# BeetleFlow: An Integrative Deep Learning Pipeline for Beetle Image Processing

**Anonymous Author(s)**
Affiliation
Address
email

## Abstract

In entomology and ecology research, biologists often need to collect a large number of insects, among which beetles are the most common species. A common practice for biologists to organize beetles is to place them on trays and take a picture of each tray to digitize them. Given the images of thousands of such trays, it is important to have an automated pipeline to process the large-scale image data for further research. Therefore, we develop a 3-stage pipeline to detect all the beetles on each tray, sort and crop the image of each beetle, and do morphological segmentation on the cropped beetles. For detection, we design an iterative process utilizing a transformer-based open-vocabulary object detector and a vision-language model to comprehensively detect all beetles in the tray. For segmentation, we manually labeled 670 beetle images and fine-tuned two variants of a transformer-based segmentation model to achieve fine-grained segmentation of beetles with relatively high accuracy. The pipeline integrates multiple deep learning methods and is specialized for beetle image processing, which can greatly improve the efficiency to process large-scale beetle data and accelerate biological research.

## 1 Introduction

In entomology and ecology research, biologists often need to collect a large number of insects to study, among which beetles are one of the most common species. Beetles account for around 25% of all known species in the world [1]. Therefore, they have significant research value in a variety of fields such as taxonomy, evolution, and biodiversity, for their species richness, widespread distribution, and representativeness.

In practice, biologists often mount beetles collected at the same site and time on a tray, using pins and in a certain order. A tray can contain from several to over 60 beetle specimens. After collection, biologists can have up to thousands of such trays and take a picture of each tray to digitize them. This procedure brings in a large amount of beetle data organized by trays, but how to process them for further study becomes a new question. On the one hand, given tens of thousands of beetles, it is an ideal amount of data to conduct machine learning study. However, machine learning algorithms need pictures of each single beetle as input, instead of the whole tray, to learn the patterns of beetles. It is time-consuming to take a picture of each beetle manually given such a large amount. On the other hand, biologists sometimes need to do manual labeling on each beetle to segment out and measure certain body parts of their interests, which is also labor-intensive. To the best of our knowledge, no work or pipeline has fulfilled this demand for large-scale beetle image processing.

Given the importance of beetle study and the challenges biologists face in processing massive beetle data, it is significant to develop an automated pipeline to process large-scale beetle images. Therefore, we develop a 3-stage pipeline utilizing multiple deep learning methods to help with beetle data processing. In the first stage, we develop an iterative process to comprehensively detect all beetles

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

in each tray image. We utilize a transformer-based open-vocabulary detector, Grounding DINO [2], for iterative beetle detection and a vision-language model, LLaVA-NeXT [3], for final verification. This approach achieves a high tray-level accuracy of 97.81%, where a tray is considered correct only if the detected beetle count exactly matches the ground-truth count. In the second stage, we crop each detected beetle from the tray image and save it as a single image. If there are digitized metadata provided for each beetle in the tray, the pipeline can also sort the detected beetles in a certain order and match each beetle to its metadata. In the third stage, we leverage an advanced and universal transformer-based model, Mask2Former [4], to segment each beetle into 5 or 9 morphological body parts. We fine-tune two variants of Mask2Former on 340 and 330 beetle images manually labeled into 5 and 9 classes respectively, and the model achieves a mean Intersection over Union (mIOU) of 85.11% for 5 class segmentation and 77.38% for 9 class segmentation.

With the pipeline, the large-scale images of trays of beetles can be efficiently processed into images of single beetles matched with metadata, which is desirable for machine learning research and single-specimen study in biology. The segmentation results also have multiple downstream usages. Firstly, biologists can acquire tens of thousands of segmented beetles within a short amount of time, saving significant effort in manual segmentation. With the digitized segmentation of different beetle body parts, the length, area, and proportion of certain parts can be automatically measured, providing high-throughput data for further biological research. Secondly, with the segmentation results, the pipeline can also detect defective specimens by finding missing classes and checking the area of each class. It helps identify poor-quality specimens from a large number of beetles, which can be useful if only high-quality specimens are needed for downstream study. Thirdly, we can easily extract body parts of our interests and do further research only on these parts, for example, exploring which body part of the beetle best reflects environmental change in its habitat.

