Save the Whales: Body Keypoints Detection for Automated Annotation

Suhaas Kiran DG
University of Massachusetts
Amherst, MA
sdg@umass.edu

Siddarth Suresh
University of Massachusetts
Amherst, MA
siddarthsure@umass.edu

Kedarnath Chimmad
University of Massachusetts
Amherst, MA
kchimmad@umass.edu

Abstract

In the realm of wildlife research, aerial photogrammetry emerges as a non-intrusive means to gather data on wild animals. However, the vast volume of data generated necessitates efficient processing methods. Specifically, annotating key points on whale bodies for metrics like body mass proves to be time-consuming. To address this, we propose a keypoint detection model based on the segmentation backbone and regression head. Using computer vision techniques, our proposed solution aims to reduce the annotation time from hours to mere minutes. However using a direct keypoint detection strategy based on segmentation presents certain issues and does not converge. We test this method, analyze the results and propose methods to overcome the problems.

1. Introduction

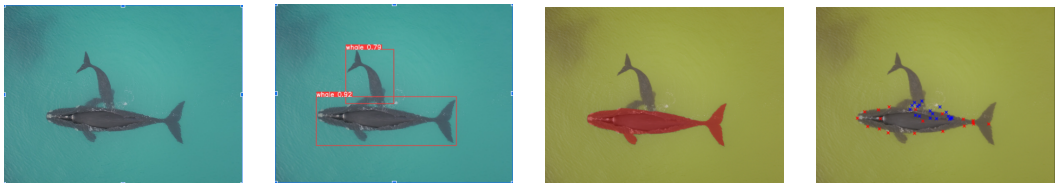
Estimating the body conditions of whales holds significant importance for marine biologists and conservationists, facilitating assessments of health, movement patterns, and responses to environmental factors. Traditional methods like whaling and direct inspection are not only risky but also costly. However, the emergence of drone technology enables the non-invasive collection of vast aerial imagery, providing valuable data for analysis. Currently, annotating these images manually as shown in Figure 2 consumes considerable time. While existing solutions like Mask-RCNN [2] offer automation, their dependency on GPU inference presents practical challenges, particularly in remote field settings where GPU-enabled devices are impractical. Addressing this, we propose a novel approach to enable CPU inference without sacrificing accuracy, leveraging models optimized for CPU execution. This innovation promises to streamline annotation processes, empowering marine biologists with efficient tools for studying whale populations and aiding in conservation efforts.

2. Related Works

The current landscape of whale body mass estimation requires addressing the need for low-compute solutions tailored to the constraints of marine biologist fieldwork. YOLO[9], renowned for its low-compute efficiency, emerges as a promising candidate for real-time on-spot detection, as evidenced in recent studies on marine life object detection[14] [3] [4] .

Presently, machine learning-based approaches[2] for body condition estimation typically involve two models: image segmentation for identifying body contours and keypoint detection for locating the body axis. Combined results yield width values used by biologists for estimating body condition parameters. Notably, existing solutions achieve high accuracy, with segmentation mask accuracy at 96% and axis detection at 93%. However, training these models is hindered by the laborious manual annotation process. For instance, in this study on Southern right whale species involved manual annotation of 300 images containing 468 whale instances, underscoring the time-consuming nature of data preparation. While the current inference model utilizes Detectron2[5], leveraging the Mask-RCNN model, its dependency on GPU at inference poses practical challenges in field settings. To mitigate this, transitioning to YOLO[14] offers a promising alternative.

Furthermore, insights from human pose detection research suggest that regression-based [11] approaches offer faster processing, albeit with slight compromises in accuracy and mean Average Precision (mAP) as compared to one-hot mask[5] and heatmap [8] approaches. Leveraging these insights, the study explores a detection-localized segmentation method followed by a regression head for keypoint detection. For detection and bounding box drawing, YOLO[9] serves as a robust choice, while SOTA segmentation methods like Segment Anything (SAM)[6] by Meta showcase adaptability and superior performance in diverse domains of computer vision and image processing, promising advancements in whale body mass estimation techniques. For localized segmentation with a minimal amount of data, UNet[10] has been proven as a good method. Wu



(a) Input Image (b) Bounding Box by Yolo (c) SAM Segmentation Mask (d) Keypoints

Figure 1. Output at each step. In (d) ground truth is in red and model's output in blue

et al.[13] used a UNET architecture to detect and segment very small objects[10] [7], suggesting its efficiency in segmenting small local objects with greater accuracy, including marine life[12].

