Accessible Automation: Evaluating Object Segmentation Solutions for Parent-Child Interaction Research*

Craig D. J. Thompson, Cátia M. Oliveira[1], Ziye Zhang[1], Yu-Kun Lai, and Hana D'Souza

***Abstract*—Lab-based parent-child interaction (PCI) studies enable researchers to observe real-time behaviours in a controlled setting. With the rise of head-mounted eye-tracking and cameras, these studies now capture even richer data. However, extracting meaningful variables often requires time-consuming manual annotation. One key variable of interest to developmental scientists, due to its links to attention and learning, is the size of objects in the child's view. Manually extracting object sizes from a single 5-minute recording (9,000 frames at 30 FPS) can take up to 225 hours. Advances in computer vision now offer automated solutions. In this study, we evaluated six automated object segmentation solutions for their ability to extract object size from PCI videos featuring distinctly coloured objects: Colour-based extraction, Segment and Track Anything (SAM-Track), Segment Anything Model 2 (SAM2), DeepLabv3, PyTorch U-Net, and You Only Look Once (YOLOv11). Some solutions require minimal setup (Colour-based extraction, SAM-Track, and SAM2), whilst others require custom training by providing manually annotated frames (DeepLabv3, PyTorch U-Net, and YOLOv11). Two of the out-of-the-box models (SAM-Track and SAM2) and two of the custom-trained models (PyTorch U-Net and YOLOv11) demonstrated very high object segmentation accuracy (median Dice scores = .92 – .96; median IoU scores = .85 – .92). Therefore, these tools offer a scalable and accessible way to automate object segmentation, reducing annotation time from months to hours, and thus enabling broader application of this approach in developmental science.**

I. INTRODUCTION

Language development is embodied and embedded in social contexts [1]. Thus, one of the key types of studies used to understand it is parent-child interaction (PCI). PCI studies provide researchers with the opportunity to observe how children interact with objects and social agents [2]. This research has demonstrated that a crucial aspect for understanding how young children learn language through interaction with others is their visual experiences [3], [4]. Although these experiences were notoriously difficult to capture in the past, rapid advancements in head-mounted eye-tracking and camera (HMET/headcam) technology have made this increasingly possible. These methods have enabled researchers to start analysing children's visual scenes, revealing critical visual patterns that shape word learning and object recognition [5].

Even though HMET/headcams have generated transformative insights in developmental science, their use, unfortunately, remains limited to a small number of research teams. One reason for this is the sheer volume of data that must be manually processed, which is highly prohibitive. For example, a key variable researchers may be interested in extracting is the sizes of objects in the child's view, due to their links to attention and learning [6]. A typical 5-minute HMET recording yields approximately 9,000 frames (5 minutes × 60 seconds/minute × 30 frames/second). If one would want to manually segment three objects in each frame, spending around 1.5 minute per frame, this would require roughly 225 hours to fully annotate every frame in a single 5-minute recording. A well-powered study would require thousands of hours of manual annotation, severely limiting the scalability of this state-of-the-art approach. Rapid developments in the field of computer vision, however, promise new avenues for addressing this issue.

In this paper, we evaluated current object segmentation solutions for a well-established lab-based parent-child interaction paradigm involving distinctly coloured objects [4], [6]. Using this design, researchers have been able to better understand key components of word learning, particularly the interplay between attention and object naming during moments of object dominance (i.e., when one of the objects dominates in the child's view).

In the past, studies extracted object sizes by utilising a variety of techniques, including pixel-based segmentation with Gaussian mixture models for object identification [3]. This method would produce pixel 'blobs' that could be used to determine the size of objects within images, however this made it hard to differentiate objects and hands during moments of overlap. Another solution previously adopted for tackling the same problem used a Graph Cut approach with optical flow to predict masks for upcoming frames, building from frequent manual polygon markings to determine object sizes within a scene [7]. Nevertheless, this method requires frequent manual intervention, thus limiting its scalability when handling large datasets. Taking this into consideration, in this paper, we focused on how recent developments in the field of computer vision can remedy some limitations of the previous solutions.

*Research supported by an EPSRC PhD Studentship awarded to C. D. J. Thompson (EP/W524682/1 - 2925134), and a UKRI Future Leaders Fellowship (MR/X032922/1) awarded to H. D'Souza. Dataset generation supported by a Baily Thomas Charitable Fund Grant, Beatrice Mary Dale Research Fellowship (Newnham College, University of Cambridge), University of Cambridge Department of Psychology Equipment Fund, and Isaac Newton Trust Project Grant awarded to H. D'Souza.