To summarize, our contributions include developing an automated pipeline for large-scale beetle image processing, designing an iterative beetle detection process with high accuracy, and fine-tuning two beetle segmentation models with two levels of segmentation granularity. Multiple downstream research on beetles can be conducted based on the high throughput data processed by the pipeline. Moreover, the pipeline has the potential to generalize to more biological data processing cases, as we have observed a similar detection-and-segmentation pattern in other biological pipelines, such as QuPath [5] for cells and PlantCV [6] for plants. This detection-and-segmentation approach is applicable to a variety of biological data processing workflows, and our work on beetles also sets an example for insects, which are one of the most numerous groups of organisms in the world.

## 2 Related Work

**Open-Vocabulary Detection and Vision-Language Models.** The reliance of traditional object detectors like the R-CNN family [7] and YOLO [8] on large-scale, predefined training datasets is a significant limitation to scalability, particularly in scientific fields where annotation is costly and requires expert knowledge. To address this, open-vocabulary detection methods [9, 10] have emerged, which leverage natural language prompts to detect arbitrary objects without class-specific training. Similarly, Vision-Language Models (VLMs) [11, 12] have extended the reasoning capabilities of large language models [13–15] to the multimodal domain, enabling joint reasoning over text and images. While VLMs are commonly applied to visual question answering, their underlying capacity for logical reasoning makes them suitable for automated verification of computer vision outputs.

**Semantic Segmentation.** Semantic segmentation has traditionally relied on convolutional neural network architectures, such as the foundational U-Net [16]. More recently, the state-of-the-art has shifted towards transformer-based models [17–19]. By leveraging self-attention to capture global context, these models are particularly effective at distinguishing between morphologically similar and adjacent parts, which is a common challenge in medical and biological imaging.

**Machine Learning in Beetle Studies.** Leveraging data from continental-scale beetle sampling programs like NEON [20], recent studies have utilized machine learning to solve problems such as beetle identification and classification. Some works have applied traditional machine learning algorithms to identify beetles based on extracted features [21], while others have explored various deep learning methods, including employing convolutional neural networks for identification [22] and evaluating deep vision models on fine-grained taxonomic classification [23, 24].

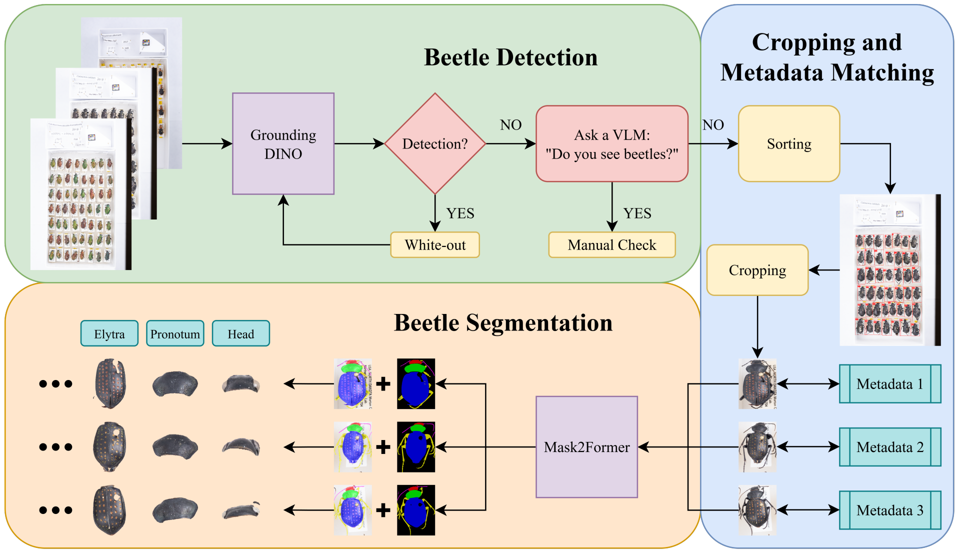


Figure 1: An overview of the 3-stage pipeline: individual detection, cropping/metadata matching, and body-part segmentation stages.

# 3 Beetle Image Processing Pipeline

The input to our pipeline is a series of images of trays containing multiple beetles. Each tray image undergoes a 3-stage processing (shown in Figure 1). This section details the complete workflow by walking through a single image. The code and data are available at `https://anonymous.4open.science/r/BeetleFlow-8BA5`.

