Study and Implementation of the state-of-the-art object detection models on DOTA

A Project Report

Submitted by

NIKILESH B

in partial fulfillment of the requirements

for the award of the degree of

DUAL DEGREE

(BACHELOR AND MASTER OF TECHNOLOGY)

DEPARTMENT OF ELECTRICAL ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY MADRAS

MAY 2022

THESIS CERTIFICATE

This is to certify that the thesis titled Study and Implementation of the state-of-the-art object detection models on DOTA, submitted by Nikilesh B, to the Indian Institute of Technology, Madras, for the award of the degree of Dual degree (B.Tech and M.Tech), is a bona fide record of the research work done by him under our supervision. The contents of this thesis, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Guide :
Dr. Ganpathy Krishnamurthi
Associate Professor
Dept. of Engineering Design
IIT Madras, 600 036

Co Guide :
Dr. Ananth krishnan
Associate Professor
Dept. of Electrical Engineering
IIT Madras, 600 036

Place: Chennai
Date: 25/05/202

ACKNOWLEDGEMENTS

At the outset, I express my deep sense of gratitude to my Project Guide Prof. Dr. Ganapathy Krishnamurthi for his guidance, support, encouragement and help throughout the period of the project work. I am highly indebted to him for devoting his valuable time to help me complete the work in time. I would also like to thank Dr. Ananth Krishnan for being my co-guide in this project. Also, I would like to thank the Department of Engineering Design and Department of Electrical Engineering, IIT Madras for providing me the opportunity to work in this challenging field.

ABSTRACT

KEYWORDS: DOTA

Object detection is an important and challenging problem in computer vision. Even though the recent years have witnessed major breakthroughs in object detection in natural scenes, such successes have been slow to aerial imagery. The reasons are not only because of the huge variation in the scale, orientation and shape of the object instances on the earth's surface, but also due to the scarcity of well annotated datasets of objects in aerial scenes

.

DOTA is a large-scale dataset for object detection in aerial images. It can be used to develop and evaluate object detectors in aerial images. The images are collected from different sensors and platforms. Each image is of the size in the range from

No table of figures entries found.**800 × 800** to **20,000 × 20,000** pixels and contains objects exhibiting a wide variety of scales, orientations, and shapes. This article focuses on state of art algorithms and objects detection models and implementing them on the DOTA and as well as simulating more real life dataset using DOTA as base and studying the effect on the model.

Contents

ACKNOWLEDGEMENTS 3

ABSTRACT 4

Table of Images 7

ABBREVIATIONS 7

CHAPTER 1 8

1.1 Introduction 8

CHAPTER 2 10

DOTA - Dataset for Object deTection in Aerial images 10

2.1 Overview 10

2.2 Motivation 10

2.3 DOTA advantages 12

CHAPTER 3 13

Dynamic Head: Unifying Object Detection Heads with Attentions. 13

3.1 Overview 13

3.2 Motivation 13

3.3 New Ideas 14

CHAPTER 4 16

Towards Transformer-Based Object Detection 16

4.1 Overview 16

4.2 Motivation 16

4.3 Advantages 17

4.4 New results 17

CHAPTER 5 19

Implementation of the model 19

5.1 Overview 19

5.2 Faster R-CNN for object detection 19

5.3 Architecture 20

5.4 Implementation and results 21

CHAPTER 6 22
Simulation of data and testing 22
6.1 Overview 22
6.2 Simulation of Motion Blur 22
6.2.1 Results 23
6.3 Simulation of Atmospheric turbulence Blur 24
6.3.1 Limitations 25
6.3.2 Implementation and results 25
6.4 Simulation of noise 26
6.4.1 Results 27
6.5 Simulation of Gaussian Blur 28

CHAPTER 7 29
Retraining the Model 29
7.1 Overview 29
7.2 Motion Blur 29
7.3 Atmospheric Turbulence Blur 31
7.4 Noisy images 31
7.4 Gaussian Blur 32

CHAPTER 8 33
8.1 CONCLUSION 33
8.2 Future possibilities 33

REFERENCES 34

Code and Supplement Data 35

Table of Images

Figure 1: An example taken from DOTA..... 11

Figure 2 : An illustration of the above mentioned Dynamic Head approach. 15

Figure 3 : An illustration of the simulation of motion blur 22

Figure 4 : An illustration of the simulation of motion blur 24

Figure 5: An illustration of the simulation of noisy..... 26

Figure 6: An illustration of the simulation of Gaussian blur 28

ABBREVIATIONS

DOTA	Large-scale dataset for object detection in aerial images
MSCOCO	Microsoft Common Objects in Context
OBB	oriented bounding boxes
NLP	Natural Language Processing
ViT	Vision Transformer
FRCNN	Faster RCNN
RoI	Region of Interest
AP	Average Precision
mAP	Mean Average Precision

CHAPTER 1

1.1 Introduction

Object detection in Earth Vision refers to identifying objects of interest (e.g., vehicles, airplanes) on the earth's surface and classifying their categories. In many this is different from conventional object detection datasets, where objects are generally oriented upward because of the gravity. Also the object instances in aerial images often appear with arbitrary orientations, as illustrated in Fig. 1, depending on the perspective of the Earth Vision platforms.

