Can VLM replace human annotators in human-in-the-loop systems?

Advanced Deep Learning in Computer Vision (02501)

Group 12: German Buttiero (s233660), Artur Adam Habuda (s233190), Hassan Hotait (s203211)

Introduction

While human-in-the-loop (HITL) systems enhance computer vision performance, they introduce significant scalability challenges. Our project replaces human annotators with Vision-Language Models in feedback loops, preserving expert guidance while eliminating manual bottlenecks. This **VLM-in-the-loop** approach maintains decision quality while enabling unlimited scalability.

Dataset

- Dataset constructed from COCO 2017, focusing on four diverse object categories: **books, birds, stop signs, and zebras**.
- Filtered images to include only high-quality annotations with minimum object area thresholds ensuring visibility.

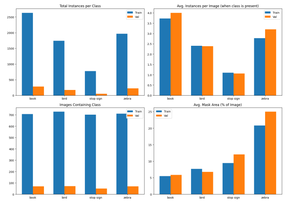


Figure 1. Dataset statistics

Data Split	Description	Proportion	Exact Count
Complete Dataset	All images	100%	3052
Training Dataset	Main training set	91.5%	2792
Test Dataset	Evaluation set	8.5%	260

Data Split	Description	Proportion	Exact Count
Training Set	For model training	90% of training dataset	2513
Validation Set	For model validation	10% of training dataset	279
Test Set	For evaluation	Independent	260

Data Split	Description	Proportion	Exact Count
Fixed Validation Set	For early stopping & model selection	10% of training	279
Active Learning Pool	Remaining training images	90% of training	2513
→ Initial Training Set	Initial labeled training data	20% of AL Pool (default)	503
→ Inference Pool	Images waiting for feedback	80% of AL Pool	2010
Test Dataset	Final evaluation set	Independent	260

Figure 2. Dataset splits

Architecture

HITL approach requires initial labeled data to train a model and relies on human annotators to verify generated masks, creating bottlenecks in scaling and consistency. Our proposal, **VLM-ITL**, replaces human verification with a VLM (although it still requires labeled data to begin the annotation process).

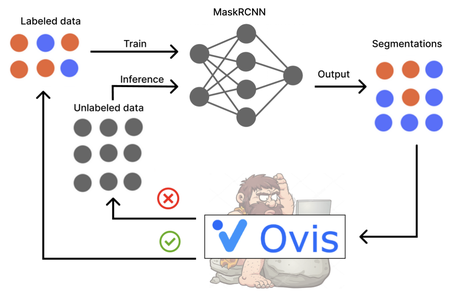


Figure 3. **VLM in the loop**: the human annotator is replaced by a VLM that decides whether the segmentations are correct or not.

VLM-Prompt: Examine this image showing object segmentation masks.
The colored areas represent the computer's identification of objects in the image.
The computer detected these objects: `{pred_class_names_str}`.
Answer with:
- Begin with "yes" if both the object classes and segmentation masks are accurate.
- Begin with "no" followed by an explanation if any objects are misclassified or poorly segmented.

Results

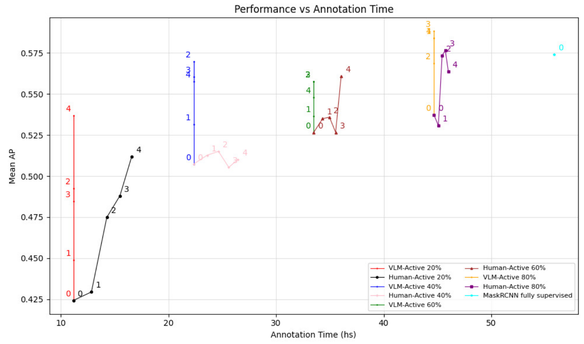


Figure 4. Results of the different experiment runs for the Human-in-the-loop approach and the VLM-in-the-loop approach. Each cluster of plots in the figure corresponds to a different initial training dataset proportion : 20%, 40%, 60%, 80%, 100%(fully supervised).

