Proc. of 2020 IEEE 20th International Conference on Software Quality, Reliability and Security (QRS), pp.111–120 (2020).

9) S. Kinoshita, Y. Watanabe, Y. Yamazaki: “Continuous Authentication for Smartphones Using Face Images and Touch-Screen Operation”, Proc. of 2022 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), pp. 1556–1560 (2022).

10) M. Izumi, T. Samura, H. Nishimura: “Keystroke Dynamics with Flick Input on Smartphone in Japanese Free Text Typing (in Japanese)”, Proc. of the Information Processing Society of Japan Kansai Branch Conference 2013, E-15 (2013).

11) S. Ito, Y. Shiraishi: “Proposal of Continuous Authentication Method Focusing on The Features of Smartphone Flick Input Method (in Japanese)”, Proc. of the 25th Multimedia Communication and Distributed Processing System Workshop, pp.1–8 (2017).

12) W. Motoyama, S. Fukumoto, M. Kashima, K. Sato, M. Watanabe: “Studies on Biometrics Authentication by Behavioral Features at Flick Input (in Japanese)”, IEICE technical report, Vol.119, No.445, BioX2019-68, pp. 35–40 (2020).

13) JustSystems Corporation: “2 out of 3 Teenagers Use Flick Input to Input Text on Smartphones”,
https://prtimes.jp/main/html/rd/p/000000098.000007597.html (2015).

14) F. Bond, T. Baldwin: “Introduction to Japanese Computational Linguistics”, in Readings in Japanese Natural Language Processing, edited by F. Bond, T. Baldwin, K. Inui, S. Ishizaki, H. Nakagawa, A. Shimazu, pp. 1–28, CSLI Publication (2016).

15) B. Schoelkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, R. C. Williamson: “Estimating The Support of A High-Dimensional Distribution”, Neural Computation, Vol.13, No.7, pp. 1443–1471 (2001).

16) Aozorabunko: “Hashire Merosu [Run, Melos]”,
https://www.aozora.gr.jp/cards/000035/files/1567_14913.html (2000).

17) M. M. Breunig, H. P. Kriegel, R. T. Ng, J. Sander: “LOF: Identifying Density-Based Local Outliers”, ACM SIGMOD Record, Vol.29, No.2, pp.93–104 (2000).

18) Y. Watanabe, Y. Yamazaki: “A Study on Continuous Authentication Considering Implementation on Smart Devices (in Japanese)”, The Journal of the Institute of Image Electronics Engineers of Japan, Vol.52, No.1, pp. 205–213 (2023).

(Received November 12, 2023)
(Revised March 27, 2024)

Shuto KINOSHITA

He graduated from the Department of Information and Media Engineering at the University of Kitakyushu in 2022. He completed the master's program in Computer Science Course at the Graduate School of Environmental Engineering at the University of Kitakyushu in 2024. He engaged in research on biometric authentication on smartphones during the academic years.

Yasushi YAMAZAKI (*Member*)

He received the B.E. and M.E. degrees in electronics and communication engineering from Waseda University, Tokyo, Japan, in 1993 and 1995, respectively, and the Ph.D. degree in electronics, information and communication engineering from Waseda University in 1998. He is currently a professor in the Department of Information Systems Engineering at the University of Kitakyushu, Fukuoka, Japan. He received the IIEEJ paper award in 2004 and the IIEEJ best paper award in 2012. His research interests include biometrics and information security. He is a member of IIEEJ, IEICE, IPSJ, IEEE and ACM.

Call for Papers

Special Issue on
Image Electronics Technologies Related to VR/AR/MR/XR

IIEEJ Editorial Committee

In recent years, Virtual Reality (VR), Augmented Reality (AR), Mixed Reality (MR), Extended/Cross Reality (XR) have been applied to systems of various fields, and have brought revolutionary innovation in them. The applied area includes Entertainment, Medical & Health Care, Education, Practice & Disciplining, Manufacturing, Remote Work & Meeting, and so on, and has been extended even for the purposes of Art and Environment Preservation. Especially, VR technology will provide fully virtual environment and invite the users to another world in order to experience unfeasible experiments. Also, AR technology will improve the efficiency of daily task by shortening the access time to the desired information, and will empower the process to understand the individual utilized context at the same time.

On the other hand, the long-time effect of such technologies on human senses and mental states has been gathering wide interest, and how to coexist with such technologies will be the future big theme for human beings.