Various studies have been conducted in the field of object detection in aerial images drawing upon recent advances in Computer Vision and accounting for the high demands of Earth Vision applications. Most of these methods attempt to transfer object detection algorithms developed for natural scenes to the aerial image domain. Recently, driven by the successes of deep learning-based algorithms for object detection, Earth Vision researchers have pursued approaches based on fine-tuning networks pre-trained on large-scale image datasets (ImageNet and MSCOCO) for detection in the aerial domain. While such fine-tuning based approaches are a reasonable avenue to explore, images such as Figure 1 reveals that the task of object detection in aerial images is different from the conventional object detection task in the following aspects: -

- The scale variations of object instances in aerial images are huge. This is not only because of the spatial resolutions of sensors, but also due to the size variations inside the same object category.
- Many small object instances are crowded in aerial images, for example, the ships in a harbor and the vehicles in a parking lot, as illustrated in Fig. 1. Moreover, the frequencies of instances in aerial images are unbalanced.

- Objects in aerial images often appear in arbitrary orientations. There are also some instances with an extremely large aspect ratio, such as a bridge

Besides these distinct difficulties, the studies of object detection in Earth Vision are also challenged by the well-known dataset bias problem i.e. the degree of generalizability across datasets is often low. In order to alleviate such biases, the dataset should be annotated to reflect the demands of real world applications.

Following this chapter the report is organized as follows: chapter 2 introduces why DOTA as a dataset is necessary and how it is different and hence how it will be useful. Chapter 3 talks about the Dynamic Head: Unifying Object Detection Heads with Attentions, idea of model. Chapter 4 talks about Towards Transformer-Based Object Detection, another useful insight for object detection algorithms. Chapter 5 talks about the state of the art model implemented on the DOTA and the results. Chapter 6 talks about data augmentation and how fine tuning helps improve the model. Chapter 7 talks about future scopes.

CHAPTER 2

DOTA - Dataset for Object deTection in Aerial images

2.1 Overview

As discussed earlier to advance the object detection research in Earth Vision, a large-scale Dataset for Object deTection in Aerial images (DOTA) was created. Over 2806 aerial images from different sensors and platforms were collected with crowd sourcing. Each image is of the size about 4000 × 4000 pixels and contains objects of different scales, orientations and shapes. These DOTA images are annotated by experts in aerial image interpretation, with respect to 15 common object categories. The fully annotated DOTA dataset contains 188,282 instances, each of which is labeled by an OBBs, instead of an axis-aligned one, as is typically used for object annotation in natural scenes.

2.2 Motivation

Datasets have played an important role in data-driven research in recent years. Large datasets like MSCOCO are instrumental in promoting object detection and image captioning research. When it comes to the classification task and scene recognition task, the same is true for ImageNet and Places, respectively. However, in aerial object detection, a dataset resembling MSCOCO and ImageNet both in terms of image number and detailed annotations has been missing, which becomes one of the main obstacles to the research in Earth Vision, especially for developing deep learning-based algorithms.

Aerial object detection is extremely helpful for vehicle counting, remote object tracking and unmanned driving. Therefore, a large-scale and challenging aerial object detection benchmark, being as close as possible to real-world applications, is imperative for promoting research in this field.