3. Dataset

The dataset comprises approximately 12,000 unlabelled raw whale images depicting four distinct species. These images exhibit resolutions of either (2592 x 4608) or (3456 x 4608) pixels. Among these, 400 images featuring Southern right whales have been meticulously annotated. A representative image from the dataset is presented in Figure 2. In approximately 90% of the images, a pair of whales is discernible, typically comprising a larger maternal whale accompanied by a smaller calf. Single whale occurrences constitute around 4% of the dataset, while the remaining images portray more than two whales.

The ground truth annotations consist of 38 coordinate values, encompassing (x, y) coordinates for each of the 19 keypoints characterizing the whale's anatomy (refer to Figure 3). In instances where multiple whales are present in an image, not all whales are annotated, primarily due to certain keypoints being obscured from view for some individuals.

4. Methodology

4.1. Data Preprocessing

For the training of the YOLOv8 instance detector tailored for whale detection, our approach comprised several key steps. Initially, bounding boxes were manually drawn around whale instances within 400 images using the CVAT online annotation tool. Subsequently, these annotations were converted into YOLO format labels to facilitate training.

Given the absence of keypoint annotations for all whale instances within an image, we devised a refinement process to enhance the dataset's quality. This involved the removal of SAM segmentations for instances lacking keypoint annotations, employing a technique known as *closest distance matching*. The following steps outline this process:

1. Find the center points of all the bounding boxes.
2. Get the center of the body of the whales from the annotations.

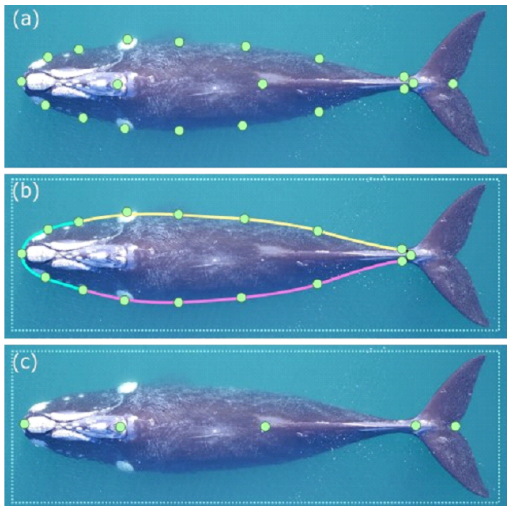


Figure 2. : (a) Point marked by annotators on each whale, (b) 3 cubic splines are fitted on the annotated points to serve as input to the segmentation model that finds the body contour, (c) 5 points along the axis that are used as input to the keypoint detection model to find the body axis. [2]

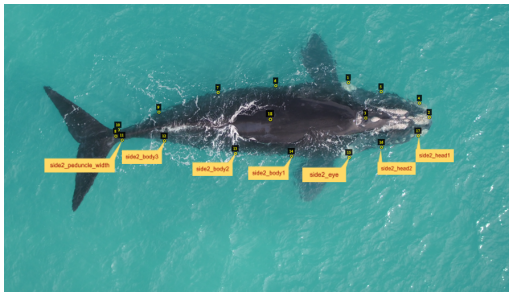


Figure 3. Manual Keypoints Annotations

3. For every annotation, (x_1, y_1), find the bounding box center (x_2, y_2) with the minimum L_2 distance from the (x_1, y_1).
4. To account for false positive detections by the YOLOv8 model, only match the pair if the point (x_1, y_1) lies inside the bounding box corresponding to (x_2, y_2).

