***Journal of Artificial Intelligence and Technology*, 2021, 1, 138-143**
https://doi.org/10.37965/jait.2021.0017

ISTP
RESEARCH ARTICLE

Deep Neural Network-based Speaker-Aware Information Logging for Augmentative and Alternative Communication

Gang Hu,[1] Szu-Han Kay Chen,[2] and Neal Mazur[1]
[1]Department of Computer Information, State University of New York Buffalo State, Buffalo, NY 14222 USA
[2]Department of Communication Disorders and Sciences, State University of New York, Fredonia, NY 14063 USA

(Received 08 April 2021; Revised 15 April 2021; Accepted 16 April 2021; Published online 19 April 2021)

Abstract: People with complex communication needs can use a high-technology augmentative and alternative communication device to communicate with others. Currently, researchers and clinicians often use data logging from high-tech augmentative and alternative communication devices to analyze augmentative and alternative communication user performance. However, existing automated data logging systems cannot differentiate the authorship of the data log when more than one user accesses the device. This issue reduces the validity of the data logs and increases the difficulties of performance analysis. Therefore, this paper presents a solution using a deep neural network-based visual analysis approach to process videos to detect different augmentative and alternative communication users in practice sessions. This approach has significant potential to improve the validity of data logs and ultimately to enhance augmentative and alternative communication outcome measures.

Key words: augmentative and alternative communication (AAC); outcome measures; visual logs; hand tracking; deep learning

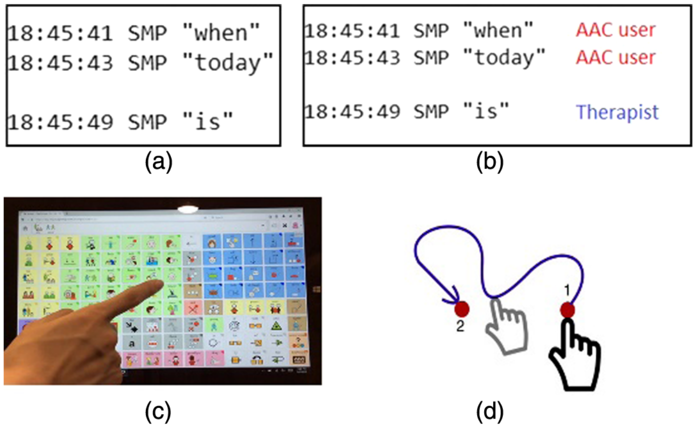
I. INTRODUCTION

An estimated 3.7 million people in the United States have severe speech and language impairments due to various medical issues such as autism, cerebral palsy, aphasia, and amyotrophic lateral sclerosis. Augmentative and alternative communication (AAC) is used to supplement or replace speech for them in the production or comprehension of spoken language [1]. One type of AAC uses high-tech AAC devices to communicate with others. AAC users without physical impairment use their hands/fingers to select words/icons on a device's touchscreen to produce a computer voice to express what they want to "say." To improve AAC users' communication performance, evidentiary interventions are necessary to help users utilize the device functionalities effectively [1]. AAC intervention is expensive and time consuming [2], [3]. A successful intervention relies on accurate log data collecting and active outcome monitoring [4], [5]. However, various types of AAC devices on the market have their own data collecting systems and are not compatible with each other. Thus, the current AAC field has no comparable means to gather, share, access, or analyze a large pool of logged data to study the statistical correlation between AAC users' performance and technical variables during the intervention. In the AAC clinical and research communities, practitioners need a set of easy-use, objective measurement, and accurate logging toolkits that can work with all AAC devices for both clinical and research purposes.

Previous AAC studies have demonstrated the success of using automatic data logging (ADL) to monitor AAC users' language and communication performance in research contexts [4], [6], [7]. ADL has two basic variables: the timestamp and the text output. Some extra variables may be available associated with different access methods: keyboard typing, eye gazing typing, and so on [8]. According to the study done by Chen *et al.* [8], by utilizing different ADL formats with various variables, AAC practitioners could compare AAC users' performance between different data logs. However, the existing ADL can only capture the AAC usage data without distinguishing its producers (users). In clinical settings, AAC intervention sessions usually involve multiple participants (an AAC user, a clinician, family members, and other communication partners). Most of the time, only one AAC device is used. While an AAC user practices on the AAC device, other participants use the same device to provide modeling to help the user to learn words and functions on the high-tech AAC system [9]. Therefore, when reading the ADL files after sessions, it is difficult to differentiate who generated particular words. Fig. 1(a) shows the ADL generated by CoughDrop [10]. By reading the ADL data, it is impossible to know if the three lines were generated by the AAC user only or other participants. This limitation not only reduces the validity of data logs and the accuracy of performance analysis including semantic analysis, syntactical analysis, and usage efficiency [11], but also further impedes the efficiency of AAC services. The current solution for this limitation is to have clinicians manually clean the ADLs (filtering or labeling) by using predetermined guidelines [7] or comparing the ADLs with the recorded videos from intervention sessions with human vision. These methods are not only time consuming and labor intensive but also error prone. In addition, it reduces the time that clinicians can work with AAC users.