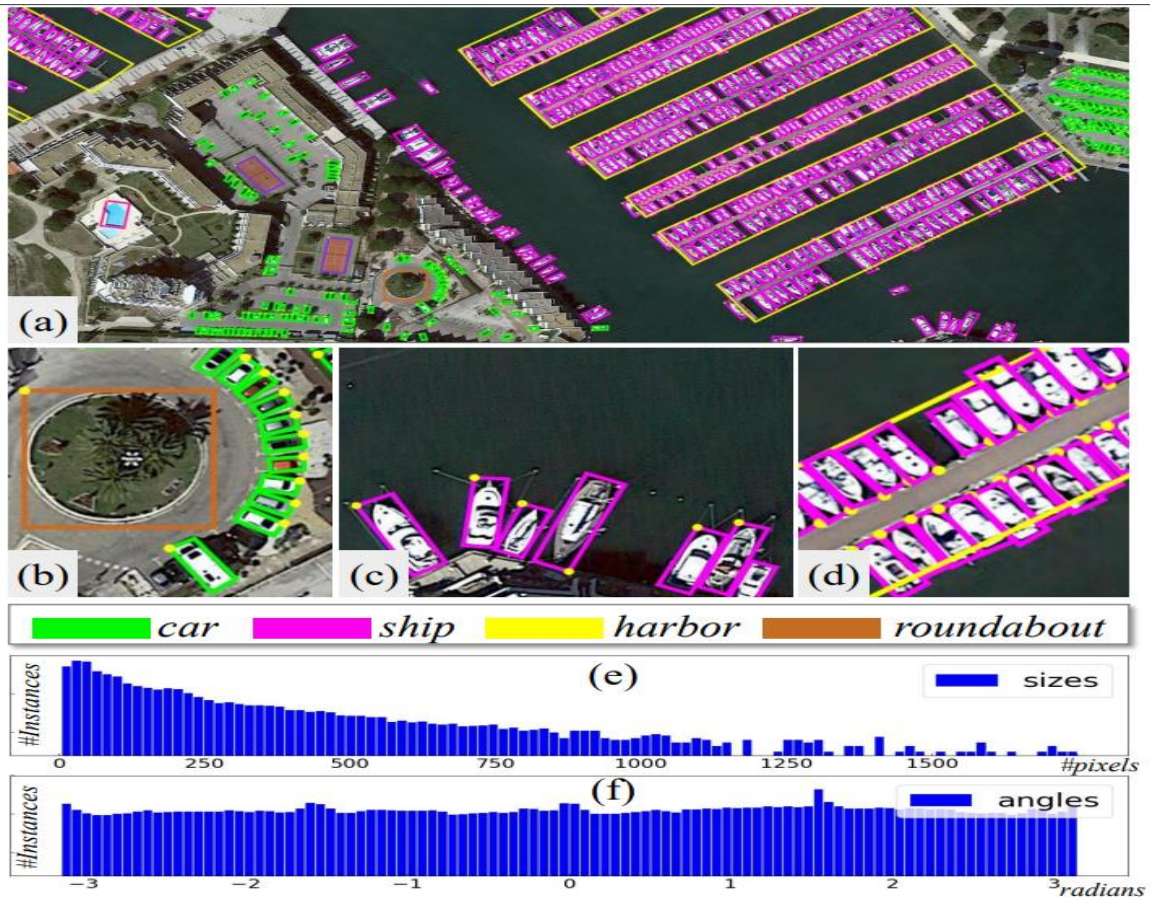


Figure 1: An example taken from DOTA.

(a) Typical image in DOTA consisting of many instances across multiple categories.

(b) Illustration of the variety in instance orientation and size.

(c),(d) Illustration of sparse instances and crowded instances, respectively.

Here it is shown four out of fifteen of the possible categories in DOTA. Examples shown in (b),(c),(d) are cropped from source image (a). The histograms (e),(f) exhibit the distribution of instances with respect to size and orientation in DOTA

2.3 DOTA advantages

One could argue that a good aerial image dataset should possess four properties, namely,

1) A large number of images,

2) Many instances per categories,

3) Properly oriented object annotation, and

4) Many different classes of objects, which make it approach to real-world applications

.

Dataset	Annotation way	#main categories	#Instances	#Images	Image width
NWPU VHR-10 [2]	horizontal BB	10	3651	800	∼1000
SZTAKI-INRIA [1]	oriented BB	1	665	9	∼800
TAS [9]	horizontal BB	1	1319	30	792
COWC [20]	one dot	1	32716	53	2000∼19,000
VEDAI [24]	oriented BB	3	2950	1268	512, 1024
UCAS-AOD [39]	oriented BB	2	14,596	1510	∼1000
HRSC2016 [17]	oriented BB	1	2976	1061	∼1100
3K Vehicle Detection [15]	oriented BB	2	14,235	20	5616
DOTA	oriented BB	**14**	**188,282**	**2806**	800∼4000

Table 1 : Comparison among DOTA and object detection datasets in aerial images.

Besides, what makes DOTA unique among the above mentioned large-scale general object detection benchmarks is that the objects in DOTA are annotated with properly oriented bounding boxes (OBB for short). OBB can better enclose the objects and differentiate crowded objects from each other

.

CHAPTER 3

Dynamic Head: Unifying Object Detection Heads with Attentions.

3.1 Overview

The complex nature of combining localization and classification in object detection has resulted in the flourished development of methods. Previous works tried to improve the performance in various object detection heads but failed to present a unified view. Here a novel dynamic head framework to unify object detection heads with attentions is studied. By coherently combining multiple self-attention mechanisms between feature levels for scale awareness, among spatial locations for spatial-awareness, and within output channels for task-awareness, the proposed approach significantly improves the representation ability of object detection heads without any computational overhead.

3.2 Motivation

Object detection is to answer the question “what objects are located at where” in computer vision applications. In the deep learning era, nearly all modern object detectors share the same paradigm – a backbone for feature extraction and a head for localization and classification tasks. How to improve the performance of an object detection head has become a critical problem in existing object detection works .The challenges in developing a good object detection head can be summarized into three categories.

- The head should be scale-aware, since multiple objects with vastly distinct scales often co-exist in an image.
- The head should be spatial-aware, since objects usually appear in vastly different shapes, rotations, and locations under different viewpoints.