4.2. Instance Detection

In consideration of real-time testing requirements and the absence of universal keypoint labels, we bifurcated the training approach into instance detection and keypoint detection phases. Instance detection, accomplished through training a YOLOv8 model separately, aimed at detecting all whale instances within an image during real-time testing. As keypoint labels were unavailable for every whale instance, image labeling was performed using an online annotator, facilitating model training.

4.3. SAM Segmentation

Furthermore, we leveraged the YOLOv8 detector's bounding box coordinates to fine-tune SAM-generated segmentation masks. This refinement process involved extracting the most confident mask and we excluded masks for the instances lacking keypoint annotations. Notably, this mask removal step was confined to the training phase and not intended for real-time application.

4.4. UNet Based Keypoint Detection

A segmentation-based model was used for the keypoint detection because identifying pixel level features was required for identifying the keypoints and a segmentaion-based model could create segmentations localized around the corresponding keypoints. We created a UNet architecture with four convolution layers of filters (32,64,128,256), a strided convolution layer with 512 filters and then corresponding de-convolution layers in the decoder step. A sigmoid layer was added after the decoder step. This UNet backbone produced an output of the same 2D dimensions as that of the input image. To obtain keypoint regressions from the this output, we added an MLP layer of layer sizes (1024,256,38) to output 38 keypoint values.
For the input of the UNet model, we used the raw whale image and concatenated the segmentation mask as the fourth channel, thus created an input of dimensions $(4, H, W)$. The segmentation mask was added to provide context and the boundary of the whale as an additional information to help in localization. So if an image had two annotations for two distinct labels, then by adding masks for different whales separately to that image, we created two different training inputs, one for each annotated whale. All images were of very high resolution and did not have the same dimensions, thus making it difficult to load them into memory and pass as an input to the model. Thus we zero-padded the smaller images to match the sizes to $(3456, 4608)$ and then resized them using bilinear downsampling to one-eighth of their original size $(432, 576)$.

The overall architecture is shown in Figure 4.

4.5. Loss

For the loss, we used **Mean Squared Error** measured between the predicted coordinate values and the actual coordinate values obtained from the annotations.

5. Experiments

5.1. YOLO Detection

We initially evaluated instance detection using the YOLOv8 model with pretrained weights. However, the model frequently failed to identify the tail fin of the whale as a part of the body and occasionally labeled the tail as a separate instance. To address this, we manually created box labels and custom-trained the YOLOv8 model using 350 examples, allocating 300 for training and 50 for validation. Training occurred over 100 epochs with default learning parameters. YOLOv8 automatically logged evaluation data for both training and validation examples.

Subsequently, we conducted qualitative testing on 50 additional test examples. Through fine-tuning, the model exhibited significant improvement, producing tight and precise bounding boxes around whales with high confidence levels.

5.2. SAM Segmentation

We did not train the SAM model explicitly on our data since we did not have segmentation annotations. However, we assisted the SAM model with the bounding box coordinates received from the previous stage to localize the segmentations. With the box prompts, the model could generate precise masks which we evaluatedwith the qualitative analysis of random samples. The segmentations of the downsampled images were also manually observed to check for any distortions. The masks were still significantly precise, however there were color distortions.

5.3. UNet based Keypoint Detection

Training
The SAM generated masks were added as the fourth channel to the images and then downsampled. We divided the data into train, validation and test sets in the split ratio of 76:9:15 respectively.

Even with the downsampled images, we could not store the entire training set into the memory at a time. Hence we divided the training set into six sets and successively trained them for 70 epochs each. After training one set, we saved the parameters and used them for the next set, thus not training every set from scratch. Initially we used an AdamW optimizer, with a leraning rate of 0.001 and weight decay of 0.00001. In the initial stages of training, the training loss and testing loss decreased gradually. However eventually the drop of validation loss started reducing, indicating

