HowToEat: Exploring Human-Object Interaction and Eating Action in Eating Scenarios

Yingcheng Wang
Department of Informatics, The University of Electro-Communications
Tokyo, Japan
wang-y@mm.inf.uec.ac.jp

Junwen Chen
Department of Informatics, The University of Electro-Communications
Tokyo, Japan
chen-j@mm.inf.uec.ac.jp

Keiji Yanai
Department of Informatics, The University of Electro-Communications
Tokyo, Japan
yanai@cs.uec.ac.jp

ABSTRACT

Recently, the analysis of multimedia of eating and diet has become a new trend in research. Detecting eating activities in videos and images is a basic requirement for further analysis. However, existing human-centric action detection tasks, such as human-object interaction detection and hand-object interaction detection lack the data in eating scenarios and annotations of eating actions. To fill this gap in research, we introduce a new large-scale dataset, HowToEat, which contains 66 days of videos in 12 eating scenarios, and 95k images with automatic annotations of hand-object interactions and eating actions. Based on the dataset, we propose an eating analysis system, which uses a single model to detect hand-object interaction and eating action at the same time.

CCS CONCEPTS

• **Computing methodologies** → **Activity recognition and understanding**.

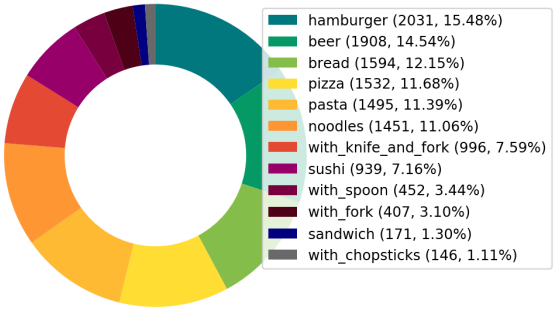
KEYWORDS

dataset, eating action recognition, human-object interaction detection, hand-object interaction detection, object detection

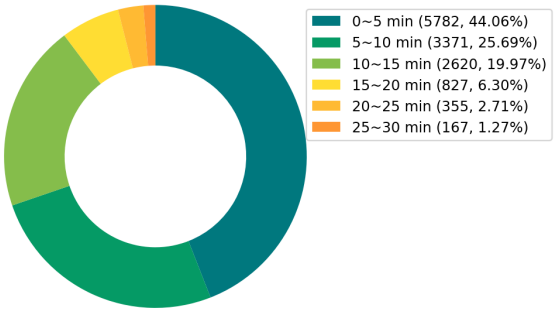
ACM Reference Format:
Yingcheng Wang, Junwen Chen, and Keiji Yanai. 2023. HowToEat: Exploring Human-Object Interaction and Eating Action in Eating Scenarios. In *Proceedings of the 8th International Workshop on Multimedia Assisted Dietary Management (MADiMa '23), October 29, 2023, Ottawa, ON, Canada.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3607828.3617790

1 INTRODUCTION

Eating scenarios are ubiquitous in real life and are an inevitable part of human daily activities. By monitoring various dietary behaviors in eating scenarios, we can glean insights into healthy eating habits, and disease prevention. However, in the field of computer vision, although there are tasks for object detection [1, 1, 19, 22, 28, 29, 40] and human-object interactive detection [2–4, 10, 21, 33, 36], the datasets for these tasks often only allow us to observe and understand behaviors in eating from a partial perspective, unable to provide all the information necessary for dietary behavior detection.



(a) The number of videos in each eating scenario.



(b) The number of videos by different durations.

Figure 1: The duration distribution of collected videos.

For example, in the **H**uman-**O**bject **I**nteraction (HOI) detection task, an HOI instance $\{B^{(h)}, B^{(o)}, C_o, C_a\}$, consists of a human bounding box $B^{(h)}$, an object bounding box $B^{(o)}$ with object class C_o, and an action class C_a between the human and the object. However, HOI task's datasets [2, 12] only focus on the interaction of a person's whole body in a coarse bounding box, but ignore interactions between specific body parts and objects. For instance, in the case of eating food, we only know it is a "person" interacting

MADiMa '23, October 29, 2023, Ottawa, ON, Canada

ACM ISBN 979-8-4007-0284-6/23/10…$15.00
https://doi.org/10.1145/3607828.3617790

with "food", without knowing which parts of the person are interacting with the "food". This global perspective fails to adequately capture key information in eating behaviors, such as the type of food, the manner of consumption, the way food contacts people, etc., all of which are important factors in understanding dietary behavior. On the other hand, the hand-object interaction detection task (e.g., EPIC KITCHEN [6], 100DOH [31]) focuses on the hand of a person. However, EPIC KITCHEN collects videos in a first-person vision and lacks labels of other parts of the person. 100DOH lacks images of eating scenarios and only hands (without interaction) or hand-object pairs are annotated. Thus, a more specialized and comprehensive dataset to study eating scenarios is necessary.