- Thirdly, the head needs to be task aware, since objects can have various representations (e.g., bounding box, center, and corner points) that own totally different objectives and constraints.

Majority of the recent studies only focus on solving one of the aforementioned problems in various ways. It remains an open problem how to develop a unified head that can address all these problems simultaneously.

3.3 New Ideas

A novel detection head is proposed, called dynamic head, to unify scale-awareness, spatial-awareness, and task-awareness all together. If we consider the output of a backbone (i.e., the input to a detection head) as a 3-dimensional tensor with dimensions level × space × channel, we discover that such a unified head can be regarded as an attention learning problem. An intuitive solution is to build a full self-attention mechanism over this tensor. However, the optimization problem was found to be too difficult to solve and the computational cost is not affordable.

Instead, attention mechanisms were deployed separately on each particular dimension of features, i.e., level-wise, spatial-wise, and channel-wise.

- The scale-aware attention module is only deployed on the dimension of level. It learns the relative importance of various semantic levels to enhance the feature at a proper level for an individual object based on its scale.

- The spatial-aware attention module is deployed on the dimension of space (i.e., height × width). It learns coherently discriminative representations in spatial locations.

- The task-aware attention module is deployed on channels. It directs different feature channels to favor different tasks separately (e.g., classification, box regression, and center/key-point learning.

In this way, a unified attention mechanism is explicitly implanted for the detection head. Although these attention mechanisms are separately applied on different dimensions of a feature tensor, their performance can complement each other.

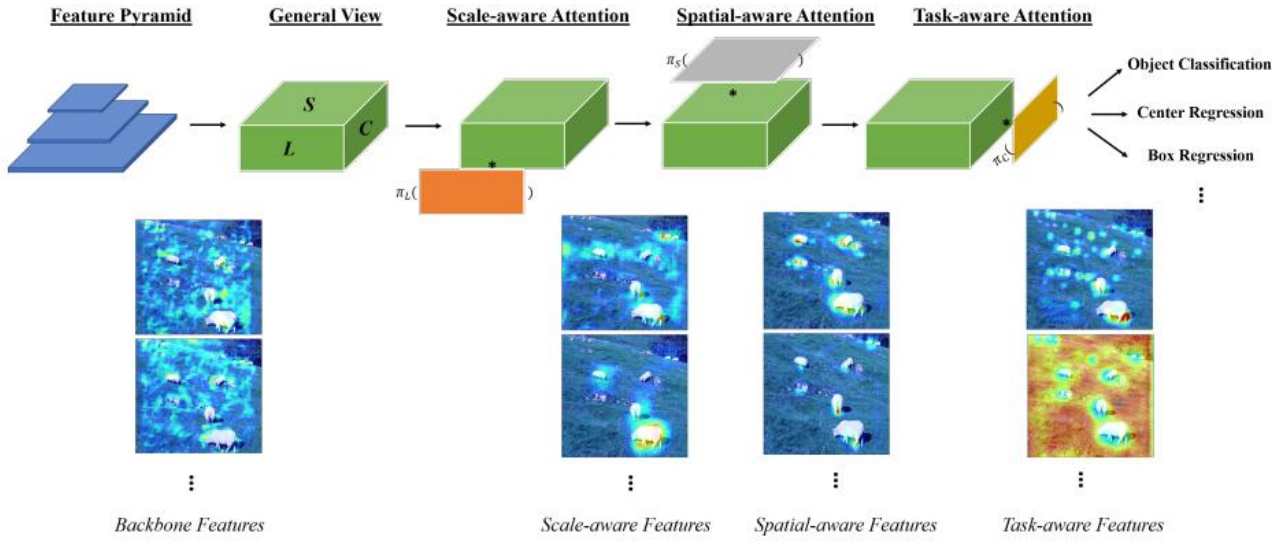


Figure 2 : An illustration of the above mentioned Dynamic Head approach.

It contains three different attention mechanisms, each focusing on a different perspective: scale-aware attention, spatial-aware attention, and task-aware attention. It is also visualized how the feature maps are improved after each attention module.

CHAPTER 4

Towards Transformer-Based Object Detection

4.1 Overview

Transformers have become the dominant model in NLPs , owing to their ability to pre-train on massive amounts of data, then transfer to smaller, more specific tasks via fine-tuning. The Vision Transformer was the first major attempt to apply a pure transformer model directly to images as input, demonstrating that as compared to convolutional networks, transformer-based architectures can achieve competitive results on benchmark classification tasks. However, the computational complexity of the attention operator means that we are limited to low-resolution inputs. For more complex tasks such as detection or segmentation, maintaining a high input resolution is crucial to ensure that models can properly identify and reflect fine details in their output.

This naturally raises the question of whether or not transformer-based architectures such as the Vision Transformer are capable of performing tasks other than classification. But it has been shown that Vision Transformers can be used as a backbone by a common detection task head to produce competitive COCO results.

