Universiteit Leiden

Master Computer Science

Enhancing Oriented Object Detection with Adaptive Tiling

Name:	Kyriakos Aristidou
Student ID:	3510123
Date:	Friday 25th October, 2024
Specialisation:	Data Science
1st supervisor:	Dr. Erwin Bakker
2nd supervisor:	Prof. dr. M.S. Lew

Master's Thesis in Computer Science

Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands

Acknowledgements

I would like to take this opportunity to thank all those who have contributed to my work as I complete this research journey. First, let me thank my academic supervisor, Dr. Erwin Bakker, for your invaluable guidance and feedback throughout this thesis. Your insights have therefore been enormously instrumental in shaping my understanding of this field. I would also like to thank Alessandro Scoppio from Mainblades, a private company which gave the opportunity to join their team for my internship and to complete it in order to enrich my thesis. Your mentorship connected theory with practice and thus gave more value to my research experience. And last but not least, to my family and friends, thanks for all of your constant support and believing in me. That was pretty motivational. To all who played a role on this journey, I thank you for your contributions. This thesis is a reflection of our combined effort.

Abstract

Object detection in high-resolution images is challenging when faced with small, multi-scale, and oriented objects while having to maintain computational efficiency. The traditional downscaling methods may result in a loss of image details and thus inaccuracies, especially for small objects. This thesis develops SegTiling, a Segmentation-based Adaptive Tiling approach for oriented object detection. The approach has been evaluated on the DOTA dataset and a proprietary dataset from Mainblades specialized in aircraft drone inspections. SegTiling consