Counting Buildings from Unmanned Aerial Vehicle Images Using a Deep Learning Based Approach

Evet Naturinda[1], Emmanuel Omia[2], Fortunate Kemigyisha[2], Jackline Aboth[2], Isa Kabenge[2], Anthony Gidudu[1]
[1]Department of Geomatics and Land Management, Makerere University, Kampala, Uganda, evet.naturinda@gmail.com
[2]Department of Agricultural and Biosystems Engineering, Makerere University, Kampala, Uganda

DOI: https://dx.doi.org/10.4314/sajg.v13i1.6

Abstract

Effective urban planning requires accurate and up-to-date spatial information. Remote sensing has contributed immensely to the efficiency of collecting this information. With remotely sensed high-spatial-resolution images, details such as buildings counted in an area can be extracted; however, traditional methods of extracting this information involve direct counting by humans, which is often demanding in terms of time. Computer vision techniques have shown promising results in handling image-related challenges in recent years. Therefore, this study aimed to adapt deep learning-based algorithms to simplify the counting of buildings from high-spatial-resolution aerial images in a fairly suburban environment. A deep learning algorithm based on convolutional neural networks, You Only Look Once (YOLO), was adapted to detect and count the buildings in the Unmanned Aerial Vehicle (UAV) sensed images. The model achieved high accuracy, with a recall rate of 0.89, an F1 score of 0.89, and an average precision of 91.12% on the validation data. When applied to new testing data, the algorithm successfully identified and counted the number of buildings with an overall accuracy of 71%. The approach presented in this research extracted building counts reliably, quickly, and accurately in a fairly suburban environment. Such information can be applied to tracking urban growth and physical planning.

Keywords: Object detection, Convolutional Neural Networks, Remote sensing, Physical planning.

1. Introduction

Accurate and up-to-date analysis of built-up areas is significant in policymaking, establishing an economic and social understanding of an area, and urban planning interventions. Statistics of building density play a vital role in urban planning and monitoring; however, they require detailed and laborious surveys that are still inadequately done in most developing countries (Shakeel et al., 2019). Most of the widely adopted methods of extracting this information from aerial images are traditional and manual, involving human interpreters counting buildings directly from high-resolution images, making them inefficient. High-resolution images have a Ground Sampling Distance (GSD) of one meter or less; moderate-resolution images have a GSD of about 15–100 meters; and low-resolution images have a GSD measured in hundreds of meters (DiBiase, 2014).

Integrating Artificial Intelligence and geospatial analysis that extracts desirable information, such as the number of buildings in an area, from a remotely sensed image can improve urban planning and disaster management. Increasingly, various industries are using UAVs equipped with imaging cameras, and there is still an ongoing improvement in methods to explore and understand the visual data retrieved from these platforms (Barbedo et al., 2020; Du et al., 2018; Hsieh et al., 2017; Xia et al., 2018; Zhu et al., 2018). Computer vision has played a vital role in improving the efficiency of handling image-related challenges such as image classification and segmentation. Furthermore, object detection and counting algorithms have progressed in the current field of computer vision with the recent development of deep learning (Zhu et al., 2020), leading to the development of several object detection techniques such as R-CNN, Fast R-CNN, Faster R-CNN, Single Shot Detector (SSD), and YOLO (Anand & Meva, 2020). The speed and accuracy of object detection and counting improve and increase as new techniques arise. The YOLO (Redmon et al., 2016) algorithm is one of the latest techniques providing fast and accurate results, with YOLO version 5 being the most recent by 2021 (Jiang et al., 2021).

According to Lechgar et al. (2019), YOLO is a state-of-the-art real-time object detection system that differs from other deep learning models because it sees the entire image once during the training and testing sessions. In addition, YOLO algorithms convolve learned functions with input data and use 2D convolutional layers, which makes them well-suited for processing 2D data, such as images. Among the many versions, YOLO version 2 (YOLO v2) is widely used in many academic research papers owing to its promising results since it is both better and faster than the original version and simplifies the network (Jiang et al., 2021). It is a Convolutional Neural Network (CNN)-based algorithm built on the Darknet-19 architecture with 19 convolutional layers that require only 5.2 billion operations. This makes it superior to the GoogleNet architecture used in version 1, which requires 8.25 billion operations, consequently reducing the amount of calculation (Jiang et al., 2021). The overall pipeline of the YOLO v2 algorithm (Shao et al., 2020) takes three steps: detect objects in each image, reconstruct the 3D surface of the background, and finally merge the per-frame detection results guided by the 3D surface. Merging over time eliminates the double detection of single objects and thus ensures the correct object count in a scene.