4.2 Motivation

The Transformer model has become the preferred solution for a wide range of natural language processing (NLP) tasks, showing impressive progress in machine translation, question answering, text classification, document summarization, and more. Part of this success comes from the Transformer's ability to learn complex dependencies between input sequences via self-attention, and its scalability that makes it possible to pre-train models of remarkable size on large datasets with no signs of saturating performance.

4.3 Advantages

The Vision Transformer (ViT) demonstrated for the first time that a transformer architecture can be directly applied to images as well, by treating an image as a sequence of patches. Although its performance on mid-sized datasets trails behind convolution-based models, the ViT seems to retain the capacity seen in NLP transformers, enabling it to pre-train on an unprecedented amount of data. In effect, ViT suggests that the standard convolution, which has been the hallmark of vision modeling for decades, maybe supplemented or replaced by attention-based components.

Transformers are capable of globally attending at every layer of the network, potentially making the spatial correspondence between the input and intermediate features weaker. This naturally raises the question of whether or not Vision Transformers can be fine-tuned to perform tasks that are more locally-sensitive, such as object detection or segmentation.

4.4 New results

A new model, ViT-FRCNN was developed, that attempts to answer the above question by augmenting a ViT with detection specific task heads to detect and localize objects in images. Importantly, ViT-FRCNN demonstrates that a transformer based backbone can retain sufficient spatial information for object detection. It is also shown that ViT-FRCNN achieves competitive results on the COCO detection challenge, while exhibiting many of the desirable properties of transformer based models. In particular, some experiments which were done suggested that object detection tasks benefit from the massive pre training paradigm commonly used with transformers.

Experiments also find improved detection performance on large objects (perhaps due to the ability of the architecture to attend globally), and fewer spurious over detections of objects.

It is also believed that ViT-FRCNN shows that the commonly applied paradigm of large scale pre training on massive datasets followed by rapid fine-tuning to specific tasks can be scaled up even further in the field of computer vision, owing to the model capacity observed in transformer-based architectures and the flexible features learned in such backbones.

CHAPTER 5

Implementation of the model

5.1 Overview

method	speed (fps)	DOTA-v1.0		DOTA-v1.5		DOTA-v2.0	
		HBB mAP	OBB mAP	HBB mAP	OBB mAP	HBB mAP	OBB mAP
RetinaNet	16.7	67.45	-	61.64	-	49.31	-
RetinaNet OBB	12.1	69.05	66.28	62.49	59.16	49.26	46.68
Mask R-CNN	9.7	71.61	70.71	64.54	62.67	51.16	49.47
Cascade Mask R-CNN	7.2	71.36	70.96	64.31	63.41	50.98	50.04
Hybrid Task Cascade*	7.9	72.49	71.21	64.47	63.40	50.88	50.34
Faster R-CNN	14.3	70.76	-	64.16	-	50.71	-
Faster R-CNN OBB	14.1	71.91	69.36	63.85	62.00	49.37	47.31
Faster R-CNN OBB + Dpool	12.1	71.83	70.14	63.67	62.20	50.48	48.77
Faster R-CNN H-OBB	13.7	70.37	70.11	64.43	62.57	50.38	48.90
Faster R-CNN OBB + RoI Transformer	12.4	**74.59**	**73.76**	**66.09**	**65.03**	**53.37**	**52.81**

Table 2: The baseline results observed on DOTA (R-FPN-50, without data augmentations) . We can see from the table that the best results are with Faster R-CNN OBB with RoI Transformer.

5.2 Faster R-CNN for object detection

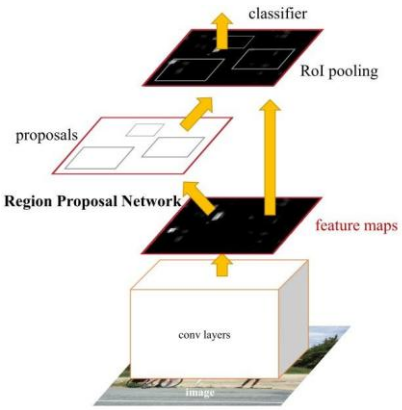
A Faster R-CNN object detection network is composed of a feature extraction network which is typically a pre-trained CNN, similar to what had been used for its predecessor. This is then followed by two sub networks which are trainable. The first is a Region Proposal Network (RPN), which is, as its name suggests, used to generate object proposals and the second is used to predict the actual class of the object. So the primary differentiator for Faster R-CNN is the RPN which is inserted after the last convolutional layer. This is trained to produce region proposals directly without the need for any external mechanism like Selective Search. After this ROI pooling is used and an upstream classifier and bounding box regressor similar to Fast R-CNN

..

5.3 Architecture

The architecture of Faster R-CNN is shown in the next figure. It consists of 2 modules:

1. **RPN**: For generating region proposals.
2. **Fast R-CNN**: For detecting objects in the proposed regions.