Assumptions

Parameter	Value	Description
Human annotation time	80 seconds	Average time required for a human to manually segment an object instance in the COCO dataset
Binary judgment time	3 seconds	Average time, it takes a human annotator to provide binary feedback(approve/reject) to a segmentation model generated mask
IoU approval threshold	0.6	Minimum intersection over union(IoU) score for a prediction to be automatically approved in the active learning loop

Figure 5. Several assumptions made to calculate the annotation time for different experiments and to simulated the human, in the "human in the loop" approach.[1],[2]

VLM-Prompt: You are an expert quality-controller for instance-segmentation.
Your task (think step-by-step in your head): 1. **List objects you *visually* see** in the coloured masks.
2. **Compare** that list with the model's declared detections: pred_class_names_str
3. **For every detected object**, judge whether the coloured mask tightly fits the object(60% IoU, little background or spill-over).
4. **Check for errors**: missing objects,wrong class, poor masks, extra masks / false positives. After you have finished your internal reasoning, **output only one line**:
- **"yes"** – if and only if **all** objects are correctly detected **and** every mask is good.
- **"no – <one-sentence reason>"** – otherwise.
(Do **not** reveal your private reasoning.)

Figure 6. Problematic samples judged by VLM.

Analysis and Observations

Observation	Summary	Description
Chain of Thought prompt	Positive effect ~ +0.04 map	We observe that guiding the VLM trough a reasoning process to judge the images has a positive effect on correctly judging segmentation masks. This translates directly into an increase in map performance. Although not included in the plot(for clarity), we get the following results when implementing the reasoning prompt into our active learning vlm approach: **Initial training proportion - map iteration 1 - map iteration 2:** 20 % - **0.667** - 0.685 | 40 % - **0.665** - 0.698 | 60 % - **0.674** - 0.665 |80 % - **0.690** - 0.682
VLM struggles	Clustered objects Many instances in one image Small objects Not fully consistent	The VLM struggles with classes that tend to appear in clusters or that tend to appear multiple in instances in images. We see additionally that the VLM is not good at judging weather a single object has been properly segmented, when there are multiple masks overlapping it. Finally we observe that the VLM is not always consistent in its judgements regarding similar segmented images. This might be a contributing factor to the disparity in results across different iterations and experiments. *This are qualitative observations and specific to Ovis2-8b(the vlm model employed in this work).

Figure 7. Observations on experiment runs.

First of all, the plots displayed in Figure 4 present clear **issues**. The starting point of the different runs (point 0 in both for human-simulated and vlm-active plots) at different initial training sizes, do not follow the **logical increasing map trajectory**. It is to be expected that the more ground-truth annotations the segmentation model has, the better its performance on test set will be. Despite this we see that for example for 40% of initial training set pretraining of mask-rcnn, the yielded map is lower than at 20%. Additionally the fully supervised scenario (theoretically the ideal scenario, given the availability of all gt masks for training) gets a lower map score than some of the iterated vlm-judged models at 80% initial pretraining.

We **hypothesize** that the current fixed epoch setting, that does not factor in the size of the training data, could be hindering the **convergence** of results at higher training data sizes. On top of that, the use of pretrained mask-rcnn, and its learned parameters, might be obscuring the results and effects of training on this relatively small data-subsets. Hence, we propose an **adaptive scheduling of epochs** dependent on training pool size, paired with early stopping to avoid overfitting. Additionally using a **non pretrained segmentation model** could remove uncertainty from the results.

In the calculation of annotation time, neither the annotations involved for the pretraining of Mask-rcnn and the VLM model have been considered. Both of which might have seen COCO. Quantifying the effect of these issues is challenging, and remains to be analyzed in future work.