C. D. J. Thompson (corresponding author; ThompsonC21@cardiff.ac.uk, +44 2922514800), C. M. Oliveira (ferreiradeoliveirac@cardiff.ac.uk), and H. D'Souza (dsouzah@cardiff.ac.uk) are with the Centre for Human Developmental Science, School of Psychology, Cardiff University, Tower Building, 70 Park Place, Cardiff, CF10 3AT, UK.

C. D. J. Thompson, Z. Zhang, and Y. Lai (LaiY4@cardiff.ac.uk) are with the School of Computer Science and Informatics, Cardiff University, Cardiff, CF24 4AG, UK.

[1]These authors contributed equally to this work.

We compared state-of-the-art segmentation solutions to evaluate their performance and effectiveness under typical lab-based PCI conditions. The following solutions were assessed: Colour-based extraction, Segment and Track Anything (SAM-Track [8]), Segment Anything Model 2 (SAM2 [9]), DeepLabv3 [10], PyTorch U-Net [11], and You Only Look Once (YOLOv11 [12]). These segmentation algorithms vary in industry and research use cases, employing different frameworks and training regimes, as well as requiring different run times. For example, some of the solutions (Colour-based extraction, SAM-Track, and SAM2) employ zero-shot tracking, meaning the methods require no prior custom training to segment objects, whilst others (DeepLabv3, PyTorch U-Net, and YOLOv11) require specific training on manually annotated images to be able to correctly segment objects. We evaluated these solutions using the Dice coefficient and Intersection over Union (IoU) between manually annotated ground truth masks and predicted segmentation outputs. We also considered how these solutions compare not only on computational demands but also on usability.

II. Methodology

A. Data Source

The data used in the current project was taken from a larger word learning study examining PCI in 30 parent-child dyads with children's ages between 17.3 and 58.4 months. This study comprised two groups: young children with Down syndrome (DS) and young typically developing (TD) children, matched on ability level [13]. The study employed head-mounted eye-tracking (Positive Science, LCC) to capture the participants' views as they interacted with novel objects. Here we analyse the children's headcam recordings (variable frame rate ~30 FPS; 640 x 480 resolution).

The study took place in a controlled laboratory setting, where the child and their parent were seated at opposite ends of a white table (85 x 61 x 73 cm). Each dyad took part in four play trials (90 seconds per trial). During these trials, participants engaged with a total of six novel objects that were organised into two alternating sets of three. For full details, see [6], [13], [14]. This design was implemented with machine learning applications in mind. Uniformly coloured distinct objects were used to allow for a more effective extraction of object size.

For the current paper, we focused on a subset of this data. For our training set, we randomly selected two children with DS (36 and 44 months) and two TD children (23 and 24 months). For our validation set, we randomly selected an additional two children (48-month-old child with DS and TD 21-month-old child). Finally, two more children were randomly selected for our test set (58-month-old with DS and TD 23-month-old). All datasets included data from children with DS and TD, ensuring a balanced representation. From our selected subset of data, we obtained a total of 2,880 seconds of footage (86,314 frames).

B. Preprocessing Steps

Before testing segmentation solutions on the data, some preprocessing steps were necessary to improve consistency and visual clarity. These preprocessing steps included video-resampling from variable to fixed frame rate (30 FPS), deinterlacing using HandBrake [15], and increasing colour saturation by 1.8x using FFmpeg [16]. These steps were applied to mitigate frame inconsistencies, visual artifacts, and to enhance object distinction. As shown in Fig. 1, after preprocessing, the data showed improved colour fidelity and compensation for motion-related distortions.

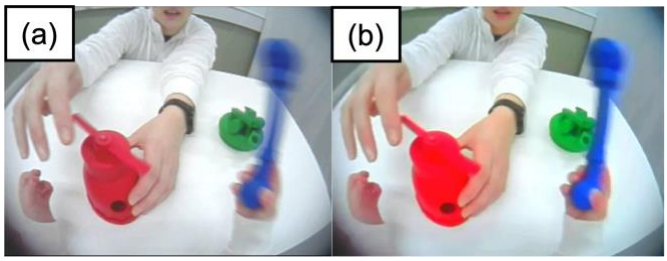

Figure 1. Demonstrating the effects of preprocessing on headcam frames. (a) Unprocessed raw frame; (b) Processed frame with deinterlacing and colour enhancement.