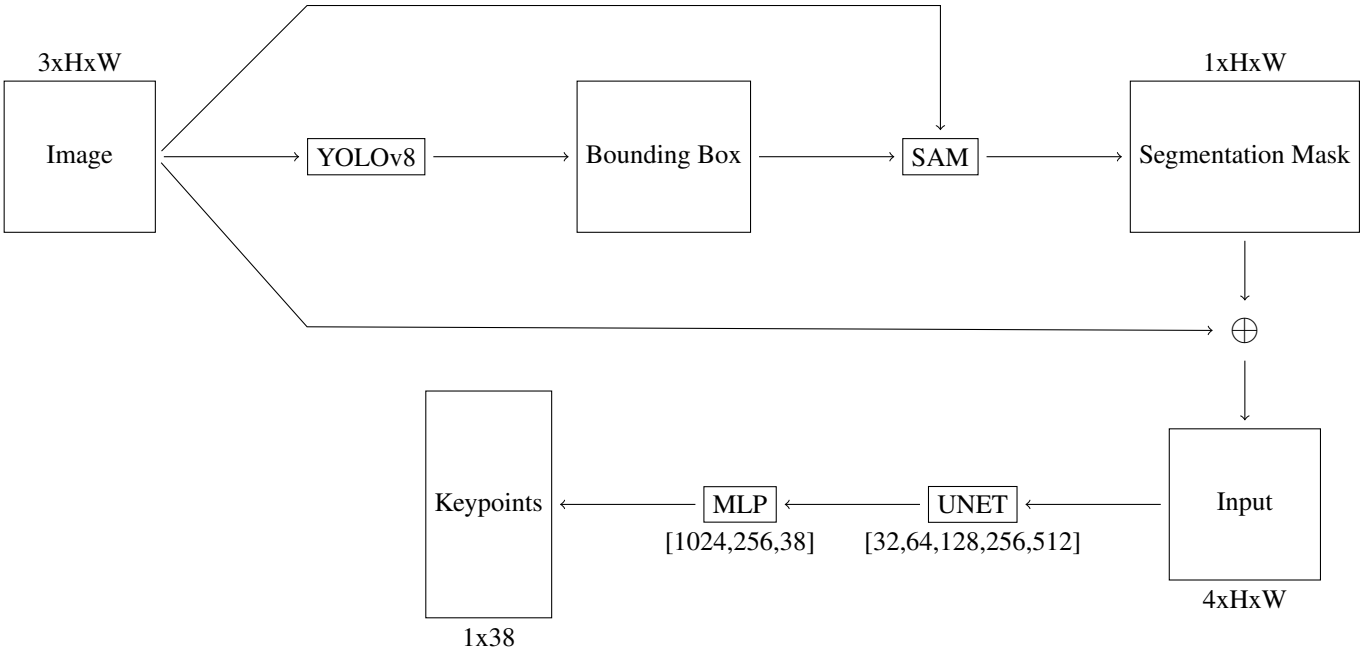


Figure 4. Architecture

overfitting (refer Figure 5). Thus in later stages, we experimented with decreasing the learning rates upto 0.000005 and increasing the weight decay upto 0.001. However there was no visible improvement and the validation loss stagnated in all scenarios.

We also tested the training by reducing the downsampling ratio from 8 to 4. There were no noticeable changes in the learning curve pattern, thus discarding the possibility of downsampling causing the poor predictions.

Keypoints Evaluation

On an average, each keypoint was about 100 pixels off from the actual values. We also visually compared the positions of predicted keypoints and actual locations. Even though the predicted keypoints were aggregated over the whale, which probably happened because of the segmentation mask provided as input, there was no recognizable pattern. This indicated towards any lack of feature level understanding of the keypoints. In some images, the shapes formed by the keypoints resembled the shape of a whale, however such instances were rare and resulted from overfitting.

UNet Block Outputs

We also examined the outputs generated by the UNet block before the MLP layer. Using the MSE loss with respect to the keypoints, we expected the concentration of distinct segmentations around the keypoints. Contrary to our expectations, the generated masks consisted segmentations over the entire body of the whale (refer Figure 8 Column 4). These masks were similar to the ones created by the SAM model which we had concatenated in the input. Thus, the intended effect of UNet was not observed and we could have obtained similar results by directly using SAM masks with the MLP layer.