## 3.1 Iterative Beetle Detection

The first stage is iterative beetle detection. For each input tray image, an iterative process is initiated. In one iteration, the tray image and the text prompt "a beetle" are sent as input to Grounding DINO, which then outputs the bounding box coordinates for the detected beetles. Next, white masks are placed over all detected beetles based on the bounding box coordinates, leaving the undetected ones in the tray image. The resulting modified image then proceeds to the next iteration for another round of Grounding DINO detection and masking. When no detection is reported by Grounding DINO, the iteration stops. The final modified image is then sent to LLaVA-NeXT with the text prompt "Do you see beetles in this image?". We also constrain the model to output "YES" or "NO" as the final word for automated check. If it answers "YES", a message is reported to the user to do manual detection. If it answers "NO", the detection process is successful and outputs a list of all the bounding box coordinates for the next stage.

## 3.2 Beetle Image Cropping

The second stage takes the bounding box coordinates as input and outputs individual cropped beetle images, with optional functions of sorting and metadata association. Given the bounding box coordinates of beetles, the pipeline crops the beetles out of the original tray image according to the coordinates and saves them as individual images. A specific order can be applied when saving the beetle images. In practice, the beetles in a tray are arranged in regular rows and columns. If digitized metadata for the beetles are available, they are typically provided in a left-to-right, top-to-bottom order corresponding to the arrangement of the beetles in the tray by the biologists. To match each beetle image to its metadata, the default sorting follows this convention. The pipeline saves the beetle images in a left-to-right, top-to-bottom order according to the top-left bounding box coordinate and associates the metadata with each beetle if applicable. The metadata matched to the beetles are outputted to a CSV file per tray.

### 3.3 Fine-grained Beetle Segmentation

The third stage takes individual beetle images as input and outputs the morphological segmentation results for each beetle. For this task, we fine-tune two variants of Mask2Former, which is chosen for its strong performance on specialized tasks with limited training data. Users can select between two levels of granularity: a 5-class model (segmenting head, pronotum, elytra, legs, and antennas) for basic morphological analysis, and a more detailed 9-class model (additionally segmenting eyes, mouthparts, tail, and pin). Both models inherently separate the beetle from the background. For each input beetle image, the model generates a colorized mask image and a beetle image overlaid with the masks. The overlaid beetle image is provided for visualization purposes, allowing for user verification. The mask image can be utilized for two subsequent functionalities of the pipeline, morphological part cropping and defective specimen detection, which are detailed in Appendix B.

## 4 Experiments

### 4.1 Experimental Setup

#### 4.1.1 Beetle Detection

**Datasets.** We apply the detection pipeline on the dataset collected by the National Ecological Observatory Network (NEON) from ecological sites across the U.S., along with associated metadata. The dataset contains 1,506 images of trays containing pinned carabid beetle specimens.

**Evaluation metrics.** Each tray in the NEON dataset is associated with respective metadata, including the ground-truth number of beetles in the tray. After running the detection process on the trays, the number of detected beetles is compared to the ground-truth number. The total detection accuracy over the 1,506 tray images is then calculated to quantify the model's performance.

**Implementation Details.** We utilized the Grounding DINO model with pre-trained weights from the `IDEA-Research/grounding-dino-base` checkpoint. A fixed text prompt, "a beetle.", was used to guide the model in locating specimens within the tray images. For post-processing, we set the box confidence threshold to 0.3 and the text relevance threshold to 0.2. Only bounding boxes with scores exceeding both respective thresholds are retained. To enhance the robustness of the detection, we also introduced a custom filtering step. Any detection box with an area exceeding 5% of the total image area was discarded. This is to remove false detections where the model misidentifies a large portion of the tray as a single beetle. We utilized the LLaVA-NeXT model with pre-trained weights from the `llava-hf/llava-v1.6-mistral-7b-hf` checkpoint for the final verification.

#### 4.1.2 Beetle Segmentation

**Datasets.** The unlabeled individual beetle images are derived from our pipeline by processing the tray images. For 5-class labeling, we label 5 parts for each beetle image: head, pronotum, elytra, legs, and antennas. We manually labeled 160 beetles and utilized an additional 180 labeled beetles from a previous work, SST [25], which follows the same 5-class scheme. A total of 340 labeled beetles were then partitioned into a training set of 272 and a test set of 68 images. For 9-class labeling, we label 9 parts for each beetle image: head, eyes, mouthparts, pronotum, elytra, tail, legs, antennas, and pin. We manually labeled 330 beetles, dividing them into a training set of 264 and a test set of 66 images.