C. Training and Accuracy Calculation

To train and evaluate the segmentation solutions, we manually annotated object boundaries. Prior to that, the data for the training and test sets were down sampled at 1/2 Hz, meaning that one frame was sampled for every 2 seconds, resulting in a total 836 frames for training, and 392 frames for testing. The sampling rates for training and test sets were selected based on a study that found that a sampling rate of 1/5 Hz was sufficient for capturing major regularities in an infant's visual scene, with no significant differences observed even when compared to 1/10 Hz [17]. Adopting a higher sampling rate provided confidence that our datasets would capture relevant visual regularities. This method of down sampling allows for an adequate trade-off between obtaining enough object variability between frames whilst also ensuring that the dataset is manageable for manual annotation to be conducted to a high standard in a reasonable timeframe. As the training sample size is likely to impact the segmentation accuracy, a robustness analysis was conducted to explore the effect of training sample size on each model. Data for the training validation set were down sampled at a rate of 1 frame every 8 seconds (1/8 Hz), as a smaller sample of frames was sufficient for capturing the variability between datasets during post-epoch evaluations. The frames were manually annotated in SuperAnnotate [18] by two trained annotators. To assess inter-rater reliability, coder 1 annotated 20% of coder 2's frames, resulting in a high median Dice coefficient of .99 and IoU of .98 (for equations, see Fig. 3a), indicating strong agreement.

All custom-trained models in this study were trained for 100 epochs to ensure adequate learning time. During each epoch, the model was exposed to all 836 images, with image transformations at each stage to increase variability in the training input. The model's performance was evaluated after each epoch using the validation set, with the best performing checkpoint selected as the trained model for this study.

DeepLabv3 and YOLOv11 were both trained using pretrained models ('deeplabv3_resnet101' [10] and 'yolo11l-seg.pt' [12], respectively), leveraging existing

detection architectures to identify our custom objects, whereas PyTorch U-Net was trained from scratch.

To evaluate model performance, we used both the Dice coefficient and IoU (also known as Jaccard index) (see equations in Fig. 3a) to measure the accuracy between the predicted segmentation masks and the manually annotated ground truth masks (see Fig. 2). Both indices are commonly used to evaluate performance of segmentation algorithms. The Dice coefficient quantifies segmentation accuracy by measuring the overlap between two masks. This coefficient is computed by doubling the number of overlapping pixels between the two masks, divided by the total number of pixels in both masks. The Dice coefficient performs particularly well at identifying the true positives (i.e., where the two masks overlap). The IoU quantifies accuracy by dividing the number of overlapping pixels between the two masks by the area of union between the two masks. While IoU also captures true positives, it is stricter because it penalises false positives (pixels present in the predicted mask, but not in the ground truth) and false negatives (pixels missing from the predicted mask but present in the ground truth) more heavily.

Both the Dice coefficient and IoU vary between 0 (low accuracy) and 1 (high accuracy). Model accuracy was compared across solutions by fitting a robust linear mixed model using the *robustlmm* package [19], with model as a fixed effect and participant as a random effect. Pairwise model comparisons were conducted using estimated marginal means via the *emmeans* package [20].

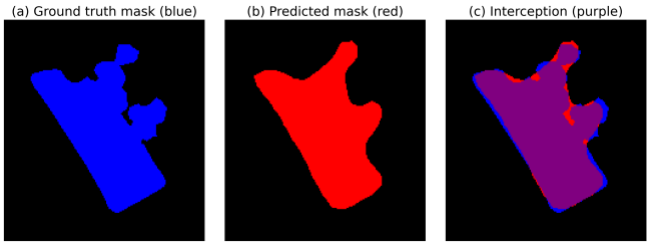

Figure 2. A visualisation of intersection calculation. (a) Ground truth mask in blue; (b) Predicted segmentation mask in red; (c) Intersection of the two masks (purple). Dice = .94; IoU = .89.

III. EVALUATION

To test possible segmentation solutions, several computer vision methods were evaluated in terms of accuracy, computational demands, and usability. These include both out-of-the-box solutions (Colour-based extraction, SAM-Track, and SAM2), as well as trained models (DeepLabv3, PyTorch U-Net, and YOLOv11).