YOLOv8 Pretrained Detection

We tested the keypoint detection using off-the-shelf YOLOv8 keypoint detector. We trained the model for 75 epochs with the same train and test split that we used for YOLOv8 instance detection. Though we saw incredible results previously when training the bounding box prediction individually, the results for the keypoint detection were bad (refer Figure 9) . Even the bounding box results were not good and whale instances were missed in many examples. This happened because of the lack of consistencies in the annotations available for all whales present in an image. As a result, the whale detection deteriorated.

6. Conclusion

Due to the limitations with available annotations, we developed a model for keypoint detection with separate instance detector, segmentation mask generator and keypoint detector. We used a regression head on the top of the segmentation-based UNet backbone. We expected the segmentation backbone to create localized patches around the keypoints and then intended to use them to infer the coordinates. However the UNet created full whale segmentations, similar to ones created by the SAM, thus deeming the addi-

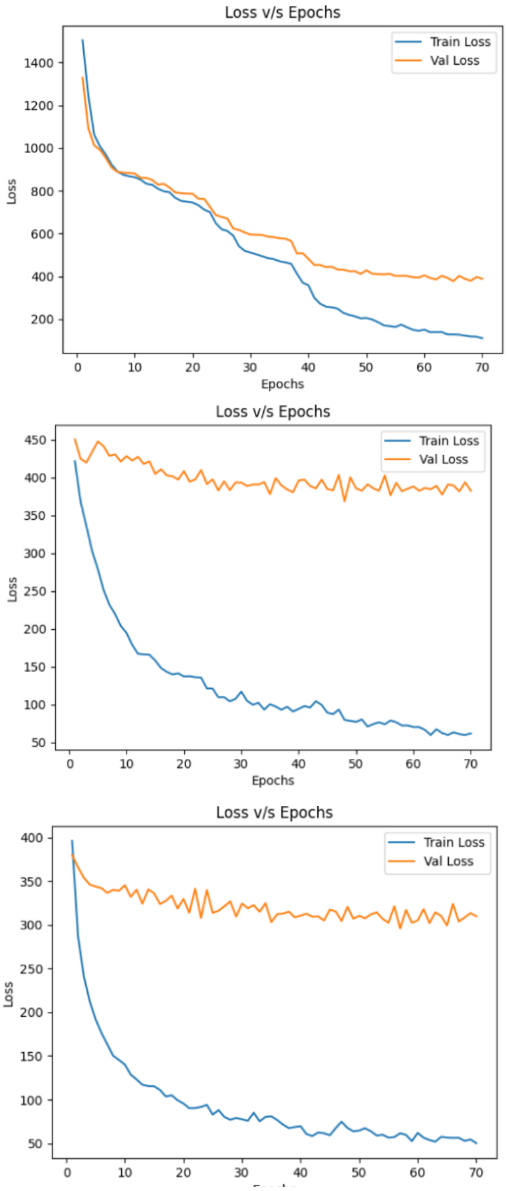


Figure 5. Training Loss and Validation Loss v/s Epochs for UNet based Keypoint Detection

tion of the UNet block redundant.

7. Future Works

For creating a top-down keypoint detection model based on the segmentation, we need to develop a model which is explicitly trained to generate loacalized patches in form of heatmaps, instead of training them to predict very specific pixel coordinates. Segmentation model cannot converge to single pixel level prediction. In such methods, we generate a heatmap which is basically a gaussian mask with very low standard deviation and centered at the keypoints, so

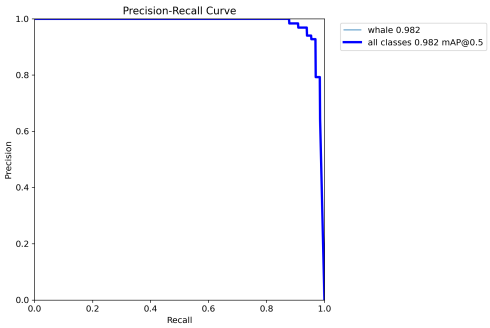


Figure 6. PR Curve for YOLOv8 Instance Detection

it becomes a small patch. Then we train the segmentation model to predict those patches, instead of a single pixel keypoint. Some non-maximal suppression strategy can be used to infer keypoints from those concentrated patches. We will also explore methods for network training refinement and post processing [1] to improve the predictions. Data augmentations can also be used to increase the dataset size and variations.

