An Investigation of Context-Aware Object Detection based on Scene Recognition

Ayman Naushad
School of Mathematical and Computer Sciences
Heriot-Watt University Dubai
Dubai, UAE
syedayman10@gmail.com

Radu-Casian Mihailescu
School of Mathematical and Computer Sciences
Heriot-Watt University Dubai
Dubai, UAE
r.mihailescu@hw.ac.uk

***Abstract*—Visual scene understanding for humans entails identifying regions of interest and then reasoning about information gained through their contextual analysis. Attempts to reproduce this process using Computer Vision techniques play an important role in improving the performance of Object Detection and general scene understanding algorithms. In this paper, we augment the training algorithm of an Object Detection model by integrating contextual information in the form of 'scene labels', through various methods to identify the superior approach. Several deep learning models were implemented and evaluated; results show that the Contextually Aware Object Detection Model performed the best, producing the highest classification accuracy of 75.14% and mean bounding box IoU of 0.69 on the constructed dataset. This was achieved by virtue of an auxiliary scene classification model used to make image scene predictions during the training phase.**

Keywords—computer vision, object detection, context, scene recognition.

I. Introduction

Object detection is a computer vision technique that involves two tasks: identifying and localizing objects within images or videos. The algorithms used typically involve deep neural networks, such as convolutional neural networks (CNNs), to analyze visual data and accurately detect the presence and location of objects of interest. Object detection plays a pivotal role in various applications ranging from surveillance systems and autonomous driving to image search engines and augmented reality.

Objects in real life do not occur in solitude, they co-vary with other objects and their surroundings creating an abundant source of associations and correlations. Context provides essential information for perceptual inference, and studies have proved that sources of contextual knowledge such as object location, relative size, scene arrangements, etc. are key factors aiding humans in performing object detection tasks. Furthermore, recognizing objects was more precise when the strength of the object-scene relationship was high [1]. Certain objects are frequently found together in a common environment while others are likely to be found in particular geographic locations. Animals, for instance, are typically encountered in their natural habitats. The composition of a scene, along with other objects in an image, provide much semantic context. How an image is illuminated because of aspects such as the weather or camera parameters affect the color, shadows, and brightness of objects within it. Information derived from three dimensional physical constraints such as a fire hydrant requiring a ground plane is yet another example of potentially valuable contextual information [14].

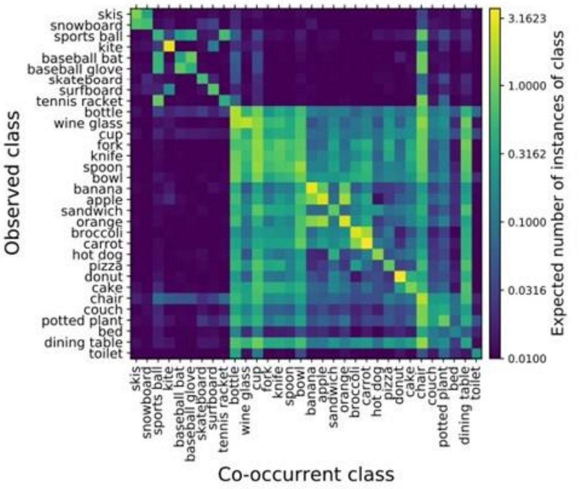

Fig. 1. Object co-occurrences in a subset of classes in COCO dataset [13]

Context plays a major role in the way humans perceive the world; hence it is rational to assume that the provision of contextual information in the object detection algorithm could enhance its performance. For instance, the modification of conventional bottom-up region proposal extraction processes to use context aware modules in a top-down processing fashion, which summarize global context features at the image level, to complement and assist the features in the pooled region proposals, has been shown to increase accuracy in object detection tasks with little extra overhead [2]. Several mechanisms that exploit global and local contextual features are implemented in the form of connective subnetworks that can be easily combined with existing models to aid the detection process [6,7].

In this paper, we propose an approach that uses the scene type of an image as a form of contextual information. By leveraging this scene-level information, the model is ought to gain a high-level view of the objects in the image in relation to their surrounding scene. We conduct an empirical evaluation and show enhanced robustness and improved handling of adverse conditions such as occlusions or cluttered backgrounds. The rest of the paper is organized as follows. In Section 2 we review related work as it pertains to object detection and contextual visual information. Section 3 introduces our proposed approach for context-aware object detection. In Section 4 we report on empirical results and Section 5 concludes the paper and points towards future work.

II. Related Works

Context in images can be used in a multitude of approaches. In [3], the authors made use of object relationships by implementing a spatial context-based solution that can group co-occurring objects in the scene. Objects are clustered into groups based on frequency of co-occurrence and, at a negligible drop in accuracy, have increased the efficiency of detection models as it facilitated quicker predictions about the presence of other objects that are known to occur based on these clustered groups.

Object layouts in relation with each other in an image is another type of useful information that can be captured with the help of graph networks because of their flexibility to describe pair-wise relations in space. These graph networks leverage spatial and semantic information to produce interpretable graph structures that model spatial relations between objects to improve classification and localization [4].

Context rescoring in post-processing assigns a new confidence score to each bounding box, by considering confidence scores of all the other bounding boxes taking into account the location, sizes of boxes, and co-occurrences of objects [9]. Fig. 1 provides an overview example regarding the occurrence of pairs of object classes in the COCO 2017 dataset, that exhibit both strong and weak co-occurrence relationships. Strong co-occurring relationships between objects is an important form of contextual cue. For instance, classes having a low context score could be attributed to low probability in co-occurrence, among other things. Rescoring works to improves performance by assigning a new higher confidence score to true positives rather than false negatives so that those detections with correct class and better bounding box coordinates predictions survive longer in the algorithm than detections with relatively poor predictions. The discussed model displayed an average precision improvement of 0.5-1 over strong region-based detectors on MS COCO 2017 [13]. Though a modest improvement, the model showed consistent improvements in accuracy by maintaining confidence of correct detections and decreasing it for out of context and duplicate detections.

However, it should be mentioned that not all contextual information is useful, as the incorporation of irrelevant contextual noise can actually hinder performance [8]. Therefore, it should be seen that the identification of helpful contextual information is the first step to creating a well performing model. Previous works have explored object relationships, such as co-occurring objects or spatial layouts [9, 3, 4], however in this work we take a different approach by considering scene-level information. By adjusting the loss values of our model based on integrating scene predictions, the model adapts to the contextual relevance of objects within the scene. This adaptability enables our model to recognize objects based on their contextual appropriateness, leading to improved performance in object detection. On the one hand, we can think about Model 1 as integrating context in an implicit fashion. On the other hand, Model 2 makes use of contextual information explicitly, by utilizing scene type information as an additional input for the object detection task.

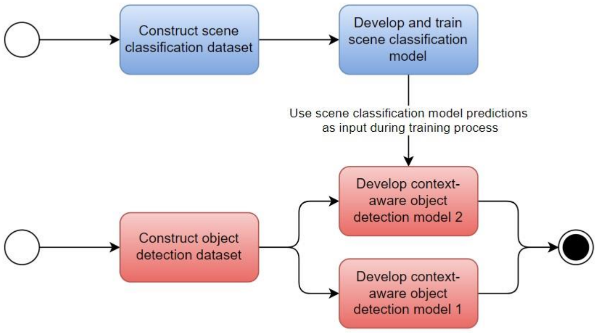
Construct scene classification dataset
Develop and train scene classification model
Use scene classification model predictions as input during training process
Construct object detection dataset
Develop context-aware object detection model 2
Develop context-aware object detection model 1

Fig. 2. Workflow diagram

III. Method

A. Approach Overview

The proposed approach with respect to the problem of developing a contextually aware system for object detection is to first implement an object detection algorithm, then choose a way to represent contextual information in images, and ultimately, consolidate that information into the training process of the model in a way that would ideally improve its performance. Consequently, the choice of machine learning framework used to develop the model had to be one which allowed for increased flexibility and control over the architecture and training process and which facilitates customization of deep learning models at a lower level. For this purpose, the Keras Functional API [11] was used to develop the deep learning models. As for the choice of contextual information used, ‘scene information’ from images was considered for this study. By analysing the scene or background of an image, a model can gain a better understanding of the likelihood of objects being present in an image and their spatial relationships with one another, as well as with their background. Additionally, scene information can provide semantic cues to improve the robustness of object detection models to changes in lighting, background, and other environmental factors because it can help the model to differentiate between objects and their surroundings [5]. In order to use this information in the training process, predictions on image scenes were used as one of the components contributing to the computation of the total loss function during the training of the model.