To this end, we introduce a new dataset specifically designed for analyzing eating scenarios, named HowToEat. We collect a large amount of video data from YouTube [1] and leverage an HOI detection method to automatically extract images containing eating behaviors of hand-object interaction for our research. The constructed HowToEat dataset includes the eating action annotation of the face and the hand-object interaction annotation of the hand-object pair to better adapt to the eating scenario. In summary, our contributions are four-fold:

- First, we introduce a new dataset HowToEat, which is specifically designed for eating scenarios. This dataset collects multi-task annotations on both eating action and hand-object interaction, providing a rich resource for exploring complex behaviors in eating scenarios.
- Second, we adapt an HOI detection method to detect hand-object interaction and achieve notable performance.
- Third, We build a novel dataset and corresponding detection methods, especially for the eating action of the face.
- Lastly, we present an eating analysis system that is capable of simultaneously detecting hand-object interactions and eating actions. This system leverages the HowToEat dataset and a powerful HOI detection method, providing a comprehensive solution for studying eating behaviors.

2 RELATED WORK

2.1 Object detection

Object detection is a fundamental problem in computer vision, aiming to classify and locate objects in images or videos. There are many common subtasks in object detection, for example, general object detection [8, 23, 32] focuses on identifying various types of objects in the image (e.g., "cars", "person"); hand detection [11, 16] concentrates on detecting and locating the position of the hand, including key points such as fingers and palms, which has wide applications in fields like virtual reality and human-computer interaction; and face detection [15, 35] focuses on the detection and identification of faces, such as recognizing human faces and detecting facial feature points, often used in scenes like biometric identification and expression recognition.

Given an input (usually an image or a video frame), the output of the object detection task is the categories of all objects of interest and their location information. This usually manifests as a bounding box and a category label. Currently, common object detection methods include the R-CNN series [9, 29], YOLO series [27, 28], DETR series [1, 19, 24, 40], etc. These methods typically extract the feature maps by using a pre-trained classification task model and then combine specific object detection algorithms for identification and localization.

2.2 Human-Object Interaction Detection

Recently, Human-Object Interaction (HOI) detection has attracted increasing attention to the computer vision community. Different from object detection, HOI detection predicts a triplet <Subject, Object, Verb>, including a pair of bounding boxes (human and object), object category label (e.g., "bench", "cat"), and interaction category label (e.g., "catch", "feed"). Therefore, interaction detection not only focuses on the detected objects but also needs to recognize the interaction between humans and objects. HOI detection allows us to understand the scenes in images or videos at a fine-grained level.

HOI detection methods aim to associate interactive pairs of humans and objects and understand their interactions, which can be mainly categorized into two paradigms: bottom-up and top-down.

Bottom-up methods, first detect humans and objects and then associate the humans and objects through a classifier [13, 38] or a graph [20, 26, 37, 39] model. The advantage of these methods is that they can fully utilize the detected individual information, thus, providing more contextual information during the association stage.

On the other hand, top-down methods, usually design an intermediate representation to denote the interaction, such as interaction points [21, 34] or queries [3–5, 17, 33], and then match the corresponding human and object through pre-defined associative rules. The strength of these methods lies in their high computational efficiency, enabling real-time HOI detection. In this paper, we use two HOI detection methods, an interaction point-based method, PPDM [21], and a query-based method, SOV-STG [3]. The PPDM model can detect HOI instances in real-time which is used to extract images from our collected videos. SOV-STG is a powerful HOI detection method with high accuracy, which is used to automatically annotate our image dataset and construct the eating analysis system.

2.3 Hand-Object Interaction Detection

Hands are the principal tools that humans use to interact with the environment. In computer vision, hand-object interaction detection plays a crucial role as it extracts and understands hand movement information. The purpose of the hand-object interaction task is to precisely detect and locate the hand, and when an interaction with an object occurs, it also aims to identify and locate the interacting object.