Although hand gesture analysis is common in other fields, so far no related computing and analysis study has been applied in the AAC field. Thus, we are working on a visual-based logging system to make the data logging user (speaker)-aware-able by analyzing AAC usage videos. Ultimately, the AAC data logs will have an additional attribute indicating the producer of each event (see the example in Fig. 1(b)). The essential part of this novel speaker-aware information logging (SAIL) system is to recognize all participants from the videos, where the hands are in an egocentric

Corresponding author: Gang Hu (e-mail: hug@buffalostate.edu).



```
18:45:41 SMP "when"
18:45:43 SMP "today"

18:45:49 SMP "is"
```

(a)

```
18:45:41 SMP "when"     AAC user
18:45:43 SMP "today"    AAC user

18:45:49 SMP "is"       Therapist
```

(b)

(c)

(d)

Fig. 1. Example of AAC use with its automatic data log (ADL) and the proposed SAIL.

viewpoint (see Fig. 1(c) and (d)). The contributions of this work are twofold:

- Collect egocentric hand gesture images with labeled tags, which is the first labeled image dataset in the AAC field.
- Present a proof-of-concept application that utilizes a deep neural network to classify different AAC participants using the same device.

In the rest of this paper, Section II introduces some related work; Section III presents the framework in detail; Section IV provides the evaluation results; and Section V draws conclusions and describes future work.

II. RELATED WORK

Many works employ deep convolutional neural networks (CNNs) to study hand-object interactions in an end-to-end framework [12], [13]. In contrast to these approaches, our solution involves the use of first-person activity recognition, where human hands are naturally the only information we can get as far as the movements are concerned. Therefore, in the context of representing activities, object manipulation and hand movements are the main source of visual information.

Thus, related works in this category describe activities using an object-centric approach that derives from the existence of specific objects in the scene [13]–[16]. Moreover, scene understanding is also used in [17]. In [18], a multitask clustering framework tailored to first-person view activity recognition is presented. Another more recent approach uses deep CNN architectures [19] to learn deep appearance and motion clues. Deep CNNs are also used to learn hand segmentations in order to understand the multiple user activities and interaction with each other [13], [20].

Some works focus on multimodal analysis of egocentric cameras and information from other wearable sensor equipment with the deployment of early or late fusion schemes [21]–[23]. More recently, González-Díaz *et al.* [24] proposed CNN-based solution to control grabbing actions in an egocentric view, where an eye tracking and image processing tools are used to measure eye movements or gaze direction for target manipulation. To handle the noisy and unstable gaze signals caused by unconstrained human posture and cluttered working environment, CNN framework is utilized to process the gaze fixation signal. It can determine the target location and classify the object in an end-to-end fashion. Motivated by the need of recognizing activities of patients with critical disease in nursing homes or hospitals, Giannakeris *et al.* [25] analyze egocentric videos provided by patient's wearable cameras. This setting is similar to ours. The detected objects incorporating the motion patterns into low-level microaction descriptors, Bag-of-Micro-Actions, and then, Gaussian mixture models clustering and Fisher vector encoding are used to recognize the activities.

III. METHODOLOGY

Visual information of AAC practices is captured from an egocentric viewpoint [26], where the camera faces the hand and the device screen with the same viewpoint as the AAC user's eyes (see Fig. 1(c)). The recorded videos contain the hand/finger actions and the screen contents without the user's face, which protects the individual's privacy and confidentiality in clinical data. To distinguish the roles of users, wearing a colored ring on the working finger is required for some users, and the color is role dependent. For example, a green ring is for family members and the red is for therapists. AAC users can be with or without a ring. The color-role association can be decided according to user's preferences. Based on this setting, the producers of the log data can be identified when an AAC practice video is processed by the SAIL system, where a customized deep neural network single shot detection [27] is used for this task.