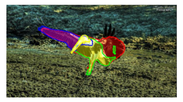
Thus, although the results at first glance suggest that the VLM-in-the-loop approach presents significant **improvements in performance** and **reductions in annotation time** with respect to an analog "human-in-the-loop" approach, we categorize our results as **inconclusive**. The inclusion of the aforementioned could potentially resolve these issues and shed some light on the results.

Example Cases: VLM judgments

A correct (green), partially correct (orange) and incorrect (red) VLM judgments are presented. Alongside, the images passed to the VLM; original image together with the overlaid masks; are provided.

"the computer detected three objects, which are the colored bird shapes on the window sill. the segmentation masks accurately identify these objects as birds, with each bird in a different color: blue, red, and green. the segmentation is clear and distinct, with no overlap between the objects. therefore, the object classes and segmentation masks are accurate.

"the computer's detection of objects in the image is not accurate. the segmentation masks, which are supposed to represent the computer's identification of objects, do not correspond to any actual objects in the image. the masks are colored and appear to be randomly placed, not aligned with any specific objects. therefore, the correct response "



"the image shows a bird in flight with various colored segmentation masks overlaid on it. the segmentation masks are used to identify different parts of the bird, such as the wings, body, and tail. the colors used for the segmentation masks are purple, blue, yellow, green, and red, which are not standard colors for object detection but are used here to distinguish different parts of the bird. the computer has detected the bird as the main object in the image. the segmentation masks appear to accurately identify the bird's different parts, with each color representing a distinct segment of the bird's anatomy. the segmentation is clear and well-defined, with no apparent misclassifications or poor segmentation.

Grounded-Segment-Anything

Both previous methods still required some labeled data to begin the process. We propose a fully automated annotation system that eliminates human intervention and requires zero labeled data to start. This investigation does not study the use of VLM in the loop, but instead is an attempt to transfer learning from the VLM(**Grounded SAM2**) to **MASK-RCNN**. **Grounded SAM2** is computationally intensive, consumes 12-24 GB of VRAM and is not designed for real-time inference (0.3-1 FPS). On the other hand, **MASK-RCNN** is light weight and real-time capable (5-10 FPS).

Therefore, it would be of great value to use as a large foundation model such as **Grounded SAM2** to auto-label data for use in training smaller finetuned models such as **MASK-RCNN** as shown in Fig 8.

Figure 8. **Grounded SAM auto-labeling**: Grounded SAM2 processes unlabeled images to generate segmentation masks, which are used as pseudo-labels to train MaskRCNN without human annotation.

Obviously, the performance of MASK-RCNN will depend heavily on the quality of the pseudo-labels generated by Grounded SAM. Therefore, we evaluate the quality of those annotations by comparing them against the ground truth COCO labels. The quality of those annotations is observed to be much lower than expected as shown in Table. 1. Furthermore, some qualitative samples are shown in Fig 8.

Set	book AP	zebra AP	stop sign AP	bird AP	mAP
Training Set	0.08	0.37	0.55	0.34	0.33

Table 1. AP per class and mAP of the pseudo-labels generated by grounded sam.

Figure 9. Sample predictions from grounded sam.

To conclude this study, we finally evaluate Mask-RCNN trained with pseudo-labels and compare it against the fully supervised one. As expected, due to the poor quality of the pseudo-labels the mAP of the auto-labeled Mask-RCNN is much less than that supervised with human annotations. We also acknowledge that the relatively okay performance of 0.49 mAP for the auto-labeled model is mainly due to the fact that we initalize our model with pre-trained weights, otherwise the performance would be alot worse.

Model	Mask-RCNN[1]	Mask-RCNN[2]
mAP	0.65	0.49

Table 2. mAP for human[1] and Grounded SAM[2] supervised Mask-RCNN.

References

[1] Abhinav Gupta et al.
Lvis: A dataset for large vocabulary instance segmentation.
In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[2] Dim P. Papadopoulos.
Scaling up instance annotation via label propagation.
In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

https://github.com/GermanButtiero/vlm-in-the-loop