EPIC-KITCHENS [6], a large-scale dataset aims to enhance first-person vision's hand-object interaction. It comprises 55 hours of non-scripted daily activities videos, in diverse kitchen environments. 100DOH [31] dataset totally contains 131 days of videos and 100K annotated hand-contact video frames. A hand-object pair annotation includes hand location, side, contact state, and a box

[1]For the construction of our HowToEat dataset, due to the copyright of videos we used, we will not provide the video data, but the "vid" of videos, annotations with frame numbers, and the data processing code.

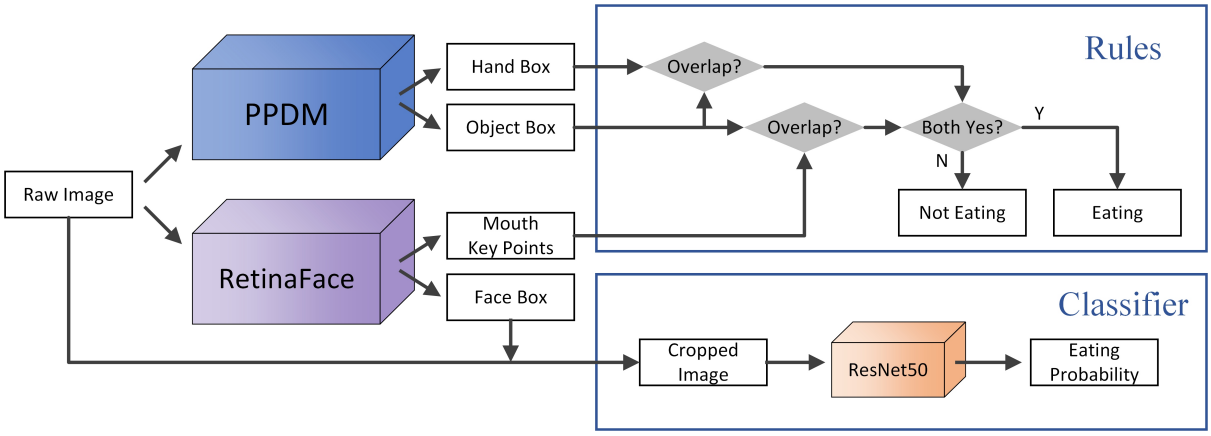


Figure 2: This figure illustrates the pipeline of automatic image extraction and labeling. The top half is a rule-based face-eating action detection pipeline, and the bottom half is face-eating action detection based on classifiers.

75866 Train 8547 Test (instance num)	Self-Contact (1521)	Another Person (163)	Portable Object (11521)	Stationary Object (830)	mAP (14035)
AP	43.56	19.64	65.80	19.86	37.22
max Recall	71.33	61.35	80.61	74.82	72.03

Table 1: The HOI detection evaluation result on 100DOH.

	Image	Hand Side		Portable Object	No Contact
		Left Hand	Right Hand		
Train	82,724	76,288	78,763	99,962	55,089
Test	9,208	8,309	8,597	11,038	5,868

Table 2: Annotation distribution of 100DOH converted to train SOV-STG

around the object in contact. Furthermore, A hand-object detection baseline model based on Faster-RCNN [29] is built.

3 CONSTRUCTING HOWTOEAT

In this section, we will provide a detailed explanation of our dataset construction process. To ensure the diversity of data, e.g. the background, the person, the food, and the tableware, we first collect videos from the internet. As shown in Figure 1(a), 12 phrases (e.g. eating hamburgers, drinking beer, eating noodles, and eating with a spoon) are used as search queries. For each phrase, we use five languages (English, Japanese, French, Chinese, German) to search, and a total of 13,122 videos are collected. The total duration of all videos is 66 days, and more than half of the videos are 0~5 minutes (44%) and 0~10 minutes (26%) as shown in Figure 1(b). To extract the necessary keyframes from video data and perform annotation, we take the following three major steps:

- First, we combine an HOI detection model PPDM trained on 100DOH with a face detection model to detect eating action. The eating action face is classified by sample rules based on the spatial information of the mouth and the object. The frames containing eating actions are extracted, and face boxes are annotated in this step.
- Second, we crop the face bounding box and manually annotate a HowToEat face dataset, and train a model for eating action recognition. Then, we use this model to label all face bounding boxes with eating action labels.
- Third, we re-categorize the class labels of the 100DOH dataset and train a powerful HOI detection model, SOV-STG to automatically provide high-quality labels and bounding boxes for hand-object pairs.