**Evaluation metrics.** We use mean Intersection over Union (mIOU) on the test set as the primary metric, averaged across all classes for each image. In addition to the overall mIOU, we also report the per-class IoU scores to facilitate a more granular analysis.

**Implementation Details.** We fine-tuned two Mask2Former models with a Swin-Large backbone, each initialized with weights from the `facebook/mask2former-swin-large-ade-semantic` checkpoint, which was pre-trained on the ADE20K dataset for semantic segmentation tasks. All input images were resized to a resolution of 512×512 pixels. The models were trained for 30 epochs with a batch size of 10, using the AdamW optimizer and an initial learning rate of 1e-4. All experiments were conducted on two NVIDIA A100 GPUs with 40GB memory each.

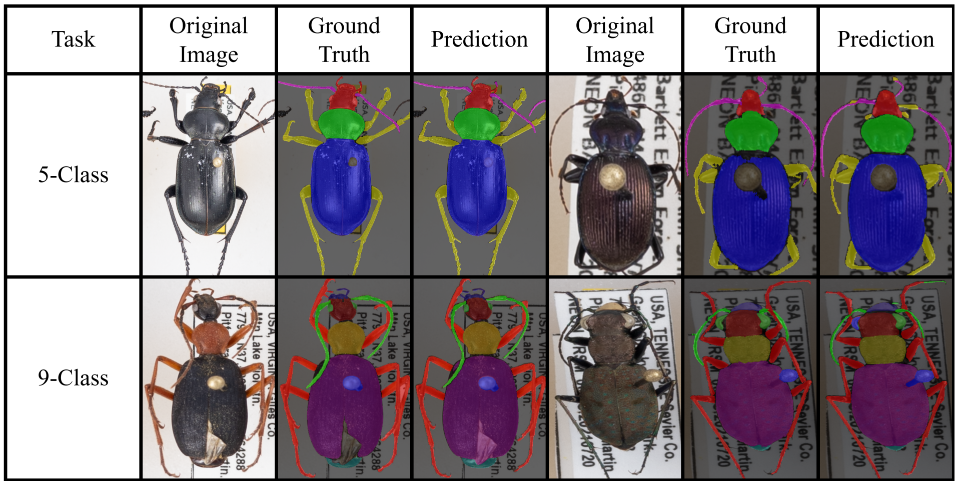


Figure 2: Qualitative segmentation results. Top row: 5-class segmentation (head, pronotum, elytra, legs, antennas). Bottom row: 9-class segmentation (adding eyes, mouthparts, tail, pin). Columns show original, ground truth, and prediction.

### 4.2 Results

**Beetle Detection.** We applied the detection process to 1,506 tray images. Of these, 1,473 trays had a perfect match between the number of detected beetles and the ground truth, yielding a total accuracy of 97.81%. Of the 33 failure cases, 32 had a higher detected beetle count than the ground truth, while only one case had a lower count. A majority of these 32 cases were due to fallen beetle heads on the trays, which the model incorrectly detected as separate beetles. The data indicates that our detection process is highly effective at detecting all beetles on a tray with minimal omissions.

**Beetle Segmentation.** We evaluated the performance of two fine-tuned models on their respective test sets. For 5-class segmentation, the mIOU is 85.11%. For 9-class segmentation, the mIOU is 77.38%. Our per-class IOU results, detailed in Appendix D (Table 1), reveal a notable trend in both segmentation tasks. Take 5-class segmentation as an example: the model achieves high IoU scores on large morphological parts like "pronotum" (91.85%) and "elytra" (94.69%) , while the scores for smaller parts like 'legs' (79.57%) and 'antennas' (65.93%) are comparatively lower. This discrepancy does not solely indicate poor segmentation quality for smaller parts. In fact, our qualitative results (shown in Figure 2) show that these parts are often segmented reasonably well. This phenomenon is partly attributable to the sensitivity of the IoU metric to object size. For small objects, minor deviations of a few pixels can lead to a significant drop in the IoU score.