B. Integrating Contextual Information

The models developed use two different approaches to incorporate context-awareness by adjusting the loss functions of the object detection models based on scene predictions; Context-Aware Model 1 (referred to henceforth as *Model 1*) is trained for a multi-task objective, both scene classification and object detection. Alternatively, Context-Aware Model 2 (referred to henceforth as *Model 2*) uses the predictions from a supplementary scene classification model as additional input during the training phase.

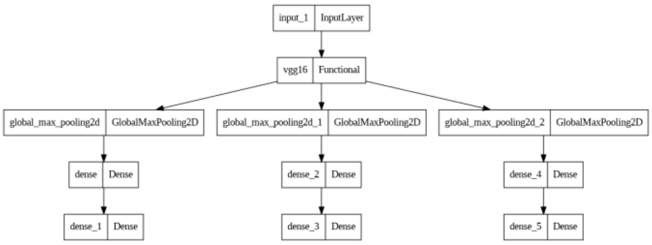
input_1 | InputLayer
vgg16 | Functional
global_max_pooling2d | GlobalMaxPooling2D
global_max_pooling2d_1 | GlobalMaxPooling2D
global_max_pooling2d_2 | GlobalMaxPooling2D
dense | Dense
dense_2 | Dense
dense_4 | Dense
dense_1 | Dense
dense_3 | Dense
dense_5 | Dense

Fig. 3. Context-Aware Model 1 architecture

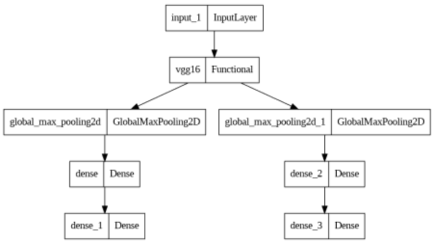

Fig. 4. Context-Aware Model 2 architecture

The workflow process is depicted in Fig. 2 and the architecture plots of Models 1 and 2 are depicted in Fig. 3 and Fig. 4 respectively. Transfer learning was used in the development of all models, the VGG16 model [12] was used as the backbone for the object detection models, whereas the ResNet50 model [12] was used as the backbone for the scene recognition model.

The rationale for Model 1 is that by altering the loss function values, there is a change in the gradients that are computed during backpropagation and these gradients are used to update the weights of the model during training. Since a term is included in the loss function that penalizes the model for making incorrect predictions about image scene information, the model can take into account the scene prediction error and adjust the weights of the model to improve scene interpretation accuracy. By adjusting the loss function in this way, the model is encouraged to learn useful features for both image context/scene comprehension and object detection, thus incorporating contextual information from the scene aiming to result in a more robust and accurate model overall.

IV. EMPRIRICAL RESULTS

The first part of this section details the generation of the datasets used for training and testing, as well as the evaluation metrics used to measure performance of the developed models. The second part discusses the results of the experiments. The implementation was run on a hardware configuration consisting of an i7-8565U CPU with 16GB RAM. The code is written in Python 3 and executed on Jupyter Notebooks within an Anaconda environment.

A. *Datasets and Evaluation Metrics*

Two datasets were constructed, one for training a scene classification model and another that was used for the training and testing of the object detection models. The decision was made to manually construct the datasets. Although creating a dataset from scratch is a time-consuming and laborious process, it provides complete control over the data and ultimately leads to a more precise model and