Finally, we will build a human-in-the-loop model to rectify the outputs using human supervision. We will determine ways to perform active learning sampling using uncertainty and diversity sampling strategies.

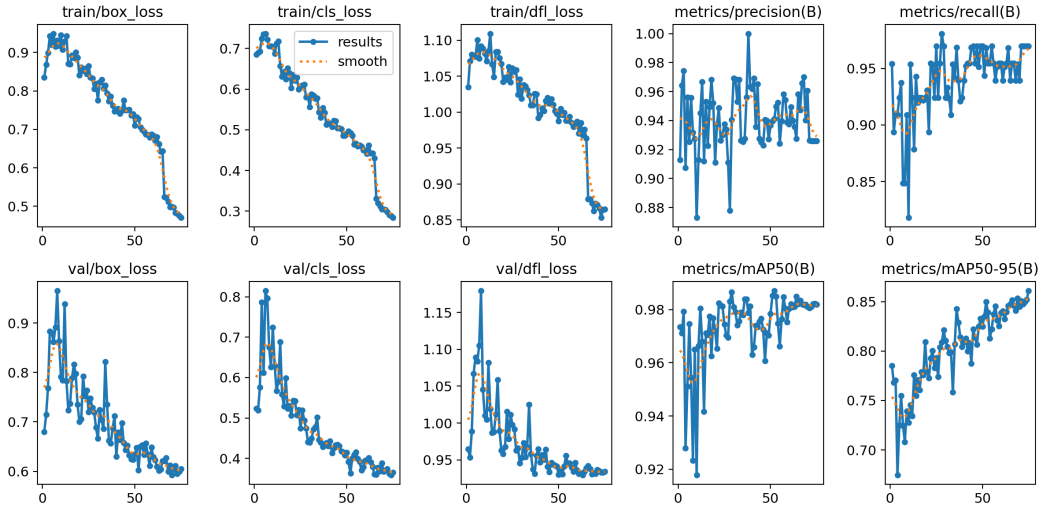


Figure 7. Train and Validation loss v/s Epochs and Mean Average Precision Scores for YOLOv8 Instance Detection