## 5 Discussion and Future Work

In this work, we develop an automated 3-stage pipeline to process large-scale beetle images. The two deep learning-based processes, iterative Grounding DINO detection and Mask2Former segmentation, have proven to be robust and highly accurate. One limitation of our pipeline is that the accuracy of the detection is influenced if there are split beetles in the tray, e.g., the head of a beetle is fallen off. We have tried methods to automatically recombine the fallen heads with their bodies, but they are not robust as the number and location of the fallen heads are highly variable. In addition, each tray also contains a scale bar and a color table, which can be utilized for beetle measurements and image color calibration. We have developed the functionalities to detect and crop the scale bar and the color table of each tray. The scale bar can be used to automatically measure the morphological statistics of beetles, such as length and area, which can aid biologists in further analysis. The color table can be used for color calibration to ensure the standardization of the beetle images. These applications can be implemented in future work. In summary, our pipeline greatly improves the efficiency of large-scale beetle image processing, yielding useful outputs for various downstream research purposes. This pipeline scheme can be further generalized to other similar organisms beyond beetles, with the potential to improve the data processing workflow in the biology field.

## References

[1] P. M. Hammond, "Species inventory," in *Global Biodiversity: Status of the Earth's Living Resources. A Report Compiled by the World Conservation Monitoring Centre*, B. Groombridge, Ed. London: Chapman and Hall, 1992, pp. 17–39.

[2] S. Liu, Z. Zeng, T. Ren, F. Li, H. Zhang, J. Yang, Q. Jiang, C. Li, J. Yang, H. Su *et al.*, "Grounding dino: Marrying dino with grounded pre-training for open-set object detection," in *European conference on computer vision*. Springer, 2024, pp. 38–55.

[3] H. Liu, C. Li, Y. Li, B. Li, Y. Zhang, S. Shen, and Y. J. Lee, "Llava-next: Improved reasoning, ocr, and world knowledge," January 2024. [Online]. Available: https://llava-vl.github.io/blog/2024-01-30-llava-next/

[4] B. Cheng, I. Misra, A. G. Schwing, A. Kirillov, and R. Girdhar, "Masked-attention mask transformer for universal image segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 1290–1299.

[5] P. Bankhead, M. B. Loughrey, J. A. Fernández, Y. Dombrowski, D. G. McArt, P. D. Dunne, S. McQuaid, R. T. Gray, L. J. Murray, H. G. Coleman *et al.*, "Qupath: Open source software for digital pathology image analysis," *Scientific reports*, vol. 7, no. 1, pp. 1–7, 2017.

[6] M. A. Gehan, N. Fahlgren, A. Abbasi, J. C. Berry, S. T. Callen, L. Chavez, A. N. Doust, M. J. Feldman, K. B. Gilbert, J. G. Hodge *et al.*, "Plantcv v2: Image analysis software for high-throughput plant phenotyping," *PeerJ*, vol. 5, p. e4088, 2017.

[7] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[8] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[9] L. H. Li, P. Zhang, H. Zhang, J. Yang, C. Li, Y. Zhong, L. Wang, L. Yuan, L. Zhang, J.-N. Hwang *et al.*, "Grounded language-image pre-training," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 10 965–10 975.

[10] X. Gu, T.-Y. Lin, W. Kuo, and Y. Cui, "Open-vocabulary object detection via vision and language knowledge distillation," *arXiv preprint arXiv:2104.13921*, 2021.

[11] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*. PmLR, 2021, pp. 8748–8763.

[12] J. Li, D. Li, S. Savarese, and S. Hoi, "Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models," in *International conference on machine learning*. PMLR, 2023, pp. 19 730–19 742.

[13] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat *et al.*, "Gpt-4 technical report," *arXiv preprint arXiv:2303.08774*, 2023.

[14] A. Grattafiori, A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Vaughan *et al.*, "The llama 3 herd of models," *arXiv preprint arXiv:2407.21783*, 2024.

[15] G. Team, P. Georgiev, V. I. Lei, R. Burnell, L. Bai, A. Gulati, G. Tanzer, D. Vincent, Z. Pan, S. Wang *et al.*, "Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context," *arXiv preprint arXiv:2403.05530*, 2024.

[16] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[17] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[18] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[19] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 10 012–10 022.