This study aimed to adapt deep learning-based algorithms to simplify the counting of buildings from high-spatial-resolution aerial images in a fairly suburban environment. The first section (1) of this work introduces the study concept, section two (2) explains the study method, and section three (3) presents and discusses the study findings and offers conclusions.

2. Methodology

2.1. Research approach

This study involved two phases, including model training and testing and the counting phase (Figure 1). During the data collection phase, a UAV equipped with a camera was used to capture

aerial images. Pre-processing involved renaming all the images with the common prefix, "img", followed by image numbering. Annotation and data augmentation, as described in the proceeding subsections, were done to label and increase the number of images six-fold. We trained and tested the CNN-based model, extracted the number of buildings in the tested images, and finally assessed the algorithm's performance based on precision, recall, and the F1 score.

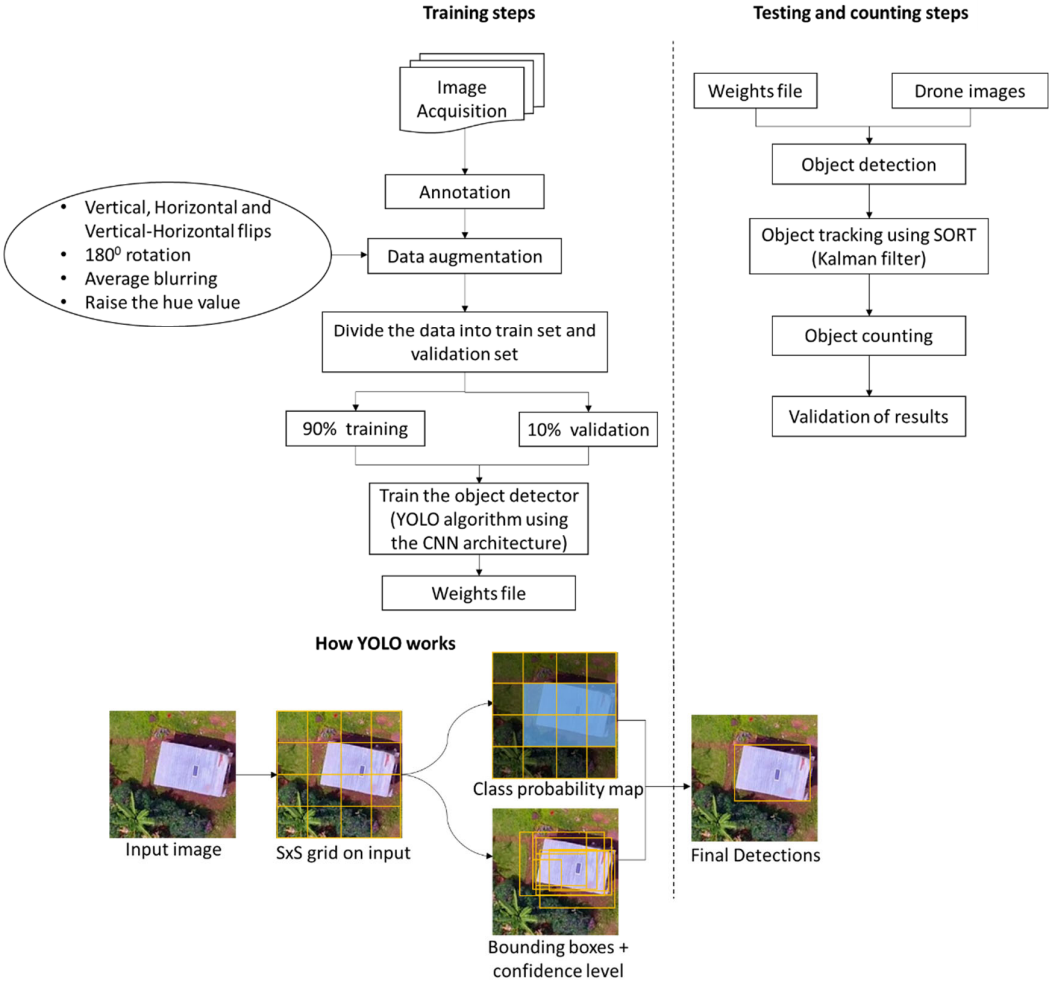

Figure 1. Research approach

2.2. Image acquisition

Remotely sensed images were acquired from Makerere University Agricultural Research Institute Kabanyoro (MUARIK) (Figure 2) in July 2021. Kabanyoro is located in Wakiso district, Uganda. It has an altitude range of 1250 to 1320 m above sea level and accommodates a fairly suburban settlement (Ivanova et al., 2021). A DJI Phantom 4 advanced UAV with an integrated camera of 12 megapixels was used. The study used UAVs because they can reveal the topographic view of an outdoor scene when they are equipped with high-resolution cameras capable of generating high spatial-resolution images. The UAV was flown at altitudes of 76 m and 107 m relative to the take-off position altitude, corresponding to Ground Sampling Distances (GSD) of approximately 2 cm/pixel

and 3 cm/pixel, respectively. These altitudes were chosen to ensure that the buildings could be observed at different heights above the ground for the model to learn from images of varying resolutions. The study considered this altitude range because it provides a suitable ground sampling distance to detect buildings. The frontal and side image overlaps were both set to 70%. Secondary data from the Kaggle dataset was used in addition to the collected data to increase the size of the training data and the generalization of the model (Luo, 2019). It included images of buildings taken from Zanzibar with characteristics similar to those of the study area.

