Balancing Human and Machine Performance When Analyzing Image Cover

Travis Mandel
Computer Science
University of Hawaii at Hilo
Hilo, HI, USA
tmandel@hawaii.edu

Drew Gotshalk
Computer Science
University of Hawaii at Hilo
Hilo, HI, USA
drewtg@hawaii.edu

Nicholas A. V. Del Moral
Computer Science
University of Hawaii at Hilo
Hilo, HI, USA
delmoral@hawaii.edu

Danielle K. Wilde
Marine Science
University of Hawaii at Hilo
Hilo, HI, USA
dwilde@hawaii.edu

Micah J. Marshall
Mathematics
University of Hawaii at Hilo
Hilo, HI, USA
micahjm@hawaii.edu

Alexander J. Spengler
Marine Science
University of Hawaii at Hilo
Hilo, HI, USA
spengler@hawaii.edu

Shane J. Murphy
Marine Science
University of Hawaii at Hilo
Hilo, HI, USA
sjmurphy@hawaii.edu

Brittany D. Wells
Marine Science
University of Hawaii at Hilo
Hilo, HI, USA
bwells4@hawaii.edu

John H. R. Burns
Marine Science
University of Hawaii at Hilo
Hilo, HI, USA
johnhr@hawaii.edu

Grady Weyenberg
Mathematics
University of Hawaii at Hilo
Hilo, HI, USA
gradysw@hawaii.edu

***Abstract*—In computer vision, it is often unclear how humans should annotate data in a way that optimizes both human effort and machine performance. This is particularly true in image cover applications, in which the goal is to determine what percentage of the image is covered by a certain class of objects (e.g. trees or grass). In this paper, we present the first study comparing the efficacy of fundamentally different annotation styles when applied to image cover problems. Our results on two real-world marine science problems indicate that the most accurate human annotations do not necessarily result in the best machine learning performance, and that by carefully selecting the format of the annotations one can achieve good performance even with limited human effort.**

***Index Terms*—computer vision, machine learning, human-computer interaction, marine science, image segmentation**

I. Introduction

In recent years, deep learning approaches to computer vision have achieved a number of notable successes [1], [2]. However, in order to successfully train these systems to perform a novel vision task, it is necessary to supply a large amount of supervision in the form of annotated images. Although techniques such as pretraining (e.g. on ImageNet [3]) have enabled deep learning to work well with less training data, the performance of deep learning pipelines remains highly dependent on the quality and quantity of the provided annotations. Practitioners working on a novel computer vision problem likely have limited resources to devote to annotating data, and thus face the problem of determining how to specify the annotation format, for example: bounding boxes or image labels?

Funding provided by NSF awards OIA-1557349, EPS-0903833

In certain cases, the form of these annotations is quite clear: For instance, if the goal is simply to detect whether there is a storefront in an image, a simple binary label will suffice. However, in other more complicated scenarios, there is a wide range of possible types of annotations, each requiring a different level of human effort and necessitating different machine learning (ML) techniques. An important example of this are "image cover" problems, in which the goal is to report the proportion of the images surface area that is covered by a certain object of interest. This problem occurs in many different domains: A surveyor might want to know what percentage of land is covered by trees [4], or a doctor might want to know what percentage of a patient's skin is covered by wounds [5].

When computer vision practitioners wish to build systems to address a new image cover problem, they face a variety of challenges in determining how their data should be annotated. Carefully segmenting the images would seem to provide a highly accurate estimate, but carefully tracing every boundary requires a large amount of human effort. On the other hand, humans could simply look at the image and visually estimate the amount of cover, which requires much less effort but is likely more inaccurate. These tradeoffs are complicated further by the fact that it is unclear how effectively the deep learning system will be able to process the annotations, especially for new or understudied problems. Most of the computer vision literature assumes a fixed annotation format, and offers little guidance on how to choose the type of annotations to best fit the task, especially in an image cover setting. Therefore,

978-1-7281-5584-5/19/$31.00 ©2019 IEEE
DOI 10.1109/CSCI49370.2019.00125

practitioners risk selecting an annotation type that results in expending a large amount of annotator effort for very little return in terms of machine learning performance.