3.1 Automatic Image Extraction and Face Box Annotation

3.1.1 Hand Interaction Detector. 100DOH randomly chooses frames in videos and filters out images containing no hands. Then, the hand bounding box $\{x_h, y_h, w_h, h_h\}$, object bounding box $\{x_o, y_o, w_o, h_o\}$, the hand side $\{left, right\}$, and the contact state {no contact, self-contact, in contact with another person, in contact with a portable object, in contact with a stationary object} are manually labeled. Our purpose is to extract images containing {Hand, Object, Action}

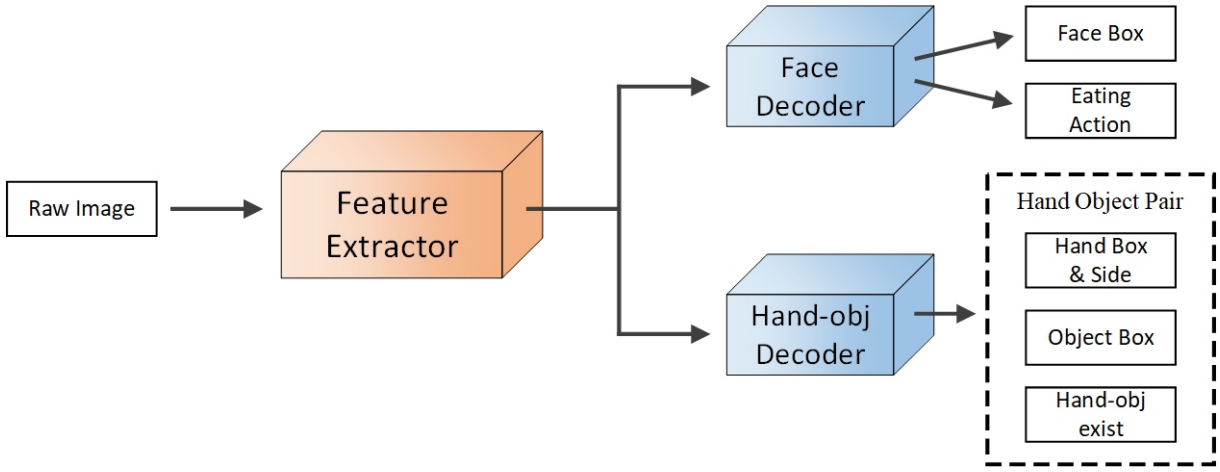


Figure 3: Multi-branch architecture of SOV-STG-H2E.

instances. Based on 100DOH, we can train an HOI detector to automatically extract images from our raw data. We simply convert 100DOH to an HOI detection dataset by removing all of the "No Contact" hand instances and the category of the hand side. PPDM is modified for training on 100DOH as follows: (1) Using the human instance prediction branch to predict hands. (2) Replacing the HOI Action categories with the hand-object action categories of the 100DOH. (3) Removing the labeling of object categories. The same as the HOI detection task, we conduct the average precision as the evaluation result. For an HOI instance, a positive true prediction must predict the right action category, and have an interaction of union (IoU) between the predicted hand bounding box B_h and the ground truth hand bounding box B'_h more than 0.5, and also the corresponding object box IoU $(B_h, B'_h) > 0.5$. The result on the test set is shown in Table 1, the AP of the "Portable Object" category, where the number of instances accounted for 82% of the test set, reach 65.8% and the maximum recall rate is 80.6%. From the results, PPDM achieves an acceptable performance for the hand-object detection of our raw data.

3.1.2 Image Extraction. In the eating scenario, interactions mostly consist of hands and portable objects, thus, we use the PPDM model trained on 100DOH to detect and extract images from the raw videos. Furthermore, we leverage an off-the-shelf face detector, RetinaFace [7] to detect face key points and bounding boxes in the image. Specifically, we implement the RetinaFace model with ResNet-50 [14] backbone to balance the trade-off between accuracy and efficiency. With face bounding boxes and hand-object instances, the eating action can be inferred by checking the overlap between the object and the mouth key points of the face and between the hand and the object, which is shown in the top half of Figure 2. Then, we use PPDM and an eating action detector to extract the frame from the raw videos. We detect one frame per second and extract the frame containing eating actions according to our rules. In this step, a total of 99k images are extracted, and the face boxes are automatically labeled by RetinaFace.