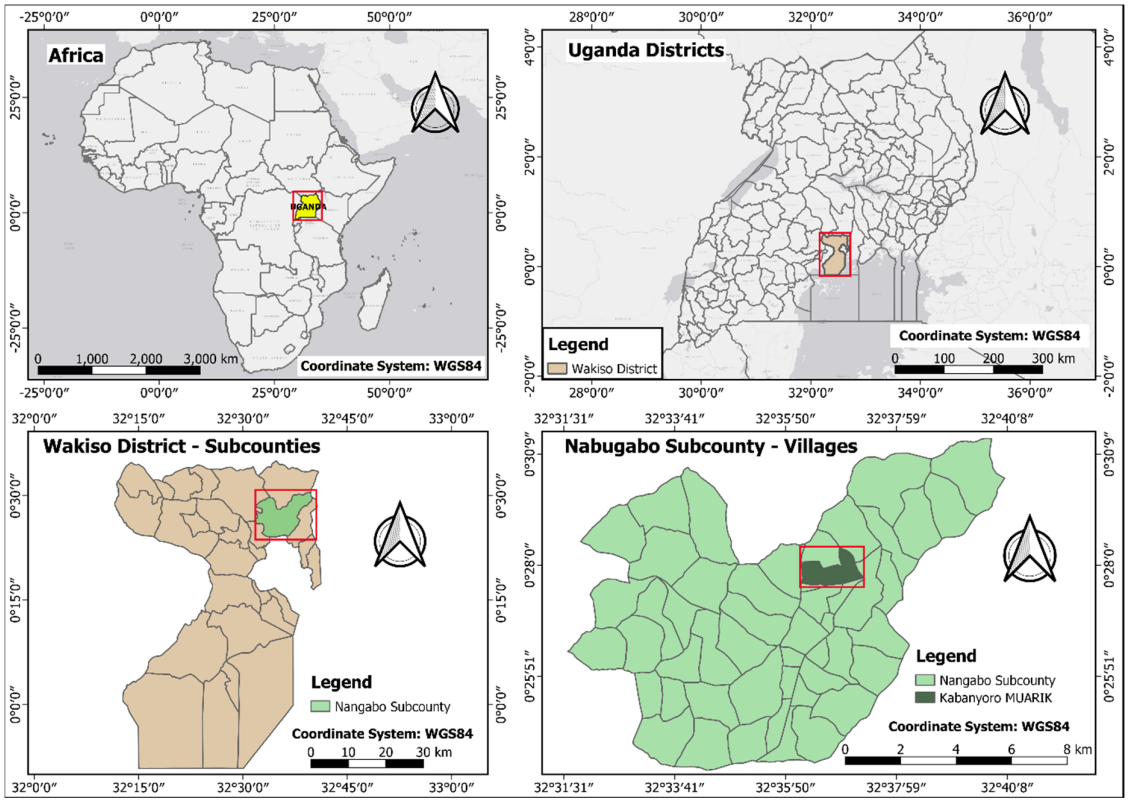

Figure 2. Map of the Study Area

2.3. Data preprocessing

Using the Agisoft Metashape software, the captured overlapping UAV images were merged to create a mosaic image for the entire area. We then split the image into regular tiles of size 480x480 pixels to match the input image size of the detection algorithm that was used. The individual tiles were sorted as either building or non-building. Non-building images were discarded, and only the remaining images containing buildings were used to train the model.

2.4. Data annotation

Computer vision algorithms require annotated or labeled data. Image annotation involves assigning labels to an image or target objects within an image. The bounding-box annotation style

was used because the detection algorithm that was applied detects objects with bounding boxes encapsulating them. In this project, LabelImg software was used to annotate the images. The software auto-generates a text file for each annotated image containing the location of the object in YOLO v2 format (category number, object center in X, object center in Y, object width in X, object width in Y). The labeled dataset was randomly split into a training set and a validation set. Since the variation in the nature and orientation of the buildings in the images was relatively high, the dataset was randomly split into 90% and 10% for training and validation, respectively. This was done because CNN models require a large amount of training data to capture the spatial and feature heterogeneity in the data. However, it is worth noting that the literature is not definitive regarding the minimum number of reference training samples required for CNN, and this requirement may vary depending on factors such as the size and spatial diversity of the area, as well as the image classification algorithm being used (Mafanya et al., 2022).

2.5. Data augmentation

As earlier mentioned, CNNs heavily rely on big data to avoid overfitting in Computer Vision tasks. To increase the number of training data, data augmentation was carried out on the images, a process that results in artificially increasing the training dataset size by transforming each image into n-images (n depends on the number of transformations chosen). For this study, the transformations that were used include flipping (vertical, horizontal, and vertical-horizontal), 180^0 rotation, average blurring, and raising the hue value (Figure). This was carried out using already defined data augmentation toolsets, available in Keras, running on top of the TensorFlow framework within the Google Co-laboratory (Google Colab) platform. Google Colab is an online platform hosted by Google that offers ready-to-use virtual machines of high computational power, such as Graphics Processing Units (GPUs) and Tensor Processing Units (TPUs).