In this paper, we take the first steps toward addressing this important problem, by comparing how various styles of annotations trade off human effort and machine performance in image cover settings. Specifically, we examine three types of annotations: bounding boxes, segment-level cover percentages, and image-level cover percentages. We build pipelines that train deep learning systems on these annotations and convert their output to an overall image-level cover percentage. We perform two case studies, analyzing live coral cover and coral disease severity, which are related to the important real-world problem of assessing the health status of coral reefs. We compare human and machine performance in these domains, finding surprising results which indicate that choosing the right form of annotation can save substantially on human time and effort while boosting ML performance.

II. Related Work

Computer vision It has been widely recognized that it is time-consuming to manually perform complex tasks such as image segmentation and object detection [6], [7], and various strategies have been used to alleviate this, such as inferring more complex annotations from simpler ones [6], or directing human effort towards annotating more useful regions [8]. Although some work examines how to optimize the annotation process; for example, through better UIs [9] or through AI assistance [10], very few studies directly compare fundamentally different types of annotations.

Closely related work in this space by Dutt Jain et al. [11], studies different forms of annotations when the goal is to fully segement an image. Specifically, they predict the difficulty of annotating an image with either a bounding box, a loose outline, or a tight outline, and use this to determine how to select the complexity of the annotations to optimize human time. Dutt Jain et al. studies when to simplify image segmentation annotations, but we consider the opposite scenario: the end goal is very simple (a single image-level percentage), but is typically thought to be better specified through more complicated annotations. Additionally, Dutt Jain et al. assumes that the types of annotations that result in good human accuracy are the types of annotations that are best to use to train machine learning systems; as we show in our experiments, this is not always the case.

Image cover Image cover problems have been studied using a number of different annotation formats. For instance, recognizing tree-covered areas in images has often used bounding box annotations [2], calculating wound surface area has used segmentation polygons [5], turfgrass cover has used image-level cover estimates [12], coral cover has asked users to individually classify large amounts of randomly chosen points [10], while benthic habitat cover has had users segment the image in full detail [13]. Although the number of types of annotations used to specify cover is large, we are not aware of any single work that has explicitly compared different annotation styles in this space.

III. Problem Setup

We consider image cover problems, in which the goal is to report information about the relative area of certain objects in an image. Specifically, we consider two variants:

In the first type of cover problem (**single-layer cover**), the goal is simply to report the overall proportion of the image which contains the desired object. In other words, the cover can be calculated as: $C = \frac{\sum_{p \in \mathcal{P}} y_p}{|\mathcal{P}|}$ where $\mathcal{P}$ is the set of all pixels in the image, and $y_p \in \{0, 1\}$ indicates whether a pixel belongs to the class of interest. For instance, in a tree cover problem y_p would be 1 if and only if the pixel is a tree.

Although useful, in certain cases simply determining the fraction of the entire image is undesirable. Instead, one wants to determine the fraction of some background class that is covered by some object of interest. For example, a doctor likely wants to know only the percentage of a patient's skin that is wounded, rather than the percentage of the entire image (which may contain numerous non-skin pixels) . We call this problem **two-layer cover** for short. This hierarchical structure can be represented by the following equation: $C = \frac{\sum_{p \in \mathcal{P}} t_p y_p}{\sum_{p \in \mathcal{P}} t_p}$ where $t_p \in \{0, 1\}$ indicates whether a pixel belongs to a desired "background" class. In the aforementioned wound cover problem, t_p would be 1 if and only if the class pixel represented skin, while y_p would be 1 if and only if the pixel represented a wound.

The end goal in these scenarios is simply to report the cover metric C. As such, a natural performance metric is to compute the mean squared error (MSE) between the estimate cover $C_{e,m}$ and the ground truth cover $C_{g,m}$ for each image $m \in \mathcal{M}$, which is computed as $\frac{1}{|\mathcal{M}|} \sum_{m \in \mathcal{M}} (C_{e,m} - C_{g,m})^2$. The square root of this quantity is often taken to get the root mean squared error (RMSE), in order to aid interpretation of the units.

IV. Methods

A. Annotations

We explore the following three types of annotations.

1) **Image-level cover annotations** In this method, annotators write down a single estimate of C by looking at the overall image.
2) **Segment-level cover annotations** In this method, the image is first segmented into rectangular regions, and users estimate C for each of those segments. In the single-layer cover setting, our pipeline simply partitions the image into disjoint 1000x1000 pixel segments. In the two-layer cover setting, the segments correspond to rectangular regions of the original image that contain the background class. This decomposes the problem for the annotator as (once the image is segmented) they only need to estimate the amount of the foreground class currently shown in the segment, without needing to worry about first identifying the regions corresponding to the background class.