[20] D. Hoekman, K. E. LeVan, C. Gibson, G. E. Ball, R. A. Browne, R. L. Davidson, T. L. Erwin, C. B. Knisley, J. R. LaBonte, J. Lundgren *et al.*, "Design for ground beetle abundance and diversity sampling within the national ecological observatory network," *Ecosphere*, vol. 8, no. 4, p. e01744, 2017.

[21] J. Blair, M. D. Weiser, M. Kaspari, M. Miller, C. Siler, and K. E. Marshall, "Robust and simplified machine learning identification of pitfall trap-collected ground beetles at the continental scale," *Ecology and evolution*, vol. 10, no. 23, pp. 13 143–13 153, 2020.

[22] L. Wu, Z. Liu, T. Bera, H. Ding, D. A. Langley, A. Jenkins-Barnes, C. Furlanello, V. Maggio, W. Tong, and J. Xu, "A deep learning model to recognize food contaminating beetle species based on elytra fragments," *Computers and Electronics in Agriculture*, vol. 166, p. 105002, 2019.

[23] S. Rayeed, A. East, S. Stevens, S. Record, and C. Stewart, "Fine-grained taxonomy with vision models: A benchmark on long-tailed and domain-adaptive classification," 2025.

[24] S. Rayeed, A. East, S. Stevens, S. Record, and C. V. Stewart, "Beetleverse: A study on taxonomic classification of ground beetles," *arXiv preprint arXiv:2504.13393*, 2025.

[25] Z. Feng, Z. Wang, S. I. Bueno, T. Frelek, A. Ramesh, J. Bai, L. Wang, Z. Huang, J. Gu, J. Yoo, T.-Y. Pan, A. Chowdhury, M. Ramirez, E. G. Campolongo, M. J. Thompson, C. G. Lawrence, S. Record, N. Rosser, A. Karpatne, D. Rubenstein, H. Lapp, C. V. Stewart, T. Berger-Wolf, Y. Su, and W.-L. Chao, "Static segmentation by tracking: A frustratingly label-efficient approach to fine-grained segmentation," 2025. [Online]. Available: https://arxiv.org/abs/2501.06749

[26] J. Gu, S. Stevens, E. G. Campolongo, M. J. Thompson, N. Zhang, J. Wu, A. Kopanev, Z. Mai, A. E. White, J. Balhoff, W. M. Dahdul, D. Rubenstein, H. Lapp, T. Berger-Wolf, W.-L. Chao, and Y. Su, "BioCLIP 2: Emergent properties from scaling hierarchical contrastive learning," 2025. [Online]. Available: https://arxiv.org/abs/2505.23883

# BeetleFlow: A 3-Stage Deep Learning Pipeline for Beetle Image Processing

# Appendix

## A Potential Improvement on Iterative Detection

We also propose a potential improvement to increase the accuracy of iterative detection, shown in Figure 3. In the new scheme, we add an iterative decrease in the box confidence threshold and text relevance threshold of Grounding DINO. As only bounding boxes with scores exceeding both respective thresholds are retained, the decrease enables more potential specimens to be detected. This improvement is for cases when Grounding DINO reports no detection but there are still specimens left in the image. The iterative decrease is performed until a predefined minimum threshold is reached. For our detection task on tray images of beetles, Grounding DINO already performs well without the iterative decrease of thresholds, therefore we do not apply it to our beetle image processing pipeline. We leave it as an implementation suggestion for other datasets when the default detection process cannot detect all the target objects in the image.

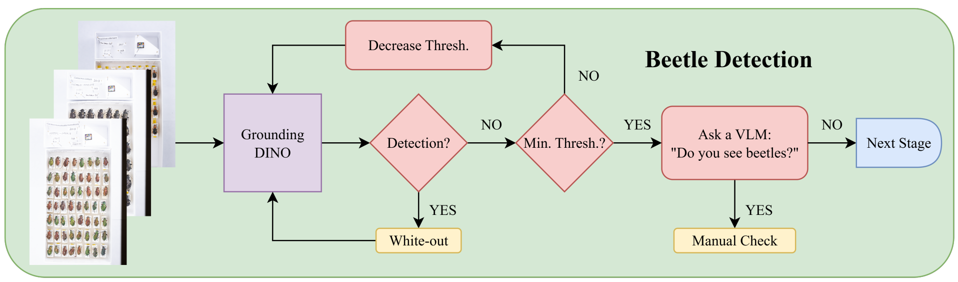


Figure 3: Improved Iterative Detection Process.