Original Image Annotated Image Horizontal Flip

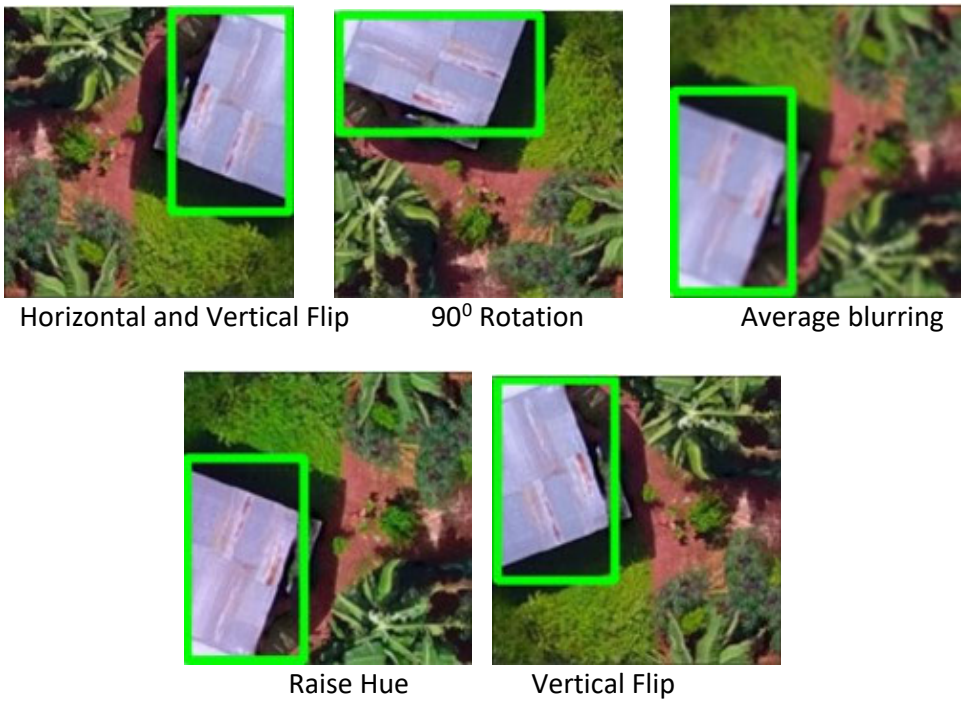

Figure 3. Augmented Images

2.6. Building detection

The YOLO v2 model built on darknet architecture was used for detection purposes in this study. YOLO v2 is a CNN-based deep learning algorithm specifically designed for object detection in imagery. In contrast to two-stage models, it skips the region proposal stage and thus requires only a single pass through the neural network to predict all the bounding boxes in an image; hence its name, You Only Look Once. This network concurrently predicts several bounding boxes and class probabilities for these boxes. The bounding boxes are weighted by the predicted probabilities. The YOLO v2 algorithm was trained and validated with a total of 11,560 and 1,284 images, respectively. The model was run for 5000 iterations until the performance improvement became fairly constant. Different hyperparameters, such as batch, width, height, and filters, were tuned to increase the performance of the model. The average loss error value reported on every iteration was monitored to track the performance of the model and to prevent the model from overfitting. Other values, such as precision, recall rate, the F1 score, and average precision, were also used to monitor the performance of the model. A weights file with the predicted probabilities was obtained at the end of the model training. Its accuracy was tested by applying it to a new set of images with different resolutions.

2.7. Counting the number of buildings

YOLO v2, being an object detector, generates bounding boxes with a class ID and confidence value for each bounding box. In this study, the tracking-by-detection approach was used for counting the buildings using the SORT (Simple Online and Real-Time Tracker) algorithm. The weight file obtained from training the model was used as input to the tracker. SORT uses the Hungarian algorithm

and Kalman filter to detect and track objects. It tracks each detection by assigning a unique ID to each bounding box, and as soon as an object is lost because of occlusion or wrong identification, the tracker assigns a new ID and starts tracking the newfound object (Bathija & Sharma, 2019).