3) **Bounding boxes** In this method annotators draw bounding boxes around objects of the target class. For the two-layer image cover problem, the images were pre-segmented so that users would only need to draw bounding boxes around the foreground class.

B. Deep Learning architectures

We now consider machine learning frameworks for each of these annotation types. The goal is to use ML to reconstruct the annotation information on new images, for example, drawing bounding boxes to match the ones specified in item 3. Items 1 and 2 fall into the framework of image regression, where given an image, the desired ML output is a single scalar number. Item 3 falls into the well-studied area of object detection, where given an image the machine learning system places bounding boxes around all objects of interest.

For object detection, we use a pipeline based on the state-of-the-art RetinaNet [14] object detection architecture. For the RetinaNet backbone we used the popular ResNet50 backbone [15], a 50 layer convolutional neural network. Because of our limited data, we used pretrained weights from the well-known ImageNet dataset [3] to initialize the network. The model was trained using the open-source Keras-RetinaNet library, for a total of 10 epochs, at 1500 steps per epoch. Per RetinaNet defaults, all images were resized to have a minimum size of 800 pixels and a maximum size of no more than 1333 pixels. To compensate for the small size of our dataset, we used limited forms of randomized data augmentation, specifically shear, scale, flip, rotate and translate.

For image regression, we use a feedforward convolutional neural network pipeline we wrote on top of the Keras and Tensorflow libraries. To keep things consistent with the object detection framework, for the top layers of our network we use a ResNet50 network, with weights initialized through pretraining on the ImageNet dataset. We flatten the output from these layers to 1 dimension, and added two densely-connected layers. The first had 128 units, and the second has 64 units, and each used rectifiers as the activation functions (i.e., the units on each of the two layers were ReLUs). To mitigate overfitting, we dropout 25% of the nodes on all of these layers when training. Finally, we add a final layer with a single unit with linear activation, which produces the regression output. Since this is a regression problem, we use mean squared error (MSE) as a loss function. All input images were resized to 500 x 500 pixels before processing. We trained the model with the RMSProp optimizer, 100 steps per epoch, and a batch size of 1. Additionally, although training loss generally went down as we increased the number of epochs, we saw significant fluctuations in performance as the model was trained, meaning the final model did not always perform the best. Therefore, we saved off versions of the model as it was training and picked the one with lowest loss on the training data.

C. Transforming other formats into cover

One challenge we encountered when building this pipeline is that, in order to compare cover across annotation styles, we need to first transform the annotations (whether human or machine generated) into a single cover value per image. This is trivial for the image-level cover numbers, but the other two methods needed a bit more thought.

To transform segment-level annotations into image-level cover value, a naïve approach would simply be to take the average reported reading over all segments. One problem here is that, in the two-layer cover problem, images can be of various different shapes and sizes based on the size and shape of the background objects. As such, when computing image-level cover, it does not make sense to weigh each equally. Therefore, we weight the mean by the area of each segment, as follows:

$$\frac{\sum_{s\in\mathcal{S}} l_s w_s y_{e,s}}{\sum_{s\in\mathcal{S}} l_s w_s} \tag{1}$$

where $\mathcal{S}$ is the set of segments, l_s is the length of segment s, w_s is the width of the segment s, and $y_{e,s}$ is the estimated cover value for segment s.

The above equation tells us that in the single-layer cover problem, all segments will be equally weighted, as the segments are all a fixed size of 1000x1000 pixels. However, the input images may be of various different sizes, which visually manifests itself as black borders on the edge of certain segments. A segment which only contains a small amount of actual image should not be weighted the same as a segment which came from the center of the image. To deal with this, we built a script which automatically analyzes the segments to quickly calculate the length and width that is actually from the source image, and uses that in equation 1.

To transform bounding boxes to cover values, we were careful not to double-count overlapping boxes, instead calculating the total area covered by any box.

V. Marine Science Case Studies

We study annotation styles for image cover analysis in the context of an important marine science problem, protecting the health of coral reefs. Coral reefs have begun deteriorating globally at an unprecedented rate [16], in large part due to human factors. Since coral reefs are host to a large amount of biodiversity [17], including numerous species of fish and invertebrates, the loss of coral presents grave concerns for the entire marine ecosystem.