Fig. 5. Examples of images in the Scene Classification dataset

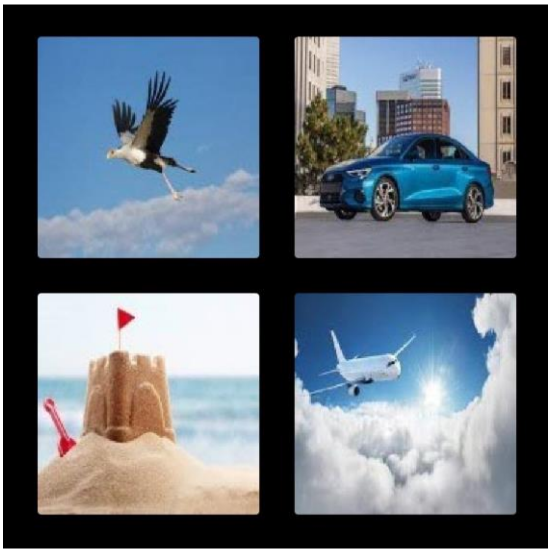

Fig. 6. Examples of images in the Object Detection dataset

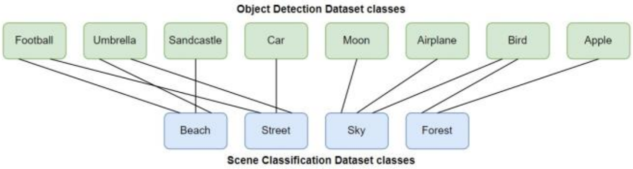

Fig. 7. Relationship between object and scene classes

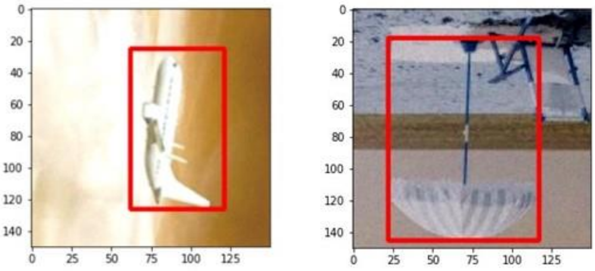

Fig. 8. Examples of Augmented images with bbox annotations

TABLE I. METRICS OBTAINED BY ALL MODELS

Models	Accuracy	Mean IoU	Precision	Recall	F1-score
Context-Aware Model 1	71.30%	0.66	0.76	0.69	0.72
Context-Aware Model 2	**75.14%**	**0.69**	0.76	**0.74**	**0.75**
Benchmark Object Detection Model	73.43%	0.67	**0.78**	0.72	**0.75**

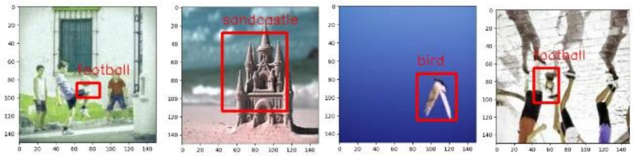

Fig. 9. Sample of class and bbox predictions made by Model 2

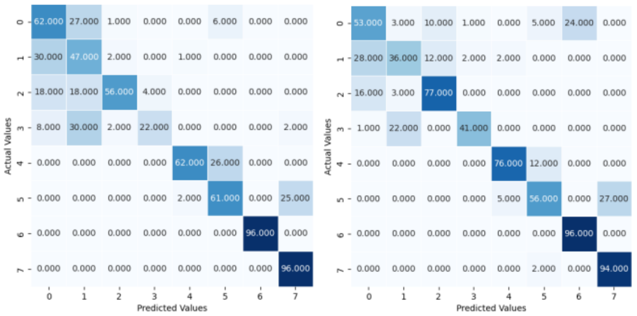

Fig. 10. Confusion matrix for Model 1 (*left*) and Model 2 (*right*). (*axes represent the 8 object classes in the dataset*)

evaluation. The images for the constructed dataset were carefully collected from Google Images to ensure a good quality dataset was considered.

The scene dataset consists of images divided into four scene classes, namely '*beach*', '*forest*', '*sky*', and '*street*'. The scenes were chosen based on where the object classes from the object detection dataset were most likely to appear in images. For example, the scene classes of 'beach' and 'street' correspond to the scenes where the object class of 'football' from the object detection dataset is likely to appear. This was done to help the model learn scene information in images that could be used to improve the performance of the context-aware object detection model. Sample images from the scene classification dataset are shown in Fig 5.

Images for the object detection dataset were chosen in such a way that objects from the object classes may appear in at least two scenes from the scene detection dataset. The object detection dataset comprises of 8 classes, namely '*car*', '*football*', '*apple*', '*umbrella*', '*bird*', '*moon*', '*sandcastle*', and '*airplane*'. The datasets were designed with an overlap between objects and scenes to prevent a one-to-one relationship connecting them. Hence, when the object detection dataset is used to train the context-aware models, the scene input to the model could provide extra information about what objects could be recognized.