2.8. Validation of Results

The performance of the model was tested by evaluating its results on the test set of images. The area covered in this testing set was diverse, containing both small and large structures in densely and moderately populated areas. Ground truth data were created by manually counting the number of buildings in each image. The model's prediction for each of the images was compared with the ground-truth count of the buildings in that image. To assess the accuracy of the algorithm's precision, recall, and the F1 score (Equations 1, 2, and 3), actual and estimated counts for each image were generated. Precision is the fraction of buildings detected among all the detections in an image. The recall is the fraction of buildings detected among all the buildings present in an image. The F1 score is a measure of a model's overall performance. These measurements range from 0 to 1, with 1 being the best.

$$\text{Precision} = \frac{TP}{TP+FP} \quad [1]$$

$$\text{Recall} = \frac{TP}{TP+FN} \quad [2]$$

$$F1 = \frac{2\times Precision\times Recall}{Precision+Recall} \quad [3]$$

Where; TP - True positive

FP - False positive

FN - False negative

3. Results and Analysis

3.1. Building detection and counting

The model was trained with images at ground resolutions of 2 cm to 3 cm to detect buildings from images of slightly varying resolutions. The model was trained for 5000 iterations, and its performance increased with the increase in the number of iterations run. The performance of the model was evaluated using the precision, recall, F1 score, and average precision computed at each model iteration. The model achieved high accuracy with a precision of 0.88, a recall rate of 0.89, an F1 score of 0.89, and an average precision of 91.12% on the validation data. The average loss at each iteration was also monitored for early stopping to prevent cases of overfitting where the model can detect objects in images from the training data only but can't perfectly detect objects in other images. This compares fairly well with other research works that have applied the YOLO v2 algorithm in object detection, such as Barbedo et al. (2020), which attained an F1 score of 90% in the application of