## B Implementation Details for Segmentation

### B.1 Pins on Beetle Specimens

One thing worth mentioning is that "pin" is included as an individual class in the 9-class segmentation. Since pinning beetles on trays is a common practice for biologists collecting beetles and thus pins are unavoidably included in the beetle images, including "pin" as a class allows the model to explicitly learn and identify this common non-biological artifact, preventing its incorrect classification as beetle body. By accurately segmenting the pin, it can also be conveniently excluded in further research. In the 5-class segmentation, the pin is not included in any class.

### B.2 Subsequent Functionalities after Segmentation

The mask images generated by the segmentation process can be utilized for two subsequent functionalities of the pipeline. The first functionality is morphological part cropping. Based on the mask image, users can select one or more parts of a beetle to crop and save as a separate image for downstream research. The second functionality is defective specimen detection. For each mask image, the pipeline performs two checks. Firstly, it detects for any missing classes. The absence of a class indicates that the corresponding part of the beetle is missing. Secondly, it compares the area of each class to the average area of that class across all mask images on a per-tray basis. A significant difference suggests that the corresponding part of the beetle is incomplete. The information for all these defective specimens is recorded in a separate file for user inspection and optional removal. This functionality can check the integrity of a large number of beetles automatically, which is useful for further studies that require high-quality specimens such as deep learning tasks.

# C Dataset Details for Segmentation

The total number of individual beetle images derived from the detection process is 51,554, covering 184 species. For 5-class labeling, we manually labeled 160 beetles, comprising 80 beetle species with two individuals selected from each species. The 180 labeled beetles utilized from SST comprise 12 species with 15 individuals from each species. For 9-class labeling, we manually labeled 330 beetles, of which 160 are from the same images for 5-class labeling. The remaining 170 individuals were selected from the 30 most distinctive species among the 184 species based on BioCLIP 2 [26] embeddings, with 5-6 individuals from each species. This adds more diversity to the training beetles, enabling the model to achieve better performance when segmenting a wider variety of beetles.

# D Per-Class IoU Results for Segmentation

In addition to the overall mIOUs, we also report the per-class IoUs for each segmentation task, shown in Table 1. Our fine-tuned models perform very well in segmenting large morphological parts such as the "pronotum" and "elytra", while have a lower performance on more fine-grained parts such as "legs" and "antennas". Despite the relatively low scores, qualitative results show good segmentation qualities on these fine-grained parts. One thing we observed is that "tail" has the lowest IOU because the tail is not visible in many beetle images, so the model cannot learn good segmentation features for it. Therefore, if provided with more high-quality labeled data, the Mask2Former is expected to yield better performance.

Table 1: Per-class IoUs for 5-class and 9-class segmentation on respective test sets.

| **Category** | **5-class IoU (%)** | **9-class IoU (%)** |
|---|---|---|
| Head | 83.64 | 83.09 |
| Pronotum | 91.85 | 90.99 |
| Elytra | 94.69 | 93.97 |
| Legs | 79.57 | 85.39 |
| Antennas | 65.93 | 70.08 |
| Eyes | – | 68.43 |
| Mouthparts | – | 60.08 |
| Tail | – | 53.49 |
| Pin (Artifact) | – | 72.13 |

# NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We have accurately stated our contributions and scope in the Abstract and the Introduction sections.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We have discussed the limitations of our work in the Discussion and Future Work section.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: This paper does not include theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We have disclosed our implementation details and experimental settings in the Beetle Image Processing Pipeline and the Experiments sections, as well as in the Appendix.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We have provided the code and the labeled dataset with instructions in the anonymized GitHub repository linked in the Beetle Image Processing Pipeline section.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: We have specified all the training and test details in the Experiments section and the Appendix.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: For our experiments on detection over 1,506 trays and segmentation over the test sets, the tray-level accuracy and the mIOU are always the same.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We have provided information on the computer resources in the Experiments section.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Our research conforms with the NeurIPS Code of Ethics in every respect.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We have discussed potential positive societal impacts in the Introduction and the Discussion and Future Work sections, and this work does not have negative societal impacts.

    Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: This paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We have cited all the models and datasets used in this paper.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We have provided our labeled dataset along with the documentation in the anonymized GitHub repository linked in the Beetle Image Processing Pipeline section.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: The core method development in this research does not involve LLMs.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.