The RPN module is responsible for generating region proposals. It applies the concept of attention in neural networks, so it guides the Fast R-CNN detection module to where to look for objects in the image.

We can see how the convolutional layers (e.g. computations) are shared across both the RPN and the Fast R-CNN modules.

The Faster R-CNN works as follows:

- The RPN generates region proposals.
- For all region proposals in the image, a fixed-length feature vector is extracted from each region using the ROI Pooling layer [2]
- The extracted feature vectors are then classified using the Fast R-CNN.
- The class scores of the detected objects in addition to their bounding-boxes are returned.

5.4 Implementation and results

The state of the art model with which the best results were observed was implemented. The configuration file was loaded and the validation dataset from the DOTA website was loaded and model was run to test this validation data.

classname: plane
ap: 0.9014526645635133

classname: baseball-diamond
ap: 0.8756156060304421

classname: bridge
ap: 0.7358691439350338

classname: ground-track-field
ap: 0.8072462286767592

classname: small-vehicle
ap: 0.7476489526033117

classname: large-vehicle
ap: 0.8886002316343158

classname: ship
ap: 0.8868232500621034

classname: tennis-court
ap: 0.9059249633791323

classname: basketball-court
ap: 0.8715753581872354

classname: storage-tank
ap: 0.9014873059340347

classname: soccer-ball-field
ap: 0.759248194164202

classname: roundabout
ap: 0.8570194710661244

classname: harbor
ap: 0.8796535504239384

classname: swimming-pool
ap: 0.8113566147588557

classname: helicopter
ap: 0.7954980842911878

map: 0.8416679746473459

classaps: [90.14526646 87.5615606 73.58691439 80.72462287 74.76489526 88.86002316
88.68232501 90.59249634 87.15753582 90.14873059 75.92481942 85.70194711
87.96535504 81.13566148 79.54980843]

CHAPTER 6

Simulation of data and testing

6.1 Overview

To test how the model performs with more real life like data, few datasets were simulated. Some of the type of simulation which was done was:

- Simulating Motion Blur images
- Simulating Atmospheric turbulence images
- Simulating Noisy images
- Simulating Gaussian Blurring images

6.2 Simulation of Motion Blur

Validation dataset was used to the simulate Motion blur to the images of that dataset, and then tested with the model. At first the kernel was created and then the kernel was applied to the input image using filter2D module available in cv2.

(a) (b)

Figure 3 : An illustration of the simulation of motion blur

(a) Normal image (b) Motion blurred image

6.2.1 Results

classname: plane

ap: 0.6924784381687126

classname: baseball-diamond

ap: 0.5934263509479027

classname: bridge

ap: 0.1502267573696145

classname: ground-track-field

ap: 0.32676635067857873

classname: small-vehicle

ap: 0.09090909090909091

classname: large-vehicle

ap: 0.321147237203076

classname: ship

ap: 0.22886234000603503

classname: tennis-court

ap: 0.620826880836447

classname: basketball-court

ap: 0.3515151515151515

classname: storage-tank

ap: 0.5260798534560256

classname: soccer-ball-field

ap: 0.3982918546280185

classname: roundabout

ap: 0.25095307917888565

classname: harbor

ap: 0.36856141853682617

classname: swimming-pool

ap: 0.322982623748434

classname: helicopter

ap: 0.30171277997364954

map: 0.36964934714376324

classaps: [69.24784382 59.34263509 15.02267574 32.67663507 9.09090909 32.11472372
22.886234 62.08268808 35.15151515 52.60798535 39.82918546 25.09530792
36.85614185 32.29826237 30.171278]

6.3 Simulation of Atmospheric turbulence Blur

Fast and accurate simulation of imaging through atmospheric turbulence is essential for developing turbulence mitigation algorithms. Recognizing the limitations of which were present already, a new concept known as the phase-to-space (P2S) transform which can significantly speed up the simulation was studied and tried to implement to our validation dataset.

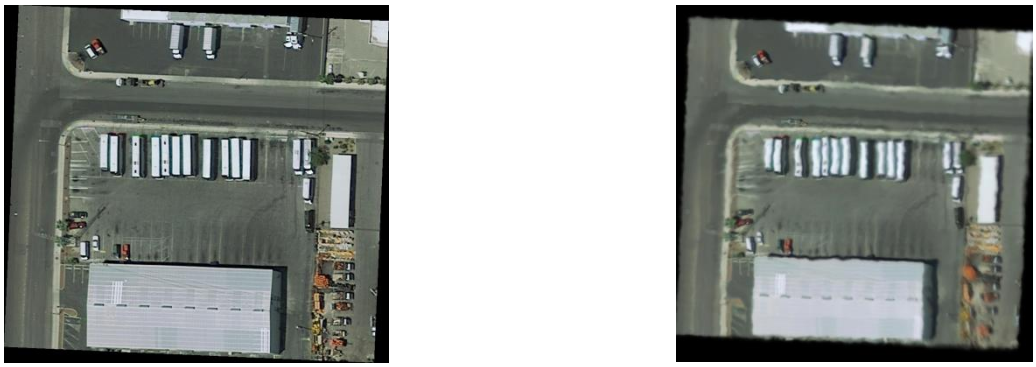
P2S is build upon three ideas:

(1) Reformulating the spatially varying convolution as a set of invariant convolutions with basis functions.