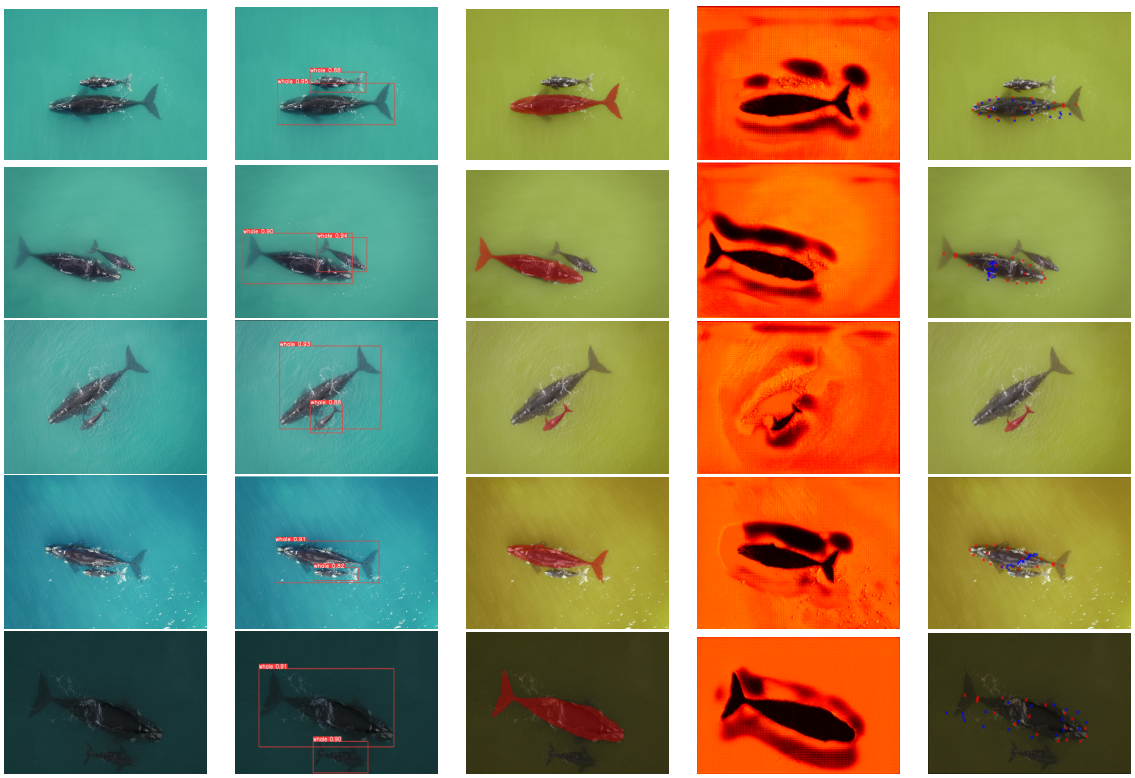


Figure 8. Keypoint Detection Results. (Order: Input-> YOLO-> SAM-> UNET-> MLP)

Figure 9. Keypoints Detection with YOLOv8 Keypoints Detector

References

[1] Haoming Chen, Runyang Feng, Sifan Wu, Hao Xu, Fengcheng Zhou, and Zhenguang Liu. 2d human pose estimation: A survey, 2022. 5

[2] Duncan Irschick Fredrik Christiansen Chhandak Bagchi, Josh Medina. An automated tool to extract body morphometric data from drone aerial photographs of southern right whales to measure body condition. Technical report, University of Massachusetts and Aarhus University, 2023. 1, 2

[3] Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, and Jian Sun. YOLOX: exceeding YOLO series in 2021. *CoRR*, abs/2107.08430, 2021. 1

[4] An Guo, Kaiqiong Sun, and Ziyi Zhang. A lightweight yolov8 integrating fasternet for real-time underwater object detection. *Journal of Real-Time Image Processing*, 21, 2024. 1

[5] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn, 2018. 1

[6] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything, 2023. 1

[7] Martin Kolarík, Radim Burget, Václav Uher, Kamil Říha, and Malay Kishore Dutta. Optimized high resolution 3d dense-u-net network for brain and spine segmentation. *Applied Sciences*, 2019. 2

[8] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. *CoRR*, abs/1603.06937, 2016. 1

[9] Joseph Redmon, Santosh Kumar Divvala, Ross B. Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. *CoRR*, abs/1506.02640, 2015. 1

[10] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *CoRR*, abs/1505.04597, 2015. 1, 2

[11] Xiao Sun, Bin Xiao, Shuang Liang, and Yichen Wei. Integral human pose regression. *CoRR*, abs/1711.08229, 2017. 1

[12] Lidiya Thampi, Rosemol Thomas, Suraj Kamal, Arun A. Balakrishnan, T. P. Mithun Haridas, and M. H. Supriya. Analysis of u-net based image segmentation model on underwater images of different species of fishes. In *2021 International Symposium on Ocean Technology (SYMPOL)*, pages 1–5, 2021. 2

[13] Xin Wu, Danfeng Hong, and Jocelyn Chanussot. Uiu-net: U-net in u-net for infrared small object detection. *IEEE Transactions on Image Processing*, 32:364–376, 2023. 2

[14] J. Zhong, M. Li, J. Qin, Y. Cui, K. Yang, and H. Zhang. Real-time marine animal detection using yolo-based deep learning networks in the coral reef ecosystem. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLVI-3/W1-2022:301–306, 2022. 1