In order to make appropriate management decisions to conserve coral reefs, policymakers need a detailed understanding of their current health status. Although historically this has been done though manual SCUBA surveys [18], more recent methods have divers or remotely operated vehicles (ROV) collect large amounts of image data, which is later analyzed in digital form to get accurate readings [19], [20]. Typically the first step in the data analysis pipeline is to stitch together imagery to form complete maps of 1 m^2 areas, called **quadrats**. Then marine scientists compute and analyze various metrics based on this imagery.

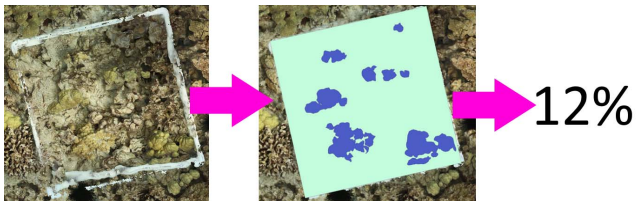

Fig. 1. Standard pipeline. Quadrat images are segmented by hand to mark the coral cover areas; however, the goal of this process is to get a single number representing the percentage of live coral. Note the difficulty of this task, both in locating the live coral colonies as well as tracing their complex boundaries.

Two of the most important metrics in this setting are live coral cover and disease severity. Live coral cover is a single-layer cover problem, where the goal is simply to determine what proportion of the image belongs to live coral colonies. Disease severity is a two-layer cover problem, where the goal is to determine how much of the live coral area is covered by disease. Coral disease comes in many forms, two of the most common being bleaching (where coral begins to turn white, often due to increasing ocean temperatures), and pigmentation response (where the coral begins to turn a pinkish hue due to external irritation). In this study, annotators were asked to count any disease, regardless of the specific type.

Our specific dataset was collected off the coast of Hawaii Island. Photos collected by SCUBA divers and snorkelers were fed into Agisoft Photoscan, which used Structure-from-Motion (SfM) techniques (as in previous marine science studies [21]) to render coherent 3D models of the scene. Those 3D models were then projected into 2D images, and then rotated and cropped so that the boundaries of the image matched the boundaries of the 1 m^2 quadrat.

The standard process for extracting these two cover metrics from the digital imagery requires a large amount of human effort to carefully trace the coral and disease outlines, and yet the end goal is simply to provide the cover percentages (see Figure 1). We seek to determine how we can best reduce this effort using machine learning techniques. Although there have been some prior explorations of how to apply machine learning to these problems [22], none have compared human feedback across fundamentally different types of annotations.

Ten annotators[1] were recruited to provide the desired annotations. All annotators had completed basic training on identifying coral and recognizing various types of coral disease. Annotators were instructed to be as accurate as possible, but to try and limit the time spent on the annotations. To give us a sense of how much time each type of annotation took the annotators, we recorded the time spent when completing the last several annotations of each type.[2]

We separated the dataset into a training dataset and held-out validation dataset, in order to test how our machine learning models generalized to new data. Although the typical approach would be to randomly hold-out some portion of the data, this would not capture the fact that some quadrats are geographically very close. Quadrats that are close together are likely to share similar characteristics, making generalization artificially easy. Therefore, to test whether our machine learning model generalizes to *new* locations, we were careful to place geographically similar regions within the same dataset.

To give us ground truth readings, annotators were instructed to fully and carefully trace/segment coral colonies and diseased areas, as per the middle image in Figure 1. Annotators traced 46 quadrats, recording the amount of time for 23 of them to give us a baseline point of comparison.

Table I shows the number of images annotated in each scenario. Some methods analyze overall quadrat images, while others analyze segment images (in which case we report the number of quadrats the images came from in parentheses). The amount of data is relatively low, but this is common when training ML systems to perform new image cover problems.

TABLE I
SIZE OF OUR TRAINING AND VALIDATION DATASETS, AS WELL AS THE NUMBER OF ANNOTATION TIME SAMPLES.

		Image-level	**Segment-level**	**Bounding Box**
Coral Cover	Train	20 quadrats	782 segments (22 quadrats)	19 quadrats
	Validate	19 quadrats	583 segments (19 quadrats)	20 quadrats
	Time	12 values	37 values	7 values
Disease Severity	Train	20 quadrats	61 segments (20 quadrats)	158 segments (14 quadrats)
	Validate	19 quadrats	60 segments (11 quadrats)	62 segments (5 quadrats)
	Time	18 values	8 values	21 values

VI. RESULTS

Our results for the coral cover study are shown in Table II. The "Annotation Time" column contains the average annotation time per quadrat for each of the three methods. RMSE stands for "Root Mean Squared Error" as calculated on the validation dataset, lower is better. Note that to determine how accurate human annotations were at predicting cover, we used our annotation transformation pipelines to compare human annotations (as well as ML predictions) to the ground truth (GT) segmentation, hence the "human RMSE" column. The best value in each column has been bolded. RMSE values for segmentation are not provided as it was used as ground truth.