(2) Learning the basis function via the known turbulence statistics models,

(3) Implementing the P2S transform via a light-weight network that directly converts the phase representation to spatial representation.

This new simulator was observed to offer 300x -- 1000x speed up compared to the mainstream split-step simulators while preserving the essential turbulence statistics. Validation dataset was used to simulate atmospheric turbulence blur to the images of that dataset, and then tested with the model.



(a) (b)

Figure 4 : An illustration of the simulation of motion blur

(a) Normal image (b) Atmospheric turbulence blurred image

6.3.1 Limitations

While trying to implement this new method to our DOTA validation set, there were Some limitation noticed.

They were:

- The present simulation can be done only on square images.
- The present simulation can be done only on images with max size (1024,1024)

6.3.2 Implementation and results.

For the above mentioned reasons it was taken care to reduce the images to square images. Then the images which can be simulated according to the size restriction were simulated and then tested with the model. The results are attached below.

classname: plane

ap: 0.0

classname: baseball-diamond

ap: 0.004784688995215311

classname: bridge

ap: 0.0

classname: ground-track-field

ap: 0.27272727272727276

classname: small-vehicle

ap: 0.09090909090909091

classname: large-vehicle

ap: 0.011363636363636364

classname: ship

ap: 0.0

classname: tennis-court

ap: 0.13114754098360656

classname: basketball-court

ap: 0.36363636363636365

classname: storage-tank

ap: 0.0

classname: soccer-ball-field

ap: 0.0

classname: roundabout

ap: 0.0

classname: harbor

ap: 0.0

classname: swimming-pool

ap: 0.0

classname: helicopter

ap: 0.0

map: 0.058304572907679035

classaps: [0. 0.4784689 0. 27.27272727 9.09090909 1.13636364

0. 13.1147541 36.36363636 0. 0. 0.

0. 0. 0.]

6.4 Simulation of noise

The validation dataset was used to simulate noise and then test with the model .Random noise module from skimage was used for the process. The mode used in the random noise was Gaussian with variance 0.1.

(a) (b)

Figure 5: An illustration of the simulation of noisy

(a) Normal image (b) Noisy image

6.4.1 Results

classname: plane

ap: 0.7237441914887486

classname: baseball-diamond

ap: 0.25668449197860965

classname: bridge

ap: 0.09090909090909091

classname: ground-track-field

ap: 0.09090909090909091

classname: small-vehicle

ap: 0.5785400169229519

classname: large-vehicle

ap: 0.772844252161494

classname: ship

ap: 0.7953298630902484

classname: tennis-court

ap: 0.723062320741326

classname: basketball-court

ap: 0.23593073593073594

classname: storage-tank

ap: 0.610276794788549

classname: soccer-ball-field

ap: 0.16083916083916083

classname: roundabout

ap: 0.17676767676767677

classname: harbor

ap: 0.7620259731904611

classname: swimming-pool

ap: 0.45403279473643443

classname: helicopter

ap: 0.2505827505827506

map: 0.4454986136691552

classaps: [72.37441915 25.6684492 9.09090909 9.09090909 57.85400169 77.28442522
79.53298631 72.30623207 23.59307359 61.02767948 16.08391608 17.67676768
76.20259732 45.40327947 25.05827506]

6.5 Simulation of Gaussian Blur

Similar to the simulation of noise mentioned above for the simulation of Gaussian blurs a module from skimage was used. The validation dataset was used to simulate Gaussian blur and then tested with the model. Gaussian filters module from skimage was used for the process..

(a) (b)

Figure 6: An illustration of the simulation of Gaussian blur

(a) Normal image (b) Gaussian blurred image

The resultant map was observed to be 0.1 and as we can see from the images it was pretty evident that the model won't be able to perform that good. The variance was set to high value (0.1) to actually test the model under extreme conditions.

The model was also tested with low variance (0.01) for instance and it performed well with map 0.45

CHAPTER 7

Retraining the Model

7.1 Overview

To test how the model performs with more real life like data, we created more datasets from simulation and tested them. The results were bad and not up to the standards of the state of the art model. To see if the model performed well if it is trained with more real life like dataset, the fine tuning process was performed.

7.2 Motion Blur

For fine tuning the model to test for motion blur the training dataset (DOTA v1.,0) was downloaded and then the motion blur was applied to the images, after which the model was trained for 3 more epochs from the checkpoint file of epoch 12.