In this special issue, we invite various categories (Ordinary paper, Short paper, System development paper, Practice Oriented Paper) of papers on designing and implementing technologies that support VR/AR/MR/XR, and on the investigation and evaluation of such technologies and systems being developed. We look forward to receiving your contributions.

1. Topics covered include but not limited to

・Technologies and Systems on VR/AR/MR/XR, Meta-verse
・User Interface，Human Computer Interaction, Mutual Interaction
・Computer Graphics, Image Processing, Image Communication
・Computer Vision, Image Understanding, Pattern Recognition, Machine Learning, AI
・Collaboration, Education, Training, Medical Application, Remote Work/Meeting/Entertainment

2. Treatment of papers

Submission paper style format and double-blind peer review process are the same as the regular paper. If the number of accepted papers is less than the minimum number for the special issue, the acceptance paper will be published as the regular contributed paper. We ask for your understanding and cooperation.

3. Publication of Special Issue:

IIEEJ Transactions on Image Electronics and Visual Computing Vol.13, No.1 (June 2025)

4. Submission Deadline:

Tuesday, October 31, 2024

5. Contact details for Inquiries:

IIEEJ Office E-mail: hensyu@iieej.org

6. Online Submission URL: http://www.editorialmanager.com/iieej/

Revised: January 6, 2017

Revised: July 6, 2018

Guidance for Paper Submission

1. Submission of Papers

(1) Preparation before submission

- The authors should download “Guidance for Paper Submission” and “Style Format” from the “Academic Journals”, “English Journals” section of the Society website and prepare the paper for submission.
- Two versions of “Style Format” are available, TeX and MS Word. To reduce publishing costs and effort, use of TeX version is recommended.
- There are four categories of manuscripts as follows:
  - Ordinary paper: It should be a scholarly thesis on a unique study, development or investigation concerning image electronics engineering. This is an ordinary paper to propose new ideas and will be evaluated for novelty, utility, reliability and comprehensibility. As a general rule, the authors are requested to summarize a paper within eight pages.
  - Short paper: It is not yet a completed full paper, but instead a quick report of the partial result obtained at the preliminary stage as well as the knowledge obtained from the said result. As a general rule, the authors are requested to summarize a paper within four pages.
  - System development paper: It is a paper that is a combination of existing technology or it has its own novelty in addition to the novelty and utility of an ordinary paper, and the development results are superior to conventional methods or can be applied to other systems and demonstrates new knowledge. As a general rule, the authors are requested to summarize a paper within eight pages.
  - Data Paper: A summary of data obtained in the process of a survey, product development, test, application, and so on, which are the beneficial information for readers even though its novelty is not high. As a general rule, the authors are requested to summarize a paper within eight pages.
- To submit the manuscript for ordinsry paper, short paper, system development paper, or data paper, at least one of the authors must be a member or a student member of the society.
- We prohibit the duplicate submission of a paper. If a full paper, short paper, system development paper, or data paper with the same content has been published or submitted to other open publishing forums by the same author, or at least one of the co-authors, it shall not be accepted as a rule. Open publishing forum implies internal or external books, magazines, bulletins and newsletters from government offices, schools, company organizations, etc. This regulation does not apply to a preliminary draft to be used at an annual meeting, seminar, symposium, conference, and lecture meeting of our society or other societies (including overseas societies). A paper that was once approved as a short paper and being submitted again as the full paper after completion is not regarded as a duplicate submission.

(2) Submission stage of a paper

- Delete all author information at the time of submission. However, deletion of reference information is the author’s discretion.
- At first, please register your name on the paper submission page of the following URL, and then log in again and fill in the necessary information. Use the “Style Format” to upload your manuscript. An applicant should use PDF format (converted from dvi of TeX or MS Word

format) for the manuscript. As a rule, charts (figures and tables) shall be inserted into the manuscript to use the “Style Format”. (a different type of data file, such as audio and video, can be uploaded at the same time for reference.)

http://www.editorialmanager.com/iieej/

- If you have any questions regarding the submission, please consult the editor at our office.

Contact:
Person in charge of editing
The Institute of Image Electronics Engineers of Japan
3-35-4-101, Arakawa, Arakawa-Ku, Tokyo 116-0002, Japan
E-mail: hensyu@iieej.org
Tel: +81-3-5615-2893, Fax: +81-3-5615-2894

2. Review of Papers and Procedures

(1) Review of a paper

- A manuscript is reviewed by professional reviewers of the relevant field. The reviewer will deem the paper “acceptance”, “conditionally acceptance” or “returned”. The applicant is notified of the result of the review by E-mail.
- Evaluation method

Ordinary papers are usually evaluated on the following criteria:

✓ Novelty: The contents of the paper are novel.
✓ Utility: The contents are useful for academic and industrial development.
✓ Reliability: The contents are considered trustworthy by the reviewer.
✓ Comprehensibility: The contents of the paper are clearly described and understood by the reviewer without misunderstanding.