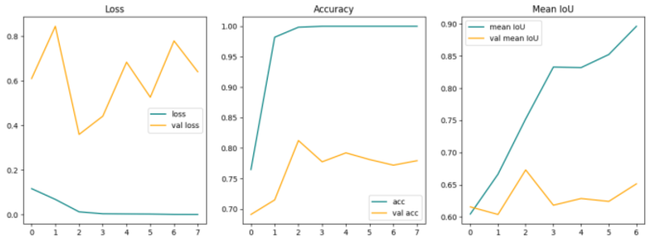

Fig. 11. Workflow diagram

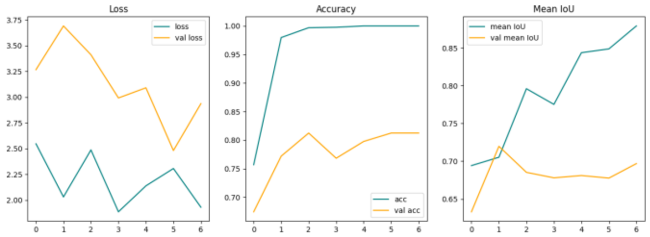

Fig. 12. Workflow diagram

Additionally, having a many-to-many relationship between the objects and scenes inhibited reducing the problem to merely scene recognition, which would have already been provided as input (in the case of *Model 2*). Instead, this configuration should help the model learn object-scene relationships and thereby improve its performance. Sample images from the object detection dataset are shown in Fig. 6 and the relationship between the two datasets is illustrated in Fig. 7.

The images were pre-processed by resizing to 150x150 pixels, filtered for incompatible file types, and normalized to facilitate faster convergence during training. Since the original dataset of manually collected images was rather small (180 images), the data was augmented to increase the size of the dataset and help reduce overfitting. The bounding box and class annotations of the original images were done manually, after which an image augmentation library [10] was used to implement a variety of image transform operations and automatically create new and accurate bounding boxes for the objects in the augmented images. Examples of the augmented images in the dataset are shown in Fig. 8 with the automatically constructed bounding box drawn over it. This process effectively increased the overall size of the dataset to ~2500 images that were then distributed into the training, validation, and testing sets using a 70:30 split.

The two aspects of object detection can be evaluated for each model based on the following metrics: accuracy, precision, and recall for object classification, and bounding box Intersection over Union (IoU) for object localization. IoU is defined as the ratio between the area of overlap between the ground truth (annotated bounding box in training dataset) and the predicted truth (bounding box output of the model) and the area of union between them.

B. Results Analysis

The performance of all models, including the benchmark base object detection model, without any context-aware capabilities, is described in Table I in order to gain a comprehensive understanding of how well the context-aware modifications work, and how beneficial or counter-productive they are.

The training of both models converged after 7 epochs using a batch size of 32. From Table I, it can be observed that Model 2 has the best performance across most of the metrics and both models performed reasonably well for the object classification task, although the IoU metrics suggest that there is room for improvement in the object localization aspect. The loss curves for both models are rather irregular which could be attributed to the large variation in images in the relatively small dataset. However, the loss trend for Model 2 was ultimately converging as expected, in contrast to the loss trend of Model 1 which is rather erratic. Model 1 obtained an accuracy and mean IoU lower than both Model 2 and the standard object detection model, which is somewhat surprising. This result demonstrated that the method used to incorporate contextual information in Model 1 by training the model to learn scene information in addition to object classification and localization actually deteriorated its performance. Potential reasons for this could be that the scene classification task led to a more complex optimization problem due to the additional noise in the training process, which may have caused the model to struggle to perform multiple tasks leading to subpar predictions. However, Model 2 obtained an accuracy improvement of 5.39% and a mean IoU increase of 4.55% from Model 1 and outperformed the standard object detection model as well, with the highest accuracy of 75.14% and highest mean IoU of 0.69. This result was rather interesting since the technique used in Model 2 had the opposite effect from Model 1 wherein the context-awareness hindered its performance. It showed that the approach used to integrate the contextual information can lead to significant improvements in determining the performance of a model. The use of a pre-trained external classification model to make intelligent scene predictions proved to be more constructive than training the base model itself for multiple tasks simultaneously.