A. Out-of-the-Box Solutions

1) Colour-based extraction

As a baseline, a colour-based extraction method was implemented to identify and segment specific colour ranges within the visual scene. The HSV (Hue, Saturation, Value) colour space was used for this method instead of the standard RGB (Red, Green, Blue) colour space, based on studies such as [21], which suggest that image segmentation often performs more accurately in HSV as it aligns better with human colour perception and reduces intra-class variance caused by lighting conditions. The reduced sensitivity to variations in lighting occurs due to the separation of colour and brightness information, thus improving segmentation robustness in different lighting conditions.

2) Segment and Track Anything (SAM-Track)

The first computer vision model evaluated in this study is the SAM-Track model [8]. This model differs from its underlying baseline Segment Anything Model (SAM) by incorporating functionality to perform object tracking and segmentation in videos, rather than solely individual frames. This is achieved via integrating SAM with the Transformers Framework DeAOT, an efficient multi-object tracking mode to track objects through videos. SAM-Track is a zero-shot model, which means that beyond providing positive prompts for the first 3 trackable objects in the first frame using a graphical interface, no further training or human intervention is required. Therefore, compared to the other object segmentation solutions, this is the most user-friendly model for researchers who are less familiar with machine learning techniques.

3) Segment Anything Model 2 (SAM2)

SAM2 [9] is the recently released successor to the original SAM. It represents a significant evolution from the original by natively integrating advanced segmentation capabilities for both images and videos. This model employs a new transformer-based architecture that includes an innovative memory mechanism for real-time video processing. We selected SAM2 due to its advanced segmentation capabilities, with the ability to process both images and videos without requiring additional fine-tuning. For this study, SAM2 was used with a pre-trained model that was not trained on our novel object data. Like SAM-Track, prompt points were required to effectively track objects throughout the video. However, unlike SAM-Track, it lacks a graphical interface, making it less accessible to users without coding experience.

B. Custom-Trained Solutions

1) DeepLabv3

DeepLabv3 [22] is a deep convolutional neural network that employs atrous convolution, enabling multi-scale contextual information without sacrificing spatial resolution. By combining atrous convolution with atrous spatial pyramid pooling, DeepLabv3 is particularly effective for complex scenes, including object overlap and fine-grained detail. Originally introduced in 2016 [23], DeepLabv3 has since become widely adopted and popular among biomedical researchers [24].

2) PyTorch U-Net

A U-Net based Convolutional Neural Network (CNN [25]) was designed for integration with PyTorch, an open-source deep learning library. Despite the U-Net architecture being much older than other solutions in this study, U-Net remains a widely used method in biomedical image segmentation due to its unique architecture, high performance, and simplicity [25]. It consists of a contracting

path used to capture contextual features, and a symmetric expanding path that enables precise localisation. This design also allows the network to learn effectively from a relatively small number of annotated images while maintaining high segmentation accuracy.

3) You Only Look Once (YOLOv11)

You Only Look Once (YOLO [26]) is a widely recognised and extensively used object detection model, known for its continuous improvements in accuracy. This is a single-shot approach that predicts bounding boxes and class probabilities in one evaluation using a single neural network. Each new iteration of the model architecture (with the newest being YOLOv11 [12]) provides better performance. YOLOv11 was chosen for its efficiency in processing large volumes of data as well as efficiency in object detection.

IV. Results

A. Model Performance Comparison

1) Out-of-the-box methods

As evidenced in the segmentation masks in Fig. 3c-e, the out-of-the-box methods often struggle when faced with frames that include overlapping objects (Fig. 3d) or images with heavy motion blur (Fig. 3e). While all out-of-the-box methods showed some inconsistency in their performance, this was especially evident in Colour-based extraction, with many incorrect masks in each frame. SAM-Track's and SAM2's performance showed the highest overall accuracy out of the three out-of-the-box solutions (Fig. 3b), with SAM-Track achieving a Dice score of .96 and IoU score of .91, and SAM2 achieving a Dice score of .96 and IoU score of .92. These results were obtained even though prompt points were only given for the first 3 coloured objects in the participants recording. To further evaluate the models, we repeated this process but instead provided prompt points at the start of each play trial, rather than solely at the start of each participant's recording. This allowed the models to receive positive prompts for each object being tracked, rather than generalising to objects across trials. The impact of this on the models' performance was negligible (SAM-Track: Dice = .96, IoU = .92; SAM2: Dice = .95, IoU = .91).