A. SINGLE SHOT DETECTOR (SSD) NETWORK STRUCTURE

The single-shot detector (SSD)-based network structure has two parts: the base network and the auxiliary network illustrated in Fig. 2. The base network (blue blocks in Fig. 2) uses the well-established VGG-16 [28] for feature extraction, and the auxiliary section (green blocks) has several convolutional layers to predict the bounding boxes and corresponding confidence scores for detecting targets (hands). The size of the auxiliary layers gradually decreases so that different sizes of hands can be detected. This model predicts multiple boundary boxes from a single image, and then, non-maximal suppression is applied to obtain the final predictions with high confidence scores in the image. This SSD-based detection model could handle the challenges in our task, such as varied/poor lighting, diverse background, diverse view angles, and unexpected occlusions.

B. TRAINING

We train the network to determine the target box via bounding box regression, and object classification of the target box.

During training, we determine the correspondence between default boxes to a ground-truth detection. For each ground-truth box, we find its default box with the best overlap, and also match the default boxes to any ground truth with overlap higher than a threshold (e.g., 0.5). This allows the network to predict high scores for multiple overlapping default boxes rather than picking only the one with maximum overlap. The overall loss function L is a weighted sum of localization loss (*loc*) and the confidence loss (*conf*):

$$L(x,c,l,g) = \frac{1}{N}(L_{conf}(x,c) + \alpha L_{loc}(x,l,g)), \tag{1}$$

where N is the number of matched boxes. The weight term α helps us in balancing the contribution of the location loss. As usual in deep

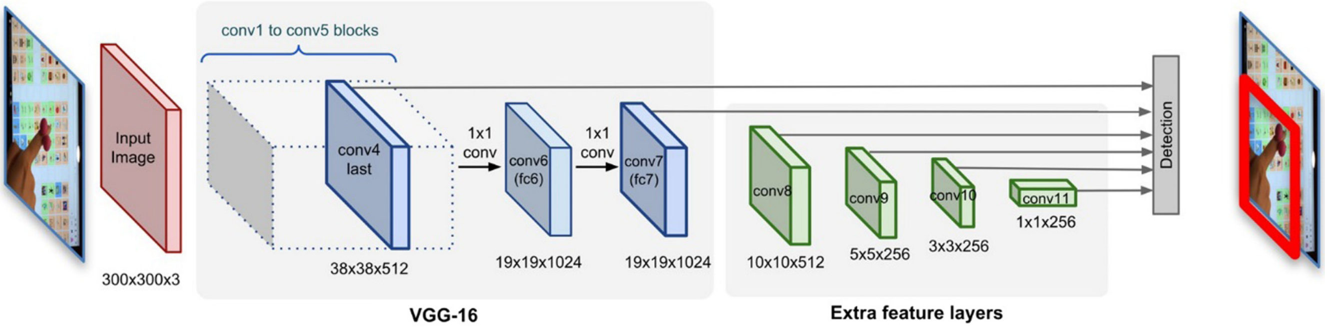


Fig. 2. Network structure of SSD.

learning, the goal is to find the parameter values that most optimally reduce the loss function, thereby bringing the predictions closer to the ground truth. In this work, α is set to 1 by cross validation.

Confidence loss (*conf*) measures how confident the network believes the true object is in the computed bounding box. Categorical cross-entropy is used to compute this loss:

$$L_{conf}(x,c) = -\sum_{i\in pos}^{N} x_{ij}^{p}\log(e(c_i^p)) - \sum_{i\in neg}\log(e(c_i^0)), \qquad (2)$$

where $x_{ij}^{p} = \{1,0\}$ is an indicator for matching the i-th default box to the j-th ground-truth box of p class; and $e(\cdot)$ is the prediction logits of class p.

Location loss (*loc*) measures how faraway the predicted bounding boxes are from the ground-truth ones from the training set:

$$L_{loc}(x,l,g) = \sum_{i\in pos}^{N}\sum_{m\in\{c_x,c_y,w,h\}} x_{ij}\text{smooth}_{L1}(l_i^m - g_j^m), \qquad (3)$$

where smooth$_{L1}$ loss between the predicted box (l) and the ground-truth box (g) parameters is used. Here, m includes the center (c_x, c_y), width (w), and height (h) of default and the ground-truth boxes. By minimizing this loss, the network regresses the target to the ground-truth box.