TABLE II
OUR MAIN RESULTS FOR THE LIVE CORAL COVER CASE STUDY.

	Annotation Time	Human RMSE	ML RMSE
Image-level	**2.4 minutes**	11.36	**21.05**
Segment-level	4.7 minutes	**9.43**	26.90
Bounding Boxes	7.5 minutes	19.42	29.65
GT Segmentation	16.9 minutes	–	–

[1] 7 of which are authors of the paper

[2] As the initial timing results from each annotator are likely to be overly long due to annotators learning how to efficiently use the tools.

TABLE III
OUR MAIN RESULTS FOR THE DISEASE SEVERITY CASE STUDY.

	Annotation Time	Human RMSE	ML RMSE
Image-level	**1.6 minutes**	15.10	**8.35**
Segment-level	16.2 minutes	13.15	16.79
Bounding Boxes	14.8 minutes	**6.63**	13.96
GT Segmentation	24.3 minutes	–	–

As we expected, annotation times increase as the type of annotation becomes more complicated, with segmenting images taking much longer than the other three methods. In terms of human accuracy, we expected that drawing bounding boxes would be the most accurate due to the additional detail; however, it performs by far the worst of the three human annotation methods, which could be due to the difficulty of drawing accurate bounding boxes around irregularly-shaped coral colonies. We find that machine learning performs best when it is given simple image-level annotations, which was surprising given the low amount of detail in these annotations compared to segment-level or bounding box annotations.

The results of the case study on disease severity are shown in Table III. To ensure a fair comparison, the reported annotation time for the second two rows of coral disease annotation include the time it took from humans to pre-process the images into individual coral colony images.

Disease severity annotation times also tend to increase with the complexity of the annotation task; however, annotating disease severity on a per-image level is an order of magnitude faster than the other methods. We found humans to be most accurate when specifying annotations in the form of bounding boxes, perhaps due to the ability to provide more fine-grained information. For machine learning; however, we again found that the overall image-level annotations performed best in terms of accuracy. In fact, the RMSE of 8.35 was substantially better than human image-level annotations[3], and close to the best human RMSE of any of the three annotation types (6.63), indicating the efficacy of this approach.

VII. DISCUSSION

Overall, we found the results to be surprising. We expected that more detailed and informative annotations would correspond with increased performance for both humans and ML systems, but the RMSE of the ML bounding box approach was quite high in both settings. Instead, the simplest possible annotation type (image-level annotations) seemed to perform best when used as training data for deep learning systems.

Given the high RMSE of the bounding box approach, one might suspect that the object detection system was unable to make much progress on learning the task given such limited data. However, Figure 2 shows that this was not the case: in fact, in certain cases the machine learning found coral colonies and diseased areas that the human annotators missed, or alternatively, identified regions of possible inconsistency in the annotations. Of course, the machine learning system was not perfect, as it did miss some objects. But, as Figure 3 shows, those errors tended to be small or similar in appearance to regions of the image that were not marked.

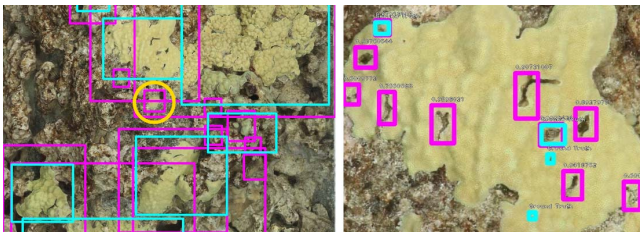

Fig. 2. Examples of where methods trained on bounding boxes outperform human annotators. Cyan boxes indicate human annotations and purple boxes indicate machine learning annotations. The left image shows a coral cover example and the right shows an example from disease severity. The left image has circled a colony that ML identified and humans missed, while in the right case, the human missed several potentially diseased areas that were very similar to the one they marked, showing potential inconsistency.