2) Custom-trained methods

As shown in Fig. 3b, all custom-trained methods performed well and yielded comparable results (DeepLabv3: Dice = .92, IoU = .85; PyTorch U-Net: Dice = .95, IoU = .91; YOLOv11: Dice = .95, IoU = .91) with a maximum overall variation of .03 for Dice and .06 for IoU. Nonetheless, all methods suffered from instances of overclassification. Examples of these include masks where hands were misclassified as a red object, or small mask artifacts were produced. The latter issue occurred only for PyTorch U-Net. Therefore, before evaluation, a small mask filter was introduced for this model to remove masks under 100 pixels. These overclassification instances were likely better captured by the IoU score, as this method penalises false positives more harshly than the Dice coefficient. Another challenge for the models were overlapping objects - YOLOv11 was the only model to segment an object through the opening of another (see Fig. 3d). However, it often underperformed by producing segmentation masks with smoother edges, which negatively impacted its Dice score. Overall, PyTorch U-Net and YOLOv11 showed the best performing solution in terms of both Dice and IoU scores.

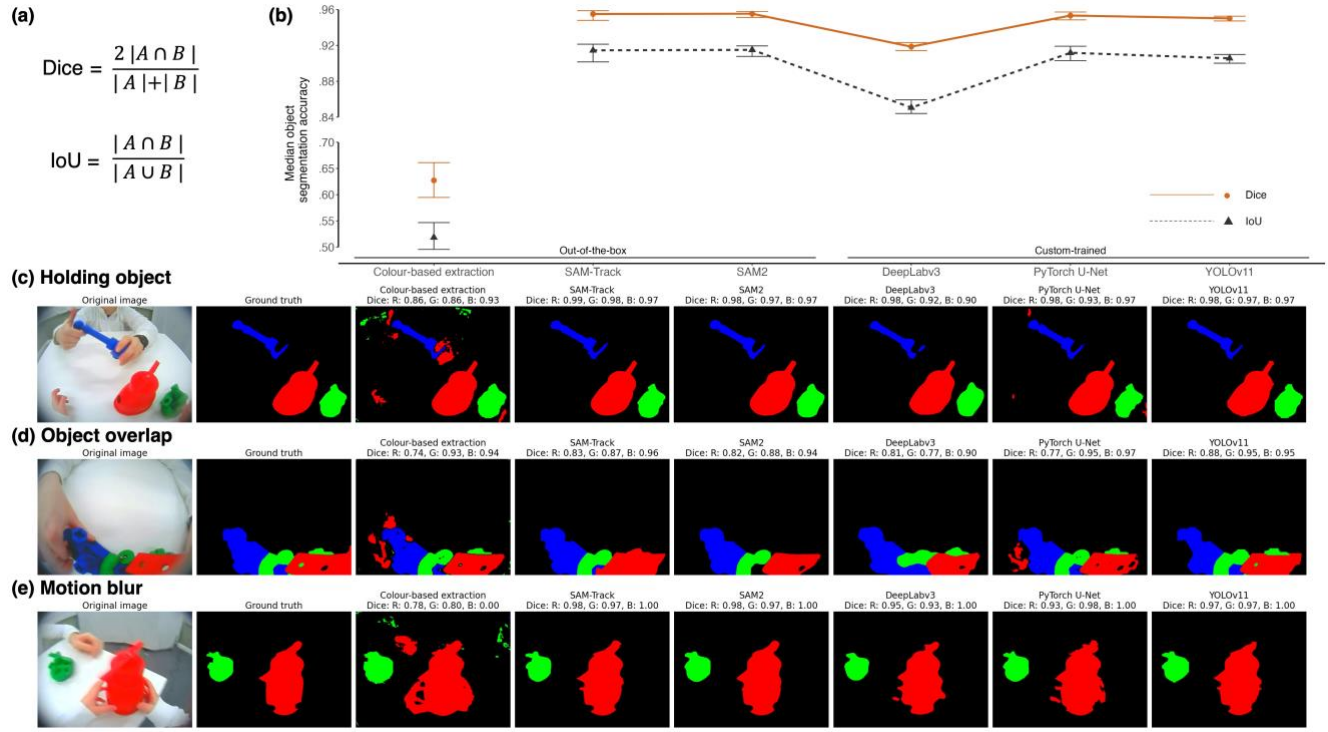

Figure 3. (a) Equations used for Dice and IoU calculation; (b) Segmentation accuracy of each method, with Dice shown in orange and IoU in grey; (c)-(e) Comparisons of models under challenging conditions: (c) Holding object; (d) Object overlap; and (e) Motion blur. Left to right: Original image, ground truth, Colour-based extraction, SAM-Track, SAM2, DeepLabv3, PyTorch U-Net, and YOLOv11.