3.2 Eating Action Annotation

Detecting eating action by rules can not handle some difficult circumstances, cause the bounding box is not sensitive to the rotation of the object. For example, the mouth overlaps with the background in the bounding box of the object. Therefore, with our dataset, we manually label 6,280 (train: 5033, test: 1247) face images cropped by RetinaFace in our dataset and train a classifier based on ResNet-50 which is shown in the bottom half of Figure 2. The distribution of face instances in different scenarios is shown in Figure 5. From the manually annotated dataset, in the first step, misclassification by rules commonly exists.

We build the eating action classifier upon ResNet-50. Specifically, we replace the classification header of ResNet-50 pre-trained on ImageNet-1K, then train the whole network by SGD [30] optimizer with a learning rate of 1e-3 on 4 GPUs (batch size 32). After training, the classifier achieves 86.4% accuracy on the test set and can be used to recognize the eating action in the wild. Then, we replace the rule-based eating action recognition approach with the learned ResNet-50 classifier and automatically annotate all the face boxes of the extracted 99k images (in Sec 3.1.2).

3.3 Hand-Object Annotation

Although the PPDM we used for automatic image extraction performs well with high efficiency, it is not the best choice to generate high-quantity annotations for our dataset. Therefore, we leverage a more accurate HOI detection method to generate hand-object annotations. Specifically, we implement SOV-STG-Hand for the hand-object interaction detection task, we select the SOV-STG, with two sizes: SOV-STG-S and SOV-STG-swin-L which performs well in the human-object interaction task. In detail, we implement a two-decoder architecture of SOV-STG-Hand, which is by removing the verb recognition part of the SOV framework and replacing the subject decoder as our hand decoder. We remove the interaction recognition head after the verb decoder and add an interaction existing head after the hand decoder. To better detect the portable objects and ignore other kinds of objects which are rare in eating scenarios, we remove the object boxes of categories other than

Model	Left Hand		Right Hand		Hand-Object
	No Contact	Portable Object	No Contact	Portable Object	mAP
SOV-STG-Hand-S	64.61	73.04	56.99	72.76	66.85
SOV-STG-Hand-Swin-L	70.16	80.50	65.35	77.05	73.26

Table 3: The results after training SOV-STG-Hand on 100DOH with redefined categories.

Model	Left Hand		Right Hand		Hand-object	Face		Face
	No Contact	Portable Object	No Contact	Portable Object	mAP	Not Eating	Eating	mAP
SOV-STG-H2E-S	61.91	87.79	47.98	88.56	71.56	57.43	73.89	65.66

Table 4: The results of the baseline model on HowToEat.

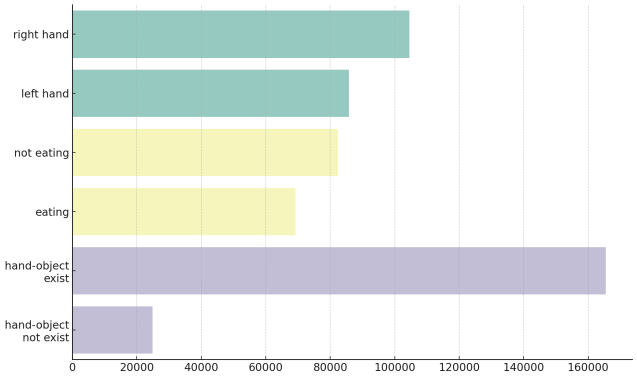


Figure 4: Statistics about the hand side, eating action, and hand-object exist distribution in the HowToEat dataset.

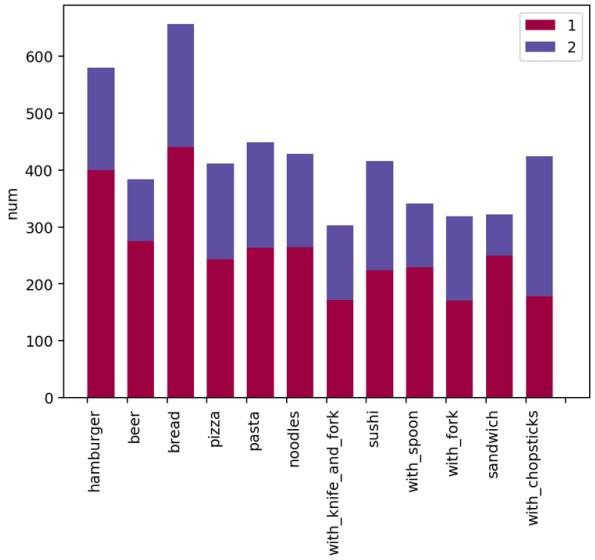


Figure 5: The distribution of face categories in the HowToEat face dataset. The labels [1] and [2] represent [eating] and [not eating], respectively.