C. AAC EGOHAND DATASET

For this proof-of-concept application, we collected four videos of simulated AAC practice sessions from four different users interacting with three AAC software: CoughDrop [10], Sono Flex [29], and Proloquo2go [30]. These were downloaded from the Apple App Store and have grid-design displays, color icons, and synthesized speech output (see Fig. 3). Depending on the AAC activities in these simulation sessions, we extracted roughly 1,200 to 2,500 hand images per video, which are then manually annotated with ground-truth (GT) labels at the pixel level. There are five hand categories defined (see Fig. 3): hands with blue, red, yellow, and green rings, and a bare hand without a ring. We have collected 7,124 labeled images for our AAC egohand dataset, which are split into training and testing sets with 5,055 and 2,069 images.

Since the size of our AAC egohand dataset is relatively small, we followed the fine-tuned transfer learning strategy to train our model. Specifically, an SSD model pretrained on the common objects in context (COCO) dataset [31] is publicly available and utilized as the starting point of our training process. Our specific hand detection network is further trained on our AAC egohand dataset. COCO, a large-scale object detection dataset, has 200k labeled images with 1.5 million object instances under 80 object categories. By utilizing the generic image features extracted from this large dataset, the training process for our task is efficient and effective.

D. DATA AUGMENTATION

Like in many other deep learning applications, data augmentation has been crucial to teach the network to become more robust to various object sizes in the input. To this end, additional training examples are generated. They are from the patches of the original image at different overlap ratios (e.g., 0.1, 0.3, 0.5, etc.) and random patches as well. Moreover, each image is also randomly horizontally flipped with a probability of 0.5, thereby making sure potential hand objects appear on left and right with similar likelihood.

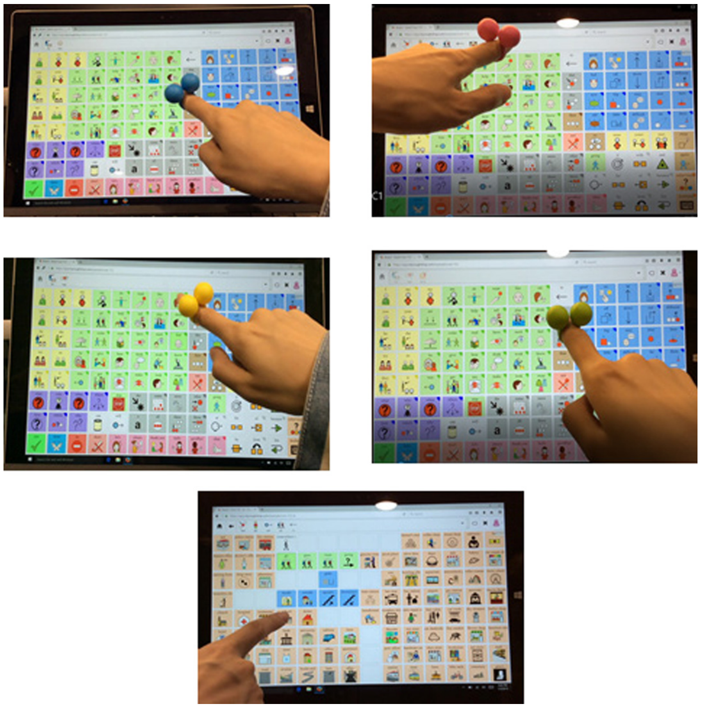


Fig. 3. AAC egohand dataset with five classes.

IV. EVALUATION

A. EXPERIMENTAL SETTING AND NETWORK TRAINING

The training and testing of this proof-of-concept system were conducted on a Dell server with two NVIDIA GeForce RTX 2080 GPUs, Intel(R) Core(TM) i7-9800 3.80 GHz CPU and 64 GB RAM. The neural network is built on the deep learning framework TensorFlow 1.15 on Ubuntu 18.04 operating system. The program code is written in Python 3.7.6. The input image size is 1240×720 pixels, which is then adjusted to 300×300 during the training process. Training is a process of network updating, and is set to 200,000 iterations with batch size = 5. The initial learning rate is set to 4e-3. After 60k, 120k, and 180k iterations of training, the rate was reduced to 3.8e-3, 3.6e-3, and 3.4e-3, respectively. Table I lists the details of the training settings. The goal of training is to minimize the difference between the predictions and the ground truth. Such differences are represented by the loss values. For the object detection tasks, there are two types of losses: classification loss and localization loss. Localization loss gives the distances between the predicted boxes to the target object (hand), while the classification loss reflects the recognition errors on the predicted target boxes. The total loss combines both with corresponding weights. Fig. 4 shows the total loss curve during the training process, which has a rapid downward trend at the beginning, and then slowly decreases until convergence. The training loss is eventually stable and reaches the minimum value 1.63 at 200,000 iterations.