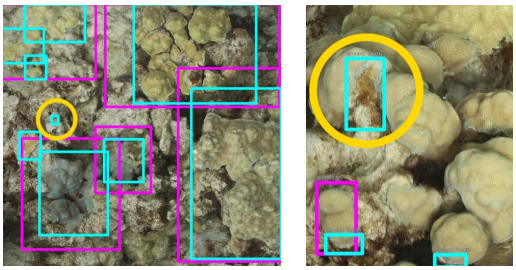

Fig. 3. Examples of where our bounding box methods miss key objects (circled) that the human successfully marked. The left image shows a coral cover example and the right shows an example from disease severity.

Therefore, one might suspect that bounding boxes might be a poor fit for this setting, since even the tightest bounding boxes might greatly overestimate the surface area of irregularly-shaped objects. Although this may be the case for coral cover, we actually found this to be the *best* form of human annotation in the disease severity study, indicating that bounding boxes were at least a reasonable fit for that task.

Instead, we suspect that the underlying cause of these results was the objective function employed by the deep learning algorithms. For bounding boxes, the objective function involves trying to match the input bounding boxes as accurately as possible (indeed, RetinaNet in large part uses focal loss [14] which is based on this). However, a relatively insignificant bounding box error (e.g. making the bounding box 10% larger in terms of area) can have a major impact on reported cover, and additionally, a large error in bounding boxes (e.g. having two bounding boxes instead of one covering a region) may have little to no impact on the reported cover. Therefore, even though the bounding boxes produced looked largely reason-

[3]One might wonder how it is possible for the ML system to perform better than the human annotators, since it was trained on their data. The answer is that if human annotations are inconsistent and have errors, and the machine learning is unable to successfully model how those errors occurred, it may perform better simply by keeping its predictions closer to the average value.

able, the resulting cover metrics were substantially different.

This also explains why, contrary to our expectations, the image-level annotations performed best. In those cases, the ML objective function was perfectly aligned with the overall cover task, while in the other cases the connection was more tenuous. This suggests that, if the goal is simply to determine cover-level metrics, it may be possible to get the "best of both worlds": high quality machine learning performance, but with little human effort required during data annotation.

Although our results are promising, we would caution against completely abandoning bounding box and segment-level annotations. More detailed annotations have other benefits that allow one to compute more complex statistics than simply the surface area (for instance, counting the objects or analyzing features of their shape).

Additionally, we acknowledge several limitations of our study. To emulate what occurs in a novel image cover problem, our study only uses a relatively small amount of training data, results may differ on well-established image cover settings. Also, annotating coral cover and disease is challenging even for trained human annotators; results may differ in other domains where human accuracy is higher, such as tree cover. Finally, although we use state-of-the-art deep learning methods, it is always possible that improvements to the computer vision algorithms might alter these results. Despite these limitations, we feel our results give valuable insight into how to approach novel and challenging image cover problems.

VIII. Conclusion

In this paper we have presented (to our knowledge) the first comparison of annotation types for image cover problems. We recruited annotators to annotate data in several different ways, built deep learning pipelines to process the various kinds of annotations, and ran case studies on two important image cover problems from the domain of marine science. We found that the type of annotation has a substantial impact on the performance of these systems, and that certain types of annotations are much better than others at balancing human effort and machine performance. Our results indicate that the deciding factor in machine learning performance in this space is how well-aligned the machine's objective is with the cover problem, which in our case happened to be the image-level annotation setting. This suggests directions for future work in terms of modifying machine learning systems to make use of more complicated forms of annotation while respecting (or perhaps even inferring) the user's desired objectives.

IX. Acknowledgments

We wish to thank Chad K. Kinoshita, Sofia B. Ferreira, Alexandra Runyan, Max Panoff, Mia Lamirand, Briana Craig, Ashley Pugh, and Bob Pelayo for their assistance and support.

References

[1] D. Ardila, A. P. Kiraly, S. Bharadwaj, B. Choi, J. J. Reicher, L. Peng, D. Tse, M. Etemadi, W. Ye, G. Corrado, D. P. Naidich, and S. Shetty, "End-to-end lung cancer screening with three-dimensional deep learning on low-dose chest computed tomography," *Nature medicine*, p. 1, 2019.

[2] B. G. Weinstein, S. Marconi, S. Bohlman, A. Zare, and E. White, "Individual tree-crown detection in rgb imagery using semi-supervised deep learning neural networks," *Remote Sensing*, vol. 11, no. 11, p. 1309, 2019.