Now with the new checkpoint file after training the motion blur test dataset which was simulated earlier is tested again. The results are attached below.

classname: plane

ap: 0.897384194537383

classname: baseball-diamond

ap: 0.8251323618666322

classname: bridge

ap: 0.5775343271399943

classname: ground-track-field

ap: 0.6855725880682307

classname: small-vehicle

ap: 0.5952708938374129

classname: large-vehicle

ap: 0.7518409612469756

classname: ship

ap: 0.7672778665054064

classname: tennis-court

ap: 0.9074144931074888

classname: basketball-court

ap: 0.7310149379101146

classname: storage-tank

ap: 0.8084151495231706

classname: soccer-ball-field

ap: 0.717572047851575

classname: roundabout

ap: 0.6195425484160633

classname: harbor

ap: 0.7877335809810467

classname: swimming-pool

ap: 0.6709034171816368

classname: helicopter

ap: 0.6362059312878985

map: 0.7319210199640688

classaps: [89.73841945 82.51323619 57.75343271 68.55725881 59.52708938 75.18409612
 76.72778665 90.74144931 73.10149379 80.84151495 71.75720479 61.95425484
 78.7733581 67.09034172 63.62059313]

7.3 Atmospheric Turbulence Blur

For fine tuning the model to test for atmospheric turbulence blur the training dataset (DOTA v1.,0) was downloaded and then the process of simulating atmospheric turbulence to the images was done, after which the model was trained for 3 more epochs from the checkpoint file of epoch 12.

Now with the new checkpoint file after training the atmospheric turbulence blur test dataset which was simulated earlier is tested again. The results are attached below.

New map: 0.10015187000642735

7.4 Noisy images

For fine tuning the model to test Noisy images the training dataset (DOTA v1.,0) was downloaded and then the process of simulating random noise to the images was done, after which the model was trained for 3 more epochs from the checkpoint file of epoch 12.

Now with the new checkpoint file after training the noisy images test dataset which was simulated earlier is tested again. The results are attached below.

New map: 0.7964068823

7.4 Gaussian Blur

For fine tuning the model to test Gaussian blur images the training dataset (DOTA v1.,0) was downloaded and then the process of simulating Gaussian blur to the images was done, after which the model was trained for 2 more epochs from the checkpoint file of epoch 12.

Now with the new checkpoint file after training the gaussian blur test dataset which was simulated earlier is tested again. The results are attached below.

New map: 0.2223478912

‘

CHAPTER 8

8.1 CONCLUSION

The state of the art model faster_rcnn_RoITrans was implemented and tested with DOTA dataset. We can see the model performed well with the original DOTA dataset with high clarity images, but when tested with simulated data the model couldn't perform that well, actually performing pretty bad except some cases. The fine tuning process performed on the model with the simulated dataset proved the resultant model can perform better on those type of data.

8.2 Future possibilities

The future improvements or possibilities which can be explored with the model on the DOTA can be perform distillation and find out how much can we reduce the size of the network. One more thing which can be tried is to try changing the backbone of the network, trying self supervised training/learning.

REFERENCES

[1] Ding, Jian and Xue, Nan and Xia, Gui-Song and Bai, Xiang and Yang, Wen and Yang, Michael and Belongie, Serge and Luo, Jiebo and Datcu, Mihai and Pelillo, Marcello and Zhang, Liangpei - IEEE Transactions on Pattern Analysis and Machine Intelligence - Object Detection in Aerial Images: A Large-Scale Benchmark and Challenges - 2021

[2] Xia, Gui-Song and Bai, Xiang and Ding, Jian and Zhu, Zhen and Belongie, Serge and Luo, Jiebo and Datcu, Mihai and Pelillo, Marcello and Zhang, Liangpei - DOTA: A Large-Scale Dataset for Object Detection in Aerial Images - The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) - 2018

[3] Jian Ding, Nan Xue, Yang Long, Gui-Song Xia, Qikai Lu- Learning RoI Transformer for Detecting Oriented Objects in Aerial Images - The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)

[4] Chen, Kai and Wang, Jiaqi and Pang, Jiangmiao and Cao, Yuhang and Xiong, Yu and Li, Xiaoxiao and Sun, Shuyang and Feng, Wansen and Liu, Ziwei and Xu, Jiarui and others - MMDetection: Open mmlab detection toolbox and benchmark - 2019

[5] Zhiyuan Mao, Nicholas Chimitt and Stanley H. Chan, ‘‘Accelerating Atmospheric Turbulence Simulation via Learned Phase-to-Space Transform’’, accepted to IEEE International Conference on Computer Vision, 2021

Code and Supplement Data

[1] https://colab.research.google.com/drive/1cPHcL3dYgxCOcicmNJ1nei1HRICMKPrQ?usp=sharing

[2] https://colab.research.google.com/drive/1uPE1_BwavFKXT2MqYrzQK9psM-riU1rZ?usp=sharing

[3] https://colab.research.google.com/drive/1klBhYFt2tiX2AgI0DZ6uczDW5F7Xn8h3?usp=sharing