"Portale object" and replace the category of hand with "No Contact". The annotation distribution of the converted 100 DOH is shown in Table 2 and Table 3 shows the results of the model.

After the re-annotation of the hand-object interaction is completed, we proceed with the filtering step. we remove images in which the largest face bounding box in the image has fewer than 400 pixels. At the same time, to ensure the accuracy of the annotations, we limited the maximum number of faces in an image to no more than 5. Additionally, for images without faces, we also removed them from the dataset, because if there is no face in the image, it cannot accurately reflect the eating scenario. After these series of filtering operations, we further delete those bounding boxes for faces smaller than 300 pixels to ensure the clarity of the face in the images.

4 IMAGE DATASET AND BASELINE MODEL

As shown in Figure 4, hands are annotated according to whether they are left or right and whether they interact with an object, while face categories are divided based on whether eating is taking place.

4.1 Annotation

For every hand in each image, we obtained the following annotations: (1) a bounding box around the hand; (2) side: left or right; (3) the hand state ({No Contact, Portable Object}), and (4) for the hand-object interaction that exists, a bounding box around the target will also be annotated.

For every face in each image, we obtained the following annotations: (1) a bounding box around the face; (2) the face state ({eating, not eating}).

4.2 Split Dataset

Following the process in Section 3, we have constructed a dietary behavior dataset: HowToEat. The dataset contains 95,190 images, which include 190,333 instances of hands and 151,620 instances of faces. We split the dataset into training and testing sets at a ratio of 4:1, with the training set consisting of 76,905 images and the testing set containing 18,285 images.

Figure 6: Qualitative examples of detections from our baseline model.

4.3 Baseline Model

Although the hand interaction detection model, SOV-STG-Hand in Sec 3.3 and the eating action detector in Sec 3.2 achieves notable results in two tasks, respectively. However, for our purpose of building an eating analysis system, adopting two independent models is not computationally efficient. Moreover, in the future, if we want to add other tasks or annotations to our dataset, the analysis system will become heavier than training a single model. To this end, we propose an encoder-decoder architecture based on SOV-STG for our HowToEat dataset, which we named SOV-STG-**H**ow2**E**at (SOV-STG-H2E). As shown in Figure 3, SOV-STG-H2E has two branches. In the hand-object decoder branch, which is the same as SOV-STG-Hand. Specifically, if the hand interacts with a certain object, the model will output a hand-object paired bounding box and hand side. Otherwise, the model will only output the hand side and bounding box. For the face decoder branch, the model outputs the bounding box of the face and whether there is an eating action. In addition, in the future, if a new task is added to our dataset, we only need to add a new decoder to incorporate it into our analysis system.

5 EXPERIMENTS

5.1 Evaluation Setup and Metric

Following the standard evaluation metric for object detection and Human-object Interaction detection, we evaluate the HowToEat dataset using mean average precision (mAP). We reported the mean Average Precision on two different tasks: (1) the face interaction detection task, and (2) the hand-object interaction detection task.

In the evaluation of hand-object interactions, we adopted a method similar to that used in object detection tasks, that is using

the intersection over union (IoU) with the ground truth greater than 0.5 to judge whether the detection result is a true positive instance. When the ground truth label for the hand is "interaction exists", the IoU of the predicted hand and object bounding boxes with the ground truth should be greater than 0.5; when the ground truth label for the hand is "no interaction exists", only the IoU of the predicted hand bounding box with the ground truth should be greater than 0.5, while the predicted object box is disregarded and not included in the accuracy calculation.

5.2 Implementation Details

We adopt the smallest size of SOV-STG as our baseline, which we named SOV-STG-H2E-S, and we apply the same hyperparameter settings as those used in SOV-STG, except that the number of queries is changed to 16. Specifically, the feature extractor consists of a ResNet50 backbone, a 6-layer deformable transformer encoder [40], and 3-layer decoders. We train the model with the AdamW [25] optimizer with a learning rate of 2e-4 (except for the backbone, which is 1e-5) and a weight decay of 1e-4. The batch size is set to 32 (4 images per GPU), and the training epochs are 30 (learning rate drops at the 20th epoch) All of the experiments are conducted on 8 NVIDIA A6000 GPUs.