YOLO v2 in counting cattle from UAV images. Likewise, Lechgar et al. (2019) were able to detect the city's vehicle fleet using YOLO v2 with 91% accuracy.

Buildings Counted: 9

Buildings Counted: 29

Buildings Counted: 1

Buildings Counted: 14

Figure 4. Images showing the detected buildings

The developed approach detects buildings in isolation (Figure) with reliable accuracy; however, for buildings clustered in one place and small-sized buildings in some images, a few errors occurred and reduced the accuracy of the model. Instance segmentation, where detection occurs by delineating each distinct object of interest appearing in an image, could be applied to resolve this issue, especially in slum areas where the buildings are next to each other (Wen et al., 2019).

The methods in this study were applied to UAV images; nevertheless, for identifying and quantifying buildings over wider areas, high-resolution satellite images could prove more effective. These images offer broader coverage compared to UAVs, which makes them ideal for examining large study regions. However, the lower resolution of freely available satellite images, such as Landsat, limits their usefulness in detecting buildings within slum areas, or smaller structures.

3.2. Validation of results

The performance of the model was tested by applying the obtained weight file to a new set of test images. Figure shows the detection and counting of the building's predictions on the testing images. The model creates a bounding box around each identified building in the tested image and attaches a confidence level. The adapted algorithm performed well in detecting buildings, with an overall performance of 0.7111 (Table 1). Clustered buildings presented a complex situation, although the algorithm yielded reasonably good estimates. To improve performance, the model was trained with more iterations to perform better in such complex situations. The errors obtained were due to the differences in the sizes of the buildings. The errors can be greatly reduced with more training data that take into consideration all the building sizes and different resolutions.

Table 1. Accuracy results

Metric	YOLO V2
Precision	0.8205
Recall	0.6275
F1 Score	0.7111

Given the constraints of current datasets, models will need to be retrained whenever new conditions other than a fairly suburban environment are considered. This emphasizes the need for data sharing for more general solutions to be feasible.

4. Conclusion

The traditional ways of extracting information from UAV images involve manual counting by humans, a process that is inefficient and prone to error. This study adopted a YOLO v2 model to

detect and count buildings from high-resolution aerial images. The adapted model performed well, with 0.71 overall accuracy. The challenges encountered, although difficult to overcome, had a relatively mild impact on the overall accuracy of the proposed algorithm. Although the adapted algorithm was for counting buildings, the methodology can be adapted to other applications such as livestock detection and tent detection in refugee camps, among others. Whereas the model training steps of this study required a time input, the results presented highlight that the model obtained from this approach can be applied to automatically count buildings from new images taken from a fairly suburban environment without the need to modify the model. Ultimately, this provides an advantage to using this approach, which will become less costly and more prevalent with technological advancement.

5. Acknowledgements

We would like to acknowledge the Regional Universities Forum for Capacity Building in Agriculture (RUFORUM) and Global Research Alliance on Agricultural Greenhouse Gases (GRA) for funding this study. We would also like to acknowledge the anonymous reviewers whose advice has gone a long way in improving the quality of the original draft. This article is based on the paper presented at the AGRC 2021 conference.

6. References

Anand J. and Meva, D., 2020. A Comparative Study of Various Object Detection Algorithms and Performance Analysis. *JCSE International Journal of Computer Sciences and Engineering, 8*(10), pp. 158-163, DOI: https://doi.org/10.26438/ijcse/v8i10.158163.

Barbedo, J.G.A., Koenigkan, L.V., Santos, P.M. and Ribeiro, A.R.B., 2020. Counting cattle in UAV images—dealing with clustered animals and animal/background contrast changes. *Sensors*, *20*(7), p.2126, DOI:https://doi.org/10.3390/s20072126.

