A Detection Confidence-Regulated Path Planning (DCRPP) Algorithm for Improved Small Object Counting in Aerial Images

Miles Krusniak  
Yale University  
New Haven, Connecticut  
miles.krusniak@yale.edu

Keaton Leppanen  
Truman State University  
Kirksville, Missouri  
kjl3514@truman.edu

Zhicheng Tang  
University of Missouri  
Columbia, Missouri  
zt253@mail.missouri.edu

Fan Gao  
University of Missouri  
Columbia, Missouri  
fgyf8@mail.missouri.edu

Yizhen Wang  
University of Missouri  
Columbia, Missouri  
ywf3m@mail.missouri.edu

Yi Shang  
University of Missouri  
Columbia, Missouri  
shangy@missouri.edu

Abstract—Computerized object counting shows potential for conservation population estimates as an alternative to manual counting, but remains unsatisfactory for minuscule objects, such as those in UAV-produced images. We improve UAV data collection by using a novel path planner which shifts altitude to maximize deep learning-based object detection confidences from a Faster Region-Convolutional Neural Network, considering energy consumption trade-offs. Using an empirical altitude-confidence relationship, our adaptive path planner (“DCRPP”) adjusts UAV altitude based on confidence, yielding better quality data given energy constraints. DCRPP achieves 11.92% greater accuracy compared to fixed-height methods in our conservation-aimed simulation.

Index Terms—Unmanned aerial vehicle, small object detection, aerial image, UAV path planning, deep learning

I. INTRODUCTION

As object detection with machine learning in images improves, small object detection has the potential to become a reliable way to automate tedious, inaccurate human counting using aerially collected data. As an example, for conservation purposes, it is useful to have a highly accurate population estimate of waterfowl in order to issue hunting licenses and guide natural resource management efforts. Currently, such counting is done as a human best guess from a manually controlled aircraft, but using an object detection system with data collected from a UAV could drastically improve the process in terms of cost and accuracy.

However, high-quality data is required for such models to achieve accurate results. For instance, resolution can have a significant effect on the accuracy of the machine learning model [1]. Furthermore, higher-quality images aid in distinguishing between two closely placed objects that may appear to be a single object at a higher altitude, a common situation in our conservation use case. When considering data collected with UAVs, we have a great level of control over the quality of the collected data, as decreasing the flight altitude of the drone will produce higher-quality images as the area captured shrinks [2]. On the other hand, decreasing the flight altitude also increases the time required for the drone to cover a given area, which can induce a constraint due to energy limitations [3].

Most contemporary aerial data collection methods use a fixed-height flight plan, such that all images are the same resolution. The area is then traversed with any of a variety of coverage path planning (CPP) algorithms. Simple two-dimensional CPP algorithms have been heavily covered in the literature, usable for automating tasks such as sweeping [4], demining [5], and agricultural work [6]. However, if the object detection model is able to run live as the drone collects data, then the drone may change height dynamically in order to optimize the confidence of the model.

For instance, consider the area shown in Fig. 1 and Fig. 2. The entire area is within the field of view at 100 meters of altitude, there are nine birds to be detected, and the confidence of correctly identifying each bird at 50 meters is shown next to the bird. (A confidence of 50% or greater is considered satisfactory for this example.) Assume that confidence decreases linearly with height, such that a one-meter increase in altitude corresponds to a one-percent decrease in confidence. The drone is to start in the lower left corner of the area.

Fig. 1 shows a naive, fixed-height approach to the problem: traverse the entire area at the lowest height needed to ensure a confidence greater than 50% for the minimum confidence bird. Each green box denotes an image taken at some altitude; in this naive approach, every image is taken at an altitude of 25 meters. While this approach does guarantee acceptable results, it requires a long path, much of which is spent over areas with no birds at all. Alternatively, were the drone to take a single image at 100 meters, the area captured by the image would be...
greater and the resulting path length would be very short, but almost all birds would have a detection confidence less than 50% at that altitude.

Fig. 2 shows a dynamic-height approach. First, an image at 100 meters is taken to profile the entire area, finding potential birds at a low confidence. Then, smaller images are taken to ascertain birds for which the confidence level is too low to be acceptable initially. It requires a significantly shorter path length than the low-altitude naive approach but retains its high accuracy.

The previous example makes some heavy assumptions: for instance, in reality, the relationship between altitude and detection accuracy is not linear, and the possibility of false positives is not considered. In our research, we have addressed these assumptions; for example, we have experimentally determined the altitude-to-detection-confidence relationship for the waterfowl-counting scenario. Using that relationship, we have developed a height-variant algorithm, referred to here as DCRPP (Detection Confidence-Regulated Path Planning) to obtain a greater average accuracy under the same energy constraints. Using a simulator created to facilitate the development of this new type of traversal algorithm, we tested DCRPP and found it to generate a path with up to 11.92% greater average accuracy than the best naive, fixed-altitude approach.