5.3 Qualitative Results

In our model testing procedure, we directly applied the detection to the test set, setting the confidence threshold for hands and faces at 0.5. As shown in Figure 6 and Table 4, Our model can effectively detect hand-object pair bounding boxes and categories, while relatively reliably recognizing and classifying facial regions, thereby establishing a robust baseline for our subsequent research. Due to limitations in the performance of the automatic facial annotation model, misclassification may occur in the final results, as shown in Figure 6(b). However, our model often has the ability to correct these errors, demonstrating its potential to improve classification accuracy.

6 CONCLUSION AND FUTURE WORK

In this paper, we introduce the HowToEat dataset, a unique resource designed for detecting eating actions. This comprehensive dataset considers both hand-object interactions and eating action to provide nuanced insights into eating scenarios. In addition, we adapt HOI interaction methods to Hand-object interaction detection and achieve acceptable results. Furthermore, we construct an eating analysis system as a baseline for our multi-task dataset and provide a benchmark for future research.

In the future, we plan to improve our automatic annotations with a model-in-loop approach like SAM [18] and reduce the inaccurate ground-truth. Furthermore, we plan to enrich our dataset by adding annotations for food items, utensils, calorie labels, etc. We also plan to refine the detection process by pairing hand-object interaction instances with eating action instances. Through these continued enhancements, we aim to advance dietary behavior research and contribute to broader applications in computer vision.

Acknowledgment: This work was supported by JSPS KAKENHI Grant Numbers, 21H05812, 22H00540, 22H00548, and 22K19808.

REFERENCES

[1] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. 2020. End-to-end object detection with transformers. In *ECCV*.
[2] Yu-Wei Chao, Yunfan Liu, Xieyang Liu, Huayi Zeng, and Jia Deng. 2018. Learning to Detect Human-Object Interactions. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*.
[3] Junwen Chen, Yingcheng Wang, and Keiji Yanai. 2023. Focusing on what to decode and what to train: Efficient Training with HOI Split Decoders and Specific Target Guided DeNoising. *arXiv preprint arXiv:2307.02291* (2023).
[4] Junwen Chen and Keiji Yanai. 2022. Parallel Queries for Human-Object Interaction Detection. In *Proceedings of the 4th ACM International Conference on Multimedia in Asia*.
[5] Mingfei Chen, Yue Liao, Si Liu, Zhiyuan Chen, Fei Wang, and Chen Qian. 2021. Reformulating HOI detection as adaptive set prediction. In *CVPR*.
[6] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. 2018. Scaling Egocentric Vision: The EPIC-KITCHENS Dataset. In *European Conference on Computer Vision (ECCV)*.
[7] Jiankang Deng, Jia Guo, Evangelos Ververas, Irene Kotsia, and Stefanos Zafeiriou. 2020. RetinaFace: Single-Shot Multi-Level Face Localisation in the Wild. In *CVPR*.
[8] Mark Everingham, SM Ali Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. 2015. The pascal visual object classes challenge: A retrospective. *International journal of computer vision* 111 (2015), 98–136.
[9] Ross Girshick. 2015. Fast R-CNN. In *ICCV*.
[10] Georgia Gkioxari, Ross Girshick, Piotr Dollár, and Kaiming He. 2018. Detecting and recognizing human-object interactions. In *CVPR*.
[11] Francisco Gomez-Donoso, Sergio Orts-Escolano, and Miguel Cazorla. 2017. Large-scale Multiview 3D Hand Pose Dataset. arXiv:1707.03742 [cs.HC]
[12] Saurabh Gupta and Jitendra Malik. 2015. Visual semantic role labeling. *arXiv preprint arXiv:1505.04474* (2015).
[13] Tanmay Gupta, Alexander Schwing, and Derek Hoiem. 2019. No-frills human-object interaction detection: Factorization, layout encodings, and training techniques. In *ICCV*.
[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *CVPR*.
[15] Vidit Jain and Erik Learned-Miller. 2010. *FDDB: A Benchmark for Face Detection in Unconstrained Settings*. Technical Report UM-CS-2010-009. University of Massachusetts, Amherst.
[16] Alexander Kapitanov, Andrey Makhlyarchuk, and Karina Kvanchiani. 2022. HaGRID - HAnd Gesture Recognition Image Dataset. *arXiv preprint arXiv:2206.08219* (2022).
[17] Bumsoo Kim, Junhyun Lee, Jaewoo Kang, Eun-Sol Kim, and Hyunwoo J Kim. 2021. HOTR: End-to-end human-object interaction detection with transformers. In *CVPR*.
[18] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. 2023. Segment Anything. *arXiv:2304.02643* (2023).
[19] Feng Li, Hao Zhang, Shilong Liu, Jian Guo, Lionel M Ni, and Lei Zhang. 2022. Dn-detr: Accelerate detr training by introducing query denoising. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 13619–13627.
[20] Yong-Lu Li, Siyuan Zhou, Xijie Huang, Liang Xu, Ze Ma, Hao-Shu Fang, Yanfeng Wang, and Cewu Lu. 2019. Transferable interactiveness knowledge for human-object interaction detection. In *CVPR*.
[21] Yue Liao, Si Liu, Fei Wang, Yanjie Chen, Chen Qian, and Jiashi Feng. 2020. PPDM: Parallel point detection and matching for real-time human-object interaction detection. In *CVPR*.
[22] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. 2017. Feature pyramid networks for object detection. In *CVPR*.
[23] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft COCO: Common objects in context. In *ECCV*.
[24] Shilong Liu, Feng Li, Hao Zhang, Xiao Yang, Xianbiao Qi, Hang Su, Jun Zhu, and Lei Zhang. 2022. Dab-detr: Dynamic anchor boxes are better queries for detr. *arXiv preprint arXiv:2201.12329*.
[25] Ilya Loshchilov and Frank Hutter. 2018. Decoupled Weight Decay Regularization. In *ICLR*.
[26] Siyuan Qi, Wenguan Wang, Baoxiong Jia, Jianbing Shen, and Song-Chun Zhu. 2018. Learning human-object interactions by graph parsing neural networks. In *ECCV*.
[27] Joseph Redmon and Ali Farhadi. 2017. YOLO9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 7263–7271.
[28] Joseph Redmon and Ali Farhadi. 2018. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767* (2018).
[29] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2016. Faster R-CNN: Towards real-time object detection with region proposal networks. In *IEEE TPAMI*.