Bathija, A. and Sharma, G., 2019. Visual object detection and tracking using YOLO and SORT. *International Journal of Engineering Research and Technology.(IJERT)*, *8*, pp.705-708.

DiBiase, D., 2014. *Nature of Geographic Information: An Open Geospatial Textbook*.

Du, D., Qi, Y., Yu, H., Yang, Y., Duan, K., Li, G., Zhang, W., Huang, Q. and Tian, Q., 2018. The unmanned aerial vehicle benchmark: Object detection and tracking. In *Proceedings of the European Conference on Computer Vision (ECCV).* pp. 370-386.

Hsieh, M.R., Lin, Y.L. and Hsu, W.H., 2017. Drone-based object counting by spatially regularized regional proposal network. In *Proceedings of the IEEE international conference on computer vision.* pp. 4145-4153.

Ivanova, A., Denisova, E., Musinguzi, P., Opolot, E., Tumuhairwe, J. B., Pozdnyakov, L., ... & Krasilnikov, P. (2021). Biological Indicators of Soil Condition on the Kabanyolo Experimental Field, Uganda. Agriculture, 11(12), 1228. https://doi.org/10.3390/agriculture11121228.

Jiang, P., Ergu, D., Liu, F., Cai, Y., & Ma, B. (2021). A Review of YOLO Algorithm Developments. *Procedia Computer Science*, *199*, 1066–1073. https://doi.org/10.1016/J.PROCS.2022.01.135.

Lechgar, H., Bekkar, H. and Rhinane, H., 2019. Detection of cities' vehicle fleet using YOLO v2 and aerial images. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information* Sciences, pp.121-126, DOI: https://doi.org/10.5194/isprs-archives-XLII-4-W12-121-2019.

Luo, D. (2019). Zanzibar OpenAI Building Footprint Mapping. Kaggle. https://www.kaggle.com/datasets/sayantandas30011998/zanzibar-openai-building-footprint-mapping?datasetId=414435

Mafanya, M., Tsele, P., Zengeya, T. and Ramoelo, A., 2022. An assessment of image classifiers for generating machine-learning training samples for mapping the invasive *Campuloclinium macrocephalum* (Less.) DC (pompom weed) using DESIS hyperspectral imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, *185*, pp.188-200.

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition.* pp. 779-788.

Shakeel, A., Sultani, W. and Ali, M., 2019. Deep built-structure counting in satellite imagery using attention-based re-weighting. *ISPRS Journal of Photogrammetry and Remote Sensing*, *151*, pp.313-321, DOI: https://doi.org/10.1016/j.isprsjprs.2019.03.014.

Shao, W., Kawakami, R., Yoshihashi, R., You, S., Kawase, H. and Naemura, T., 2020. Cattle detection and counting in UAV images based on convolutional neural networks. *International Journal of Remote Sensing*, *41*(1), pp.31-52, DOI: https://doi.org/10.1080/01431161.2019.1624858.

Wen, Q., Jiang, K., Wang, W., Liu, Q., Guo, Q., Li, L., & Wang, P. (2019). Automatic Building Extraction from Google Earth Images under Complex Backgrounds Based on Deep Instance Segmentation Network. *Sensors*, *19*(2), 333. MDPI AG. Retrieved from http://dx.doi.org/10.3390/s19020333.

Xia, G.S., Bai, X., Ding, J., Zhu, Z., Belongie, S., Luo, J., Datcu, M., Pelillo, M. and Zhang, L., 2018. DOTA: A large-scale dataset for object detection in aerial images. In *Proceedings of the IEEE conference on computer vision and pattern recognition.* pp. 3974-3983.

Zhu, P., Sun, Y., Wen, L., Feng, Y. and Hu, Q., 2020. Drone-based RGBT vehicle detection and counting: A challenge. *arXiv preprint arXiv:2003.02437*.

Zhu, P., Wen, L., Bian, X., Ling, H. and Hu, Q., 2018. Vision meets drones: A challenge. *arXiv preprint arXiv:1804.07437*.