Apart from the novelty and utility of an ordinary paper, a short paper can be evaluated by having a quickness on the research content and evaluated to have new knowledge with results even if that is partial or for specific use.

System development papers are evaluated based on the following criteria, apart from the novelty and utility of an ordinary paper.

✓ Novelty of system development: Even when integrated with existing technologies, the novelty of the combination, novelty of the system, novelty of knowledge obtained from the developed system, etc. are recognized as the novelty of the system.
✓ Utility of system development: It is comprehensively or partially superior compared to similar systems. Demonstrates a pioneering new application concept as a system. The combination has appropriate optimality for practical use. Demonstrates performance limitations and examples of performance of the system when put to practical use.

Apart from the novelty and utility of an ordinary paper, a data paper is considered novel if new deliverables of test, application and manufacturing, the introduction of new technology and proposals in the worksite have any priority, even though they are not necessarily original. Also, if the new deliverables are superior compared to the existing technology and are useful for academic and industrial development, they should be evaluated.

(2) Procedure after a review

- In case of acceptance, the author prepares a final manuscript (as mentioned in 3.).
- In the case of acceptance with comments by the reviewer, the author may revise the paper in consideration of the reviewer’s opinion and proceed to prepare the final manuscript (as

mentioned in 3.).

- In case of conditional acceptance, the author shall modify a paper based on the reviewer's requirements by a specified date (within 60 days), and submit the modified paper for approval. The corrected parts must be colored or underlined. A reply letter must be attached that carefully explains the corrections, assertions and future issues, etc., for all of the acceptance conditions.
- In case a paper is returned, the author cannot proceed to the next step. Please look at the reasons the reviewer lists for the return. We expect an applicant to try again after reviewing the content of the paper.

(3) Review request for a revised manuscript

- If you want to submit your paper after conditional acceptance, please submit the reply letter to the comments of the reviewers, and the revised manuscript with revision history to the submission site. Please note the designated date for submission. Revised manuscripts delayed more than the designated date be treated as new applications.
- In principle, a revised manuscript willl be reviewed by the same reviewer. It is judged either accceptance or returned.
- After the judgment, please follow the same procedure as (2).

3. Submission of final manuscript for publication

(1) Submission of a final manuscript

- An author, who has received the notice of "Acceptance", will receive an email regarding the creation of the final manuscript. The author shall prepare a complete set of the final manuscript (electronic data) following the instructions given and send it to the office by the designated date.
- The final manuscript shall contain a source file (TeX edition or MS Word version) and a PDF file, eps files for all drawings (including bmp, jpg, png), an eps file for author's photograph (eps or jpg file of more than 300 dpi with length and breadth ratio 3:2, upper part of the body) for authors' introduction. Please submit these in a compressed format, such as a zip file.
- In the final manuscript, write the name of the authors, name of an organizations, introduction of authors, and if necessary, an appreciation acknowledgment. (cancel macros in the Style file)
- An author whose paper is accepted shall pay a page charge before publishing. It is the author's decision to purchase offprints. (ref. page charge and offprint price information)

(2) Galley print proof

- The author is requested to check the galley (hard copy) a couple of weeks before the paper is published in the journal. Please check the galley by the designated date (within one week). After making any corrections, scan the data and prepare a PDF file, and send it to our office by email. At that time, fill in the Offprint Purchase Slip and Copyright Form and return the scanned data to our office in PDF file form.
- In principle, the copyrights of all articles published in our journal, including electronic form, belong to our society.
- You can download the Offprint Purchase Slip and the Copyright Form from the journal on our homepage. (ref. Attachment 2: Offprint Purchase Slip, Attachment 3: Copyright Form)

(3) Publication

- After final proofreading, a paper is published in the Academic journal or English transaction (both in electronic format) and will also be posted on our homepage.

IIEEJ Trans. on Image Electronics and Visual Computing

Vol. 12 No. 1 2024

Editor in Chief: Osamu Uchida
The Institute of Image Electronics Engineers of Japan
3-35-4-101, Arakawa,　Arakawa-ku, Tokyo 116-0002, Japan

Print: ISSN 2188-1898
Online: ISSN 2188-1901
CD-ROM: ISSN 2188-191x