[3] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.

[4] L. Ma, M. Li, X. Ma, L. Cheng, P. Du, and Y. Liu, "A review of supervised object-based land-cover image classification," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 130, pp. 277–293, 2017.

[5] E. S. Papazoglou, L. Zubkov, X. Mao, M. Neidrauer, N. Rannou, and M. S. Weingarten, "Image analysis of chronic wounds for determining the surface area," *Wound repair and regeneration*, vol. 18, no. 4, pp. 349–358, 2010.

[6] A. Bearman, O. Russakovsky, V. Ferrari, and L. Fei-Fei, "Whats the point: Semantic segmentation with point supervision," in *European conference on computer vision*. Springer, 2016, pp. 549–565.

[7] K. Konyushkova, J. Uijlings, C. H. Lampert, and V. Ferrari, "Learning intelligent dialogs for bounding box annotation," in *CVPR*, 2018, pp. 9175–9184.

[8] O. Russakovsky, L.-J. Li, and L. Fei-Fei, "Best of both worlds: human-machine collaboration for object annotation," in *CVPR*, 2015, pp. 2121–2131.

[9] D. P. Papadopoulos, J. R. Uijlings, F. Keller, and V. Ferrari, "Extreme clicking for efficient object annotation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 4930–4939.

[10] O. Beijbom, P. J. Edmunds, C. Roelfsema, J. Smith, D. I. Kline, B. P. Neal, M. J. Dunlap, V. Moriarty, T.-Y. Fan, C.-J. Tan *et al.*, "Towards automated annotation of benthic survey images: Variability of human experts and operational modes of automation," *PloS one*, vol. 10, no. 7, p. e0130312, 2015.

[11] S. Dutt Jain and K. Grauman, "Predicting sufficient annotation strength for interactive foreground segmentation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 1313–1320.

[12] M. Richardson, D. Karcher, and L. Purcell, "Quantifying turfgrass cover using digital image analysis," *Crop Science*, vol. 41, no. 6, pp. 1884–1888, 2001.

[13] G. M. Gavazzi, F. Madricardo, L. Janowski, A. Kruss, P. Blondel, M. Sigovini, and F. Foglini, "Evaluation of seabed mapping methods for fine-scale classification of extremely shallow benthic habitats–application to the venice lagoon, italy," *Estuarine, Coastal and Shelf Science*, vol. 170, pp. 45–60, 2016.

[14] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.

[15] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016, pp. 770–778.

[16] B. A. Weiler, T. E. Van Leeuwen, and K. L. Stump, "The extent of coral bleaching, disease and mortality for data-deficient reefs in eleuthera, the bahamas after the 2014–2017 global bleaching event," *Coral Reefs*, vol. 38, no. 4, pp. 831–836, 2019.

[17] T. P. Hughes, M. L. Barnes, D. R. Bellwood, J. E. Cinner, G. S. Cumming, J. B. Jackson, J. Kleypas, I. A. Van De Leemput, J. M. Lough, T. H. Morrison *et al.*, "Coral reefs in the anthropocene," *Nature*, vol. 546, no. 7656, pp. 82–90, 2017.

[18] L. J. Raymundo, C. S. Couch, C. D. Harvell, J. Raymundo, A. W. Bruckner, T. M. Work, E. Weil, C. M. Woodley, E. Jordan-dahlgren, B. L. Willis *et al.*, "Coral disease handbook guidelines for assessment, monitoring & management," 2008.

[19] W. Figueira, R. Ferrari, E. Weatherby, A. Porter, S. Hawes, and M. Byrne, "Accuracy and precision of habitat structural complexity metrics derived from underwater photogrammetry," *Remote Sensing*, vol. 7, no. 12, pp. 16 883–16 900, 2015.

[20] D. T. Bayley and A. O. Mogg, "New advances in benthic monitoring technology and methodology," in *World Seas: An Environmental Evaluation*. Elsevier, 2019, pp. 121–132.

[21] J. Burns, D. Delparte, L. Kapono, M. Belt, R. Gates, and M. Takabayashi, "Assessing the impact of acute disturbances on the structure and composition of a coral community using innovative 3d reconstruction techniques," *Methods in Oceanography*, vol. 15, pp. 49–59, 2016.

[22] P. Lozada-Misa, B. Schumacher, and B. Vargas-Angel, "Analysis of benthic survey images via coralnet: a summary of standard operating procedures and guidelines," 2017.