TABLE I. Parameters in SSD Model

Parameter name	Value
Original image size	1240×720
Adjusted image size	300×300
Initial learning rate	4×10^{-3}
Learning rate decay steps	60,000
Learning rate decay factor	0.95
Image number in each iteration	5
Iteration number	200,000

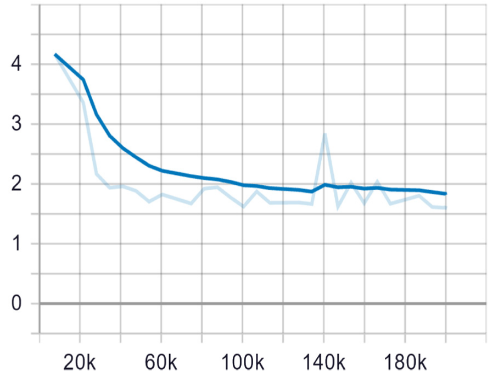


Fig. 4. Training loss curve.

B. NON-MAXIMUM SUPPRESSION

Given the large number of boxes generated during a forward pass of the network at inference time, it is essential to prune most of the bounding box by applying non-maximum suppression. Specifically, the boxes with a low confidence value (e.g., <0.01) and a small overlap ratio (e.g., <0.45) are discarded, and only several top predictions are kept. This ensures only the most likely predictions are retained by the network, while the much noisier ones are removed.

C. PERFORMANCE ASSESSMENT

Several metrics are used to assess the effectiveness of our hand detection model. The object detection accuracy is associated with the bounding box overlap between the prediction and the ground truth. The intersection over union (IoU) is defined to express the overlap:

$$IoU = \frac{P \cap GT}{P \cup GT}, \tag{4}$$

where P and GT represent the bounding boxes of the prediction and ground truth, respectively. A threshold $\theta_{IoU} \geq 0.5$, indicates an object has been detected. Furthermore, both precision and sensitivity are used to compute the ratios of detected true positives against predictions and entire data samples, respectively. The mean average precision (mAP) combines both sensitivity and precision for detected hands:

$$mAP = \frac{\sum_{q=1}^{Q} AP}{Q}, \tag{5}$$

where AP is the average precision and Q is the number of testing cases. Fig. 5 shows the mAPs for different IoU thresholds. When IoU = 1, the performance of our model reaches 85%+. mAP is close to 100% when IoU = 0.5, which is a common threshold treating an object as detected. This result demonstrates that the trained model performs well on the testing dataset.

In addition, the sensitivity score is also present on Fig. 6 which shows a curve of average sensitivity for top 10 detection ranking. The score gradually increased until it reached its highest peak at 91% when the training was close to iteration # 200k.

Table II is the confusion matrix of our AAC participants role detection based on IoU = 0.75. According to the experimental assumption, therapist, both family members, and social worker wear blue, red, yellow, and green rings, respectively, while AAC user (the patient) does not wear anything. This matrix uses different font colors to indicate the roles of AAC participants. From the matrix, the AAC user with bare hand and Therapist (with blue ring) hand can be recognized with almost perfect accuracies. However,

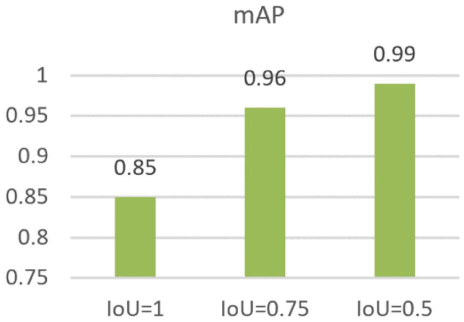


Fig. 5. mAP of different IoU.

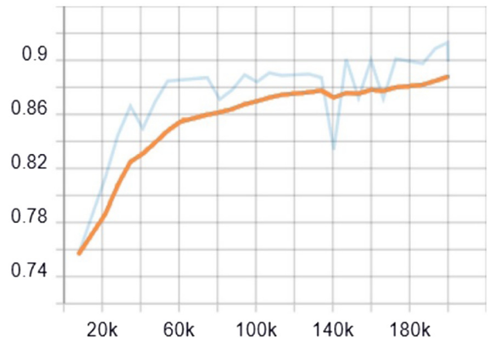


Fig. 6. Average sensitivity for top 10.