II. BACKGROUND

A. Small object detection

Many authors have observed significant decreases in the accuracy of even the most state-of-the-art object detection algorithms when used to detect small objects as compared to medium or large objects [7]. Furthermore, researchers have observed a significant relationship between the resolution of input images and the detection accuracy of the images they contain. Objects comprised of fewer pixels, such as those in lower-resolution images including UAV imagery, are notably harder for deep learning algorithms to detect and count than objects which are larger or of higher resolution [1].

For the purposes of this research, a small object is any object which has a rectangular area of 1024 pixels or fewer [8]. By this metric, many aerial images of waterfowl are exceptionally small. The rectangular area of medium to small sized waterfowl in the aerial photos collected for this research ranges from 35 to 650 pixels in images taken from altitudes between 50 and 150 meters. As such, waterfowl in these images are particularly difficult for deep learning algorithms to detect.

B. UAV coverage

From the perspective of data collection, several areas of UAV trajectory planning interact with the object-counting problem. Direct path planning (with object avoidance) for UAVs has been a subject of heavy research, but our context occurs in a largely free space which must be covered instead of only traveled through. As a more problem-relevant example, significant research has been done towards using UAVs for search and rescue operations, which require object localization. However, search and rescue operates under a significantly different set of parameters than object counting (for instance, prioritizing speed over accuracy and using multiple drones), which is not applicable to a conservation scenario. Additionally, for conservation use case, areas with no objects detected are significant, whereas in search and rescue, areas with survivors are more urgent.

Addressing a different component of the UAV counting problem, changes in altitude have been used in other research to affect UAV image data and improve an algorithm. For instance, Helble and Cameron [9], in their development of an aerial tracking system, used altitude to overcome local minima when determining the trajectory of a target. Though tracking is a distinct problem from counting, the idea of using altitude in an otherwise two-dimensional scenario to improve an algorithm’s performance is highly relevant.

C. Coverage path planning

Current naive solutions to the aerial object-counting problem as a whole are based on the coverage path planning (CPP) problem, formally stated as follows: Given a two-dimensional region \( R \) and a tool \( T \) (here, the field of view), determine a path \( P \) such that every point in \( R \) is within \( T \) when \( T \) is centered on some point on \( P \), and so that some cost function \( \gamma(P) \) is minimized. (Generally, \( \gamma(P) \) is the amount of energy required to travel \( P \).)

A variety of CPP algorithms exist, four of which are shown in Fig. 3. We tested Choset’s boustrophedon algorithm [10] and two variants of the minimum-sum-of-widths (MSOW)
algorithm [11], [13] on twenty random polygons which each
had five to fourteen sides, based on real-life bird refuge
boundaries for our conservation use case. The results are
summarized in Fig. 4, where “coverage quotient” is the ratio
of the generated path to the bare minimum path (that is,
the area of the shape divided by the tool width). Choset’s
boustrophedon method and Huang’s MSOW algorithm are
important nuances, such as the following:

- The CPP problem assumes a constant tool, whereas the
  field of view of a UAV varies with altitude;
- all area in the CPP problem is to be covered with a
  minimum cost, whereas in the counting problem, only
  objects are of interest and not the space between them;
- space covered in the CPP problem is immediately-con-
dered "fully covered," but in the counting problem,
undetected objects may still exist in an area which has
been covered, and an area covered with a high confidence
is preferable to an area covered with low confidence.

Our approach will focus on using those nuances to create a
better path planner for the counting problem which, instead of
focusing directly on minimizing energy cost, covers an area
as accurately as possible with an energy budget.

III. PROPOSED APPROACH

A. Problem formulation

Area formalization. Consider a two-dimensional region $R$
which contains a set $B$ of $n$ objects of interest $\{b_1\ldots b_n\}$ and
a set $C$ containing $m$ objects not of interest $\{c_1\ldots c_m\}$. Let the
stochastic function

$$\rho(j \in B \cup C, h)$$

represent the confidence of detecting object $j$ at height $h$ to
be of interest, and let the function

$$\eta(j) \in \{-1, 1\}$$

be a binary function representing the ground truth of the
object, such that $\eta(j \in B) = 1$ and $\eta(j \in C) = -1$.

Drone state formalization. Let $p_0 \in \mathbb{R}^3 = (x_0, y_0, z_0)$
denote the initial position of the drone. Let

$$\tau(p) \in \mathbb{R}^3 = \{j \in B \cup C : j \in T(p)\}$$

where $T(p) \in \mathbb{R}^3$ denotes the sweep area of the tool carried
by the drone at position $p$. Let

$$p_i = (x_i, y_i, z_i) \in \mathbb{R}^3 = \delta(\{p_0, \ldots, p_{i-1}\}, \{\tau(p_0)\ldots \tau(p_{i-1})\})$$

for some function $\delta$ (representing a next-waypoint function) and
$i \in \mathbb{Z} > 0$.