Object detection predictions made by Model 2 on images in the testing set are shown in Fig. 9. The confusion matrix showing classification performance of Model 1 and 2 is depicted in Fig. 10. As can be observed, both models perform quite well in terms of avoiding misclassification of object, however Model 2 outperforms Model 1. The values of precision and recall metrics for each class described in Table I are the macro average values. The macro average calculates the average performance of the model by treating each class equally, regardless of class size or imbalance. This means that the values for each class from the confusion matrix contributes equally to the final score, regardless of the number of instances, so as to account for the minor class distribution imbalance in the dataset. The loss, accuracy and mean IoU performance trends are displayed in Fig. 11 for Model 1 and Fig. 12 for Model 2. The accuracy, mean IoU and recall show a clear performance improvement for Model 2 in comparison to the benchmarking model, which does not include any contextual information.

V. Conclusion

This paper proposed two implementations of context-aware object detection models using scene information from images. Context-Aware Model 1 was trained in a multi-task fashion, to make predictions on image scenes in addition to object detection, whereas Context-Aware Model 2 used the predictions from a supplementary scene classification model as additional input during its training phase. The evaluation of the results obtained by Model 2 compared to the benchmark object detection model proved that the appropriate implementation of context-aware functionality can, in fact, be effective in moderately improving its performance. The experiment results lead to the conclusion that contextual knowledge from images, when used correctly, is a valuable source of information that can be used to improve object detection.

In future work we aim to also investigate other sources of contextual information besides scene type, as well as other computer vision tasks and the role of context-aware models in improving their performance.

References

[1] H. Hock, G. P. Gordon, and R. Whitehurst. “Contextual relations: The influence of familiarity, physical plausibility, and belongingness,” Perception & Psychophysics, vol. 16, pp. 4-8, Feb. 1974.

[2] J. Peng, H. Wang, S. Yue, Z. Zhang, “Context-aware co-supervision for accurate object detection,” Pattern Recognition, vol. 121, Jan. 2022.

[3] M. Naseem, S. Reda, “AdaCon: Adaptive Context-Aware Object Detection for ResourceConstrained Embedded Devices.” 2021 IEEE/ACM International Conference on Computer-Aided Design, Aug. 2021.

[4] H. Xu, C. Jiang, X. Liang, Z. Li, “Spatial-aware Graph Relation Network for Large-scale Object Detection.” 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Jun. 2019.

[5] H. Sun, Z. Meng, P. Y. Tao and M. H. Ang, "Scene Recognition and Object Detection in a Unified Convolutional Neural Network on a Mobile Manipulator," 2018 IEEE International Conference on Robotics and Automation (ICRA), Sep. 2018.

[6] J. Li et al., "Attentive Contexts for Object Detection," IEEE Transactions on Multimedia, vol. 19, no. 5, pp. 944-954, May 2017.

[7] R. Mottaghi et al., "The Role of Context for Object Detection and Semantic Segmentation in the Wild," 2014 IEEE Conference on Computer Vision and Pattern Recognition, pp. 891-898, Jun. 2014.

[8] A. Shrivastava and A. Gupta, “Contextual Priming and Feedback for Faster R-CNN,” Carnegie Mellon University, Jun. 2017.

[9] P.F. Felzenszwalb, R.B. Girshick, D. McAllester and D. Ramanan, "Object Detection with Discriminatively Trained Part-Based Models," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, no. 9, pp. 1627-1645, Sept. 2010.

[10] Albumentations. [Online]. Available: https://albumentations.ai/

[11] Keras. (2019). The Functional API [Online]. Available: https://keras.io/guides/functional_api/

[12] Keras.io. Keras Applications. https://keras.io/api/applications/

[13] L. V. Pato, R. Negrinho and P. M. Q. Aguiar, "Seeing without Looking: Contextual Rescoring of Object Detections for AP Maximization," 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 14598-14606, Mar. 2020.

[14] S.K. Divvala, D. Hoiem, J.H. Hays, A.A Efros and M. Hebert “An empirical study of context in object detection,” IEEE Conference on computer vision and Pattern Recognition, pp. 1271-1278, Jun. 2009.