TABLE II. Confusion matrix of our hand recognition method

	Therapist	Family member-1	Family member-2	Social worker	AAC user
Therapist	.98	0	.01	.01	0
Family member-1	0	.97	.02	.01	0
Family member-2	0	.01	.92	.05	.02
Social worker	.01	0	.04	.94	.01
AAC user	0	0	.01	0	.99

the family member with the yellow ring is relatively easy to be confused with social worker with the green ring. The major reason for this issue is related to the light condition and the ring colors, that is, the green color under the indoor light was turned to yellowish color, which makes the network hard to distinguish both roles. We will change the ring shape or color in future study.

V. CONCLUSION AND FUTURE WORK

Our proof-of-concept application demonstrates a deep neural network framework for differentiating different "Speakers" using one AAC device. Our system can detect above 90% of different hands successfully across three different AAC software and four different users. The model verifies that the concept does indeed function as envisioned. This approach provides a solution to the clinical issue in the AAC domain. The system is able to improve the validity of automatic data logs, and potentially will make AAC users' performance analysis more accurate. Our next step is to improve the detection accuracy by increasing the diversity of training data, utilizing more salient hand features. We also will use an interoperable data log format [6] to generate our SAIL data logs, then conduct a user study to evaluate their consistency and usability by comparing with traditional language sample analysis.

REFERENCES

[1] D. R. Beukelman and P. Mirenda, *Augmentative and Alternative Communication*. Baltimore: Paul H. Brookes, 1998.

[2] H. P. Parette, Jr., and D. D. Marr, "Assisting children and families who use augmentative and alternative communication (AAC) devices: Best practices for school psychologists," *Psychol. Schools*, vol. 34, no. 4, pp. 337–346, 1997.

[3] E.-H. Wang, L. Zhou, S.-H. K. Chen, K. Hill, and B. Parmanto, "An health platform for supporting clinical data integration into augmentative and alternative communication service delivery: user-centered design and usability evaluation," *JMIR Rehabil. Assist. Technol.*, vol. 5, no. 2, p. e14, 2018.

[4] S. N. Chuileann, "Investigating preference for self-voice on a speech generating device by the child with autism," *Archivos Med.*, vol. 6, no. 2, p. 20, 2015.

[5] R. W. Schlosser, R. Koul, and J. Costello, "Asking well- built questions for evidence-based practice in augmentative and alternative communication," *J. Commun. Disorders*, vol. 40, no. 3, pp. 225–238, 2007.

[6] D. J. Higginbotham and C. R. Engelke, "A primer for doing talk-in-interaction research in augmentative and alternative communication," *Augmentative Altern. Commun.*, vol. 29, no. 1, pp. 3–19, 2013.

[7] T. Kovacs and K. Hill, "Language samples from children who use speech-generating devices: Making sense of small samples and utterance length," *Am. J. Speech Lang. Pathol.*, vol. 26, no. 3, pp. 939–950, 2017.

[8] S.-H. K. Chen, S. Wadhwa, and E. Nyberg, "Design and analysis of interoperable data logs for augmentative communication practice," in *The 21st Int. ACM SIGACCESS Conf. Comput. Accessibility*, Pittsburgh, PA, USA, 2019, pp. 533–535.

[9] S. C. Sennott, J. C. Light, and D. McNaughton, "AAC modeling intervention research review," *Res. Pract. Persons Severe Disabilities*, vol. 41, no. 2, pp. 101–115, 2016.

[10] CoughDrop, Inc., "CoughDrop App," 2019. [Online]. Available: https://https://www.assistiveware.com/products/proloquo2go/, Accessed on: Apr. 2, 2021.

[11] S. W. Blackstone, "Equipment: ADL in AAC devices," *Augmentative Commun. News*, vol. 13, no. 3, p. 5, 2000.

[12] L. Wang, Y. Xiong, Z. Wang, and Y. Qiao, "Towards good practices for very deep two-stream convents," arXiv preprint arXiv: 1507.02159, 2015.

[13] Y. Zhou, B. Ni, R. Hong, X. Yang, and Q. Tian, "Cascaded interactional targeting network for egocentric video analysis," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Las Vegas, NV, USA, 2016, pp. 1904–1913.