Given the above formulation, the dynamic-height coverage
path problem asks the following: Find the path defined by
$\{p_0, \ldots, p_q\}$ such that the length of the path is less than some
constant $c$, and

$$\sum_{i=0}^{q+1} \eta(j_i) \rho(j_i, k(j_i))$$

, where $k(j) = \min(\{z_i : j \in \tau(p_i)\})$, is at its absolute
minimum value.

Since this is a generalization of the coverage path problem,
which is NP-hard, the dynamic-height coverage path problem
is also NP-hard.

Given our tool, which is a rectangular camera with fixed
direction, view width angle $\theta_w$, and view height angle $\theta_h$, we
can further formalize $T(p_i)$, so that

$$T(p_i) = [x_i - z_i \sin(\theta_w), y_i - z_i \sin(\theta_h)] \times [x_i + z_i \sin(\theta_w), y_i + z_i \sin(\theta_h)]$$

.
The remainder of this paper discusses an empirical determination of $\rho(j,h)$, the altitude-confidence relationship, and a heuristic $\delta\{p_0..p_{i-1}\},\{\tau(p_0)..<\tau(p_{i-1})\}$.

**B. Altitude-confidence relationship**

In order to inform algorithm design for the counting problem, the effect of drone altitude on a machine learning model’s ability to detect small objects and its confidence in the detected objects must first be quantified.

1) **Data collection**: Two datasets were used in the training and testing of the machine learning model used in this project.

The first is the Little Birds Aerial Imagery (LBAI) dataset provided by the Illinois Natural History Survey at the University of Illinois at Urbana-Champaign. This dataset contains $440 \times 3940$ pixel images featuring a variety of waterfowl in their natural habitats. These images include several different colors, shapes, resolutions, backgrounds, and scales. Due to the images’ size, they were segmented into sub-images of size $512 \times 512$ pixels, and edges produced by this segmentation were discarded [14].

The second is a dataset of height-variant data, collected and labeled specifically for this project. This dataset controls for a number of variables, such as lighting and background, so as to isolate the effect altitude has on detection accuracy and confidence. It contains $225 \times 3648$ pixel images featuring decoy waterfowl arranged upon a grassy background. There are three general sizes of decoys (small, medium, and large) with minor size variations in each category. Images were taken at altitudes ranging from 40 meters to 150 meters at increments of 10 meters using our target UAV, the DJI Mavic Pro. The images were segmented into $11,340 \times 512 \times 512$ pixel sub-images and edges produced by this segmentation were again discarded. Example images from the height-variant dataset can be seen in Fig. 5.

2) **Relationship quantification**: For the purposes of determining a relationship between altitude and detection confidence, we used Facebook’s Detectron implementation of the Faster-RCNN deep learning model with minor modifications. The model was trained using images from the LBAI dataset and tested using our own height-variant dataset. The F1 score for the test results for each altitude showed a significant decrease as altitude increased. Additionally, we separated results into false positives and true positives, and recorded average confidence output by the model for both at each altitude (with undetected true positives considered to have a confidence of zero). We found a negative relationship between altitude and confidence in true positives, as well as a positive relationship between altitude and confidence in false positives. We then performed a polynomial regression on the confidence data to determine a quantitative relationship between altitude and detection confidence of our model. Our results are shown in Fig. 6.

**C. Heuristic algorithm**

Our detection confidence-regulated path planning algorithm is based on a two-phase approach. First, before flight (indicated by the upper branch in “path planning” in Fig. 7), a naive path is generated within an energy budget. Second, during flight, the path is altered as objects are detected.

Before flight, the area is decomposed according to the minimum-sum-of-altitudes algorithm [13]. A naive path is generated using a constant altitude such that the proposed path has an energy cost less than $p\%$ of the budget, where $p$ is a constant related to the expected number of objects in the area. This plan is represented as a number of “overview points,” at each of which an image is to be taken such that the set of all overview images maps the entire area at a high altitude.

During flight, the drone initially travels from overview point to overview point, taking high-altitude images at each one. However, detection is immediately run on every image in order
to localize any possible objects in the frame. No confidence thresholding is used, and false positives are expected in the output. The detected points are reordered using any heuristic travelling salesman algorithm to create an approximate path of minimum length connecting the current location to the next overview point through points over every detected object.