3) Overall model comparison

The robust linear mixed model and the pairwise contrasts using *emmeans* revealed that Colour-based extraction significantly underperformed when compared to other solutions ($p < .05$) on both metrics. For both Dice and IoU scores, there was no difference in performance between SAM2, PyTorch U-Net, and YOLOv11. These models outperformed DeepLabv3. For Dice, YOLOv11 showed higher accuracy than SAM-Track. This pattern was not found for IoU, as both YOLOv11 and SAM-Track showed comparable accuracy, and all four models outperformed DeepLabv3 (see appendices for more details: https://osf.io/syebh).

B. Impact of Training Sample Size

To evaluate how annotation density affects segmentation accuracy, each custom-trained model was trained under 4 different training conditions. These include sampling at 1/2 Hz, 1/4 Hz, 1/6 Hz, and 1/8 Hz. Table 1 shows that both PyTorch U-Net's and YOLOv11's segmentation accuracy is consistently high through every training density tested, even when presented with only 209 training images (1/8 Hz), while DeepLabv3 shows a steady decline. This strong performance is likely a result of PyTorch U-Net's unique architectural design, and YOLO's large pre-trained detection model.

TABLE 1. Impact of training saturation across trained solutions (medians).

Model	Index	Training sample size			
		1/2 Hz	1/4 Hz	1/6 Hz	1/8 Hz
DeepLabv3	Dice	.92	.89	.86	.66
	IoU	.85	.80	.76	.53
PyTorch U-Net	Dice	.95	.95	.94	.92
	IoU	.91	.91	.90	.85
YOLOv11	Dice	.95	.95	.95	.95
	IoU	.91	.90	.90	.90

C. Applications

1) Object size and position analysis

By using the segmentation masks obtained from the most robust model (YOLOv11), we were able to conduct further analyses, including automatic calculations of object size and their location (see Fig. 4). Object size is quantified by calculating the total number of pixels covered by a given mask (visible object) relative to the total pixels within the visual scene. In addition, spatial data can be gathered by forming a boundary box around each object. This allows us to pinpoint an object's centre and determine its location within the frame. These metrics can then be leveraged for further analyses, depending on the research question.

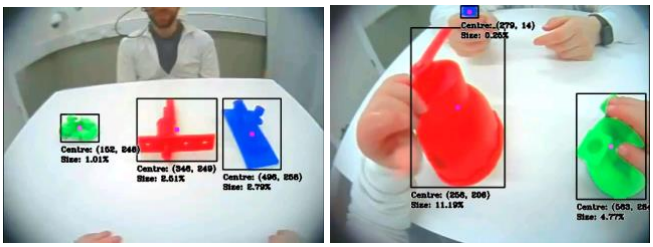

Figure 4. Visualisation of object sizes within the play trial, including boundary boxes and centre coordinates outlining location using masks from YOLOv11.

2) Identifying visual object dominance

By collecting object size data from section C.1, we can apply this to answer research questions about the role of object dominance in PCI [4], [13], [27] (see Fig. 5). This requires minimal manual intervention between providing video frames and obtaining size data, allowing for large datasets to be analysed quickly.

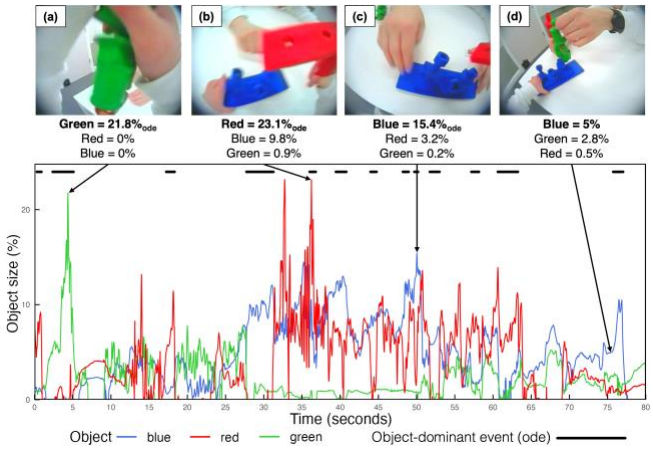

Figure 5. Moments of visual dominance within a play session. Frame (a), (b) and (c) are defined as containing a dominant object (i.e., object-dominant events) as its object size comprised at least 5% of the image, its relative size is greater than 50% of all other objects in view, and these conditions lasted over 500 ms [13].