[14] A. Fathi, A. Farhadi, and J. M. Rehg, "Understanding ego-centric activities," in *2011 Int. Conf. Comput. Vision*, Barcelona, Spain, 2011, pp. 407–414.

[15] T. McCandless and K. Grauman, "Object-centric spatio-temporal pyramids for egocentric activity recognition," in *BMVC*, Bristol, United Kingdom, 2013, vol. 2, p. 3.

[16] H. Pirsiavash and D. Ramanan, "Detecting activities of daily living in first-person camera views," in *2012 IEEE Conf. Comput. Vision Pattern Recognit.*, Zurich, Switzerland, 2012, pp. 2847–2854.

[17] G. Vaca-Castano, S. Das, J. P. Sousa, N. D. Lobo, and M. Shah, "Improved scene identification and object detection on egocentric vision of daily activities," *Comput. Vis. Image Underst.*, vol. 156, pp. 92–103, 2017.

[18] Y. Yan, E. Ricci, G. Liu, and N. Sebe, "Egocentric daily activity recognition via multitask clustering," *IEEE Trans. Image Process.*, vol. 24, no. 10, pp. 2984–2995, 2015.

[19] X. Wang, L. Gao, J. Song, X. Zhen, N. Sebe, and H. T. Shen, "Deep appearance and motion learning for egocentric activity recognition," *Neurocomputing*, vol. 275, pp. 438–447, 2018.

[20] S. Bambach, S. Lee, D. J. Crandall, and C. Yu, "Lending a hand: Detecting hands and recognizing activities in complex egocentric interactions," in *Proc. IEEE Int. Conf. Comput. Vision*, Santiago, Chile, 2015, pp. 1949–1957.

[21] C. F. Crispim-Junior, V. Buso, K. Avgerinakis, G. Meditskos, A. Briassouli, J. Benois-Pineau, I. Y. Kompatsiaris, and F. Bremond, "Semantic event fusion of different visual modality concepts for activity recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 8, pp. 1598–1611, 2016.

[22] C. F. Crispim-Junior, A. G. Urıa, C. Strumia, M. Koperski, A. Konig, F. Negin, S. Cosar, A. T. Nghiem, D. P. Chau, G. Charpiat, *et al.*, "Online recognition of daily activities by color-depth sensing and knowledge models," *Sensors*, vol. 17, no. 7, p. 1528, 2017.

[23] G. Meditskos, P.-M. Plans, T. G. Stavropoulos, J. Benois-Pineau, V. Buso, and I. Kompatsiaris, "Multi-modal activity recognition from egocentric vision, semantic enrichment and lifelogging applications for the care of dementia," *J. Vis. Commun. Image Represent.*, vol. 51, pp. 169–190, 2018.

[24] I. Gonzalez-Diaz, J. Benois-Pineau, J. P. Domenger, D. Cattaert, and A. de Rugy, "Perceptually-guided deep neural networks for ego-action prediction: Object grasping," *Pattern Recognit.*, vol. 88, pp. 223–235, 2019.

[25] P. Giannakeris, P. C. Petrantonakis, K. Avgerinakis, S. Vrochidis, and I. Kompatsiaris, "First-person activity recognition from micro-action representations using convolutional neural networks and object flow histograms," *Multimedia Tools Appl.*, pp. 1–21, 2020.

[26] S. Bambach, D. J. Crandall, and C. Yu, "Viewpoint integration for hand-based recognition of social interactions from a first- person view," in *Proc. 2015 ACM Int. Conf. Multimodal Inter.*, Seattle, Washington, USA, 2015, pp. 351–354.

[27] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," in *Eur. Conf. Comput. Vision*, Springer, Amsterdam, Netherlands, 2016, pp. 21–37.

[28] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv preprint arXiv: 1409.1556, 2014.

[29] Tobii Dynavox LLC, "Sono Flex App," 2019. [Online]. Available: https://www.tobiidynavox.com/software/content/sono-flex-for-communicator-5/?MarketPopupClicked=true, Accessed on: Apr. 2, 2021.

[30] AssistiveWare, "Proloquo2go App," 2016. [Online]. Available: http://www.mycoughdrop.com, Accessed on: Apr. 2, 2021.

[31] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollar, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *Eur. Conf. Comput. Vision*, Springer, Zurich, Switzerland, 2014, pp. 740–755.