For every detected potential object in the reordered list, a lower altitude $h$ is calculated such that the difference between the object’s detection confidence and the confidence of discrimination is predicted to be over some threshold when detected from $h$. (The threshold may be lowered to decrease the energy consumption of the drone at the cost of detection accuracy.) The drone is then to fly to altitude $h$ and to the closest location such that the object is within the drone’s field of view at that altitude. Any altitude-confidence relationship may be used within the drone to determine $h$ (regardless of the actual relationship, which may have some variance); in the examples that follow, we have assumed a simple, linear one to make our implementation detection-model-agnostic.

Once all detected objects have been visited at a lower altitude, traversal continues at the overview height until either the entire overview path has been travelled, or the drone only has sufficient energy to return back to the starting point.

IV. ANALYSIS

A. Simulator

We created a path planning simulator to aid in the development of our DCRPP algorithm. It is generalizable, and we hope that it may be a useful tool for developing more complex algorithms akin to DCRPP.

The simulator uses the specifications of the DJI Mavic Pro, a high-end consumer-grade drone which fits our conservation use case. The energy model used by the simulator is taken from DiFranco and Buttazzo [3] and re-scaled to match the specifications of the Mavic Pro (as their research used the older and less efficient, but otherwise comparable, Iris drone). The energy model accounts for expenditure in acceleration, deceleration, changes in altitude, and turns. The drone is assumed to maintain its most efficient speed (15.5 meters per second) whenever possible.

Areas of interest in our simulator may be randomly generated or loaded from preexisting files, such as those produced by Google Earth [15]. In either case, a number of true-positive and false-positive detectable objects are distributed in the area, by default as a uniform random distribution or optionally as a clustered distribution, as shown in Fig. 8.

Objects may be detected during traversal in the simulator with a confidence according to the altitude-confidence relationship we previously determined empirically, and with Gaussian noise determined empirically in the same way. (Objects detected with confidence less than or equal to zero are considered undetected.) Once traversal is complete, a guess is generated for each object based on the detection confidence of the object at the lowest altitude it was detected at, and precision, recall, and other statistics are calculated.

B. Algorithm results

To compare our DCRPP algorithm with the naive approach, we used a modified version of Huang’s minimum-sum-of-altitudes algorithm as a standard of comparison, in which the entire area is traversed at the lowest height possible such that the traversal cost is less than the energy budget. We tested this naive algorithm against our own on randomly generated pentagons (to reduce test duration, as the minimum-sum-of-altitudes algorithm is time complex with the number of concave vertices in the area).

The results of this comparison for a random distribution of fifty objects are summarized in Fig. 9. Fig. 10 shows the accuracy curve for an energy budget of 150 kilojoules and a varied number of objects. Our DCRPP algorithm shows a consistently better performance in the 100 kilojoules to 200 kilojoules range, where the energy capacities of many
commercial drones lie. The Mavic Pro, for instance, has a capacity of about 157 kilojoules [16].

We also compared the two algorithms on the clustered distribution of objects, yielding Fig. 11 and Fig. 12. Note that the decrease in accuracy as the number of objects rises is significantly less steep, suggesting that it is not the increased number of objects, but the increased number of clusters, which requires more energy to cover. Our DCRPP algorithm creates a path with higher average accuracy with a clustered distribution, making it promising for the conservation use case, where clusters of birds are to be expected. In fact, since the simulator does not account for effects of object proximity and overlap on confidence, and as lower-altitude imaging intuitively makes separating objects easier, it is conceivable that the confidence payoff for descending, leveraged by DCRPP, is greater in reality than in the simulator.

V. CONCLUSION

The use of the detection confidence-regulated UAV path planning algorithm as proposed here to facilitate data collection for counting small objects leads to a significant improvement in accuracy when compared to naive, fixed-height approaches. Specifically, when tested using a simulator designed for this problem, this algorithm shows an 8.68% increase in accuracy when surveying objects arrayed in a random distribution and an 11.92% increase in accuracy when surveying objects arrayed in a clustered distribution, given an energy budget comparable with state-of-the-art drones.

Contributions to the field derived from this research include:

1) a labeled and curated height variant dataset of aerial small object imagery (decoy birds), 2) a quantified altitude-detection confidence relationship for small objects using deep learning, 3) a mathematical definition for a height-variant UAV coverage path planning problem, 4) a generalizable simulator for energy-constrained UAV coverage path planning problems, and 5) a novel and generalizable detection confidence-regulated UAV path planning algorithm.

To progress on these contributions, the proposed algorithm could be implemented on a physical drone to allow for field testing. In addition, the effects of utilizing different deep learning algorithms in conjunction with the proposed algorithm could be explored. Furthermore, these contributions are relevant in not only our use case of waterfowl counting explored in this paper, but also in similar scenarios such as crowd counting and invasive flora detection.

REFERENCES