V. Discussion

This study aimed to evaluate a range of out-of-the-box and custom-trained computer vision solutions for object segmentation in lab-based PCI studies. The results suggest that custom-trained models show comparable performance to some of the out-of-the-box solutions in terms of accuracy and accessibility. Specifically, SAM-Track and SAM2 showed a comparable Dice and IoU accuracy score to the custom-trained models. Nevertheless, this was not the case for Colour-based extraction, with accuracy scores well below those of other models.

Out of the 3 trained solutions used in this study, PyTorch U-Net and YOLOv11 achieved the highest Dice and IoU scores. Given the comparable performance for these models, if speed and ease of implementation is a priority, YOLOv11 was notably faster and easier to implement than the other solutions, with detailed guides and premade training scripts available [12]. Furthermore, YOLOv11 excelled in its detection and segmentation, showing a particularly high and consistent performance as evidenced by narrow confidence intervals presented in Fig. 3b, was the only model capable of segmenting objects through openings in other objects, and it was robust to the manipulation of training sample size. However, this model is not without limitations as due to its reliance on initial object detection using bounding boxes, this approach introduces subtle artifacts on the boundaries of objects, such as flat edges, which are noticeable in some resulting masks.

When considering out-of-the-box solutions, SAM2 and SAM-Track emerged as the most effective models, both achieving strong accuracy, comparable to those of custom-

trained models. Thus, these models are excellent options as they require minimal setup and zero custom data training. One limitation of the zero-shot solutions is that longer videos demand significant memory allocation, which can result in out-of-memory issues. This issue was observed when running SAM2, which loaded all frames into memory before processing, requiring an increase in available memory compared to other models. Whilst this issue did not arise for SAM-Track, it is possible that it might occur if analysing bigger datasets than those included in this study. Taking this into consideration, SAM-Track emerges as a better alternative since it also provides a more intuitive and user-friendly interface. This lowers the barriers to entry for researchers with limited technical expertise. Recently, new tools such as Low-Rank Adaptation (LoRA [28]) have been developed which enable efficient fine-tuning of large models like SAM, allowing for more tailored output. However, these tools are less accessible to researchers with limited knowledge of how to implement such techniques.

Controlled settings, like those utilised in the current study, allow for certain practices that can enhance the effectiveness of automated object segmentation. These should be (and often are) considered at the design and data collection stages of lab-based studies. Researchers can adopt a background colour distinct from the objects, use objects that are distinct from each other, ensure the room is adequately lit, and reduce the number of objects in the room unrelated to the study. These steps would maximise the accuracy of automated segmentation pipelines, which require minimal setup time and allow meaningful data to be extracted from hours of video in a fraction of the time that would be required from manual annotation. Most of the models tested in the current study indeed demonstrated high accuracy. However, it remains uncertain how well these solutions generalise from our controlled lab-based study to more naturalistic, everyday videos. Resolving this is essential for enabling large-scale naturalistic studies, further advancing our theories of development, and informing diagnostic and intervention approaches.

ACKNOWLEDGMENT

The authors would like to thank all the families who contributed their time to this study. This work would also not have been possible without the support of the Cardiff Babylab team. We would particularly like to thank Kate Mee and Sophia Ivackovic, for their help with data collection and manual annotation, respectively.

REFERENCES

[1] C. S. Tamis-LeMonda and L. R. Masek, "Embodied and Embedded Learning: Child, Caregiver, and Context," *Curr. Dir. Psychol. Sci.* , vol. 32, no. 5, pp. 369–378, Jul. 2023.

[2] S. H. Suanda, M. Barnhart, L. B. Smith, and C. Yu, 'The signal in the noise: the visual ecology of parents' Object Naming', *Infancy*, vol. 24, no. 3, pp. 455–476, 2019.

[3] A. F. Pereira, H. Shen, L. B. Smith, and C. Yu, 'A first-person perspective on a parent-child social interaction during object play', *Proc. Annu. Meet. Cogn. Sci. Soc.*, vol. 31, no. 31, Jan. 2009.

[4] L. B. Smith, C. Yu, and A. F. Pereira, 'Not your mother's view: the dynamics of toddler visual experience', *Dev. Sci.*, vol. 14, no. 1, pp. 9–17, Jan. 2011.

[5] J. M. Franchak, K. S. Kretch, K. C. Soska, and K. E. Adolph, 'Head-mounted eye tracking: A new method to describe infant looking', *Child Dev.*, vol. 82, no. 6, pp. 1738–1750, Oct. 2011.

[6] C. Chen, D. M. Houston, and C. Yu, 'Parent–child joint behaviors in novel object play create high-quality data for word learning', *Child Dev.*, vol. 92, no. 5, pp. 1889–1905, Aug. 2021.

[7] Q. Mirsharif, S. Sadani, S. Shah, H. Yoshida, and J. Burling, 'A semi-automated method for object segmentation in infant's egocentric videos to study object perception', 2016, *arXiv:1602.02522*.

[8] Y. Cheng et al., 'Segment and Track Anything', 2023, *arXiv:2305.06558*.

[9] N. Ravi *et al.*, 'SAM 2: Segment anything in images and videos', 2024, *arXiv: 2408.00714*.

[10] PyTorch. 'Deeplabv3.', Pytorch. Accessed: Mar. 5, 2025. [Online] Available: https://pytorch.org/hub/pytorch_vision_deeplabv3_resnet101/

[11] milesial. 'Pytorch-UNet.' Github. Accessed: Mar. 5, 2025. [Online]. Available: https://github.com/milesial/Pytorch-UNet

[12] Ultralytics. 'Ultralytics.' Github. Accessed: Apr. 02, 2025. [Online]. Available: https://github.com/ultralytics/ultralytics.

[13] C. Bocchetta, C. D. J. Thompson, C. Suarez-Rivera, C. Yu, and H. D'Souza, 'Same scenes, different movements: Different dyadic motor patterns underlie object dominance in typically developing and neurodivergent young children', *submitted for publication*.

[14] C. Yu and L. B. Smith, 'Multiple sensory-motor pathways lead to coordinated visual attention', *Cogn. Sci.*, vol. 41, pp. 5–31, Mar. 2016.

[15] The HandBrake Team. 'HandBrake: The open source video transcoder'. Accessed: Mar. 17, 2025. [Online]. Available: https://handbrake.fr/

[16] FFmpeg. 'FFmpeg'. Accessed: Mar. 26, 2025. [Online]. Available: https://www.ffmpeg.org/

[17] C. M. Fausey, S. Jayaraman, and L. B. Smith, 'From faces to hands: Changing visual input in the first two years', *Cognition*, vol. 152, pp. 101–107, Jul. 2016.

[18] SuperAnnotate AI. 'SuperAnnotate'. Accessed: Mar. 17, 2025. [Online]. Available: https://www.superannotate.com.

[19] M. Koller, 'robustlmm: An R package for robust estimation of linear mixed-effects models', *J. Stat. Softw.*, vol. 75, no. 6, pp. 1–24, Dec. 2016.

[20] Lenth, R.V. Emmeans: Estimated Marginal Means, Aka Least-Square Means. (2021). [Online]. Available: https://CRAN.R-project.org/package=emmeans

[21] Z. Shu, G. Liu, Z. Xie, and Z. Ren, 'Segmentation algorithm of color block target captured by CCD camera based on region growing', in *2016 3rd International Conference on Information Science and Control Engineering (ICISCE)*, Jul. 2016, pp. 597–600.

[22] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, 'Rethinking atrous convolution for semantic image segmentation', 2017, *arXiv:1706.05587*.

[23] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, 'DeepLab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs', 2016, *arXiv:1606.00915*.

[24] S. Vedpathak, P. Soni, S. Gaikwad, and M. Parmar, '2D Brain MRI segmentation: U-Nets versus optimized DeepLab Models', in *2024 IEEE International Conference on Information Technology, Electronics and Intelligent Communication Systems (ICITEICS)*, Jun. 2024, pp. 1–6.

[25] O. Ronneberger, P. Fischer, and T. Brox, 'U-Net: convolutional networks for biomedical image segmentation', 2015, *arXiv:1505.04597*.

[26] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, 'You only look once: Unified, real-time object detection', 2016, *arXiv:1506.02640*.

[27] C. Yu, L. B. Smith, H. Shen, A. F. Pereira, and T. Smith, 'Active information selection: Visual attention through the hands', *IEEE Trans. Auton. Ment. Dev.*, vol. 1, no. 2, pp. 141–151, Aug. 2009.

[28] Z. Zhong, Z. Tang, T. He, H. Fang, and C. Yuan, 'Convolution Meets LoRA: Parameter Efficient Finetuning for Segment Anything Model', 2024, *arXiv:2401.17868*.