[30] Herbert Robbins and Sutton Monro. 1951. A stochastic approximation method. *The annals of mathematical statistics* (1951), 400–407.
[31] Dandan Shan, Jiaqi Geng, Michelle Shu, and David F. Fouhey. 2020. Understanding Human Hands in Contact at Internet Scale. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*.
[32] Shuai Shao, Zeming Li, Tianyuan Zhang, Chao Peng, Gang Yu, Xiangyu Zhang, Jing Li, and Jian Sun. 2019. Objects365: A large-scale, high-quality dataset for object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*. 8430–8439.
[33] Masato Tamura, Hiroki Ohashi, and Tomoaki Yoshinaga. 2021. QPIC: Query-based pairwise human-object interaction detection with image-wide contextual information. In *CVPR*.
[34] Tiancai Wang, Tong Yang, Martin Danelljan, Fahad Shahbaz Khan, Xiangyu Zhang, and Jian Sun. 2020. Learning human-object interaction detection using interaction points. In *CVPR*.
[35] Shuo Yang, Ping Luo, Chen Change Loy, and Xiaoou Tang. 2016. WIDER FACE: A Face Detection Benchmark. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
[36] Aixi Zhang, Yue Liao, Si Liu, Miao Lu, Yongliang Wang, Chen Gao, and Xiaobo Li. 2021. Mining the Benefits of Two-stage and One-stage HOI Detection. In *NeurIPS*.
[37] Frederic Z Zhang, Dylan Campbell, and Stephen Gould. 2021. Spatially conditioned graphs for detecting human-object interactions. In *ICCV*.
[38] Frederic Z. Zhang, Dylan Campbell, and Stephen Gould. 2022. Efficient Two-Stage Detection of Human-Object Interactions with a Novel Unary-Pairwise Transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 20104–20112.
[39] Penghao Zhou and Mingmin Chi. 2019. Relation parsing neural network for human-object interaction detection. In *ICCV*.
[40] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. 2020. Deformable DETR: Deformable transformers for end-to-end object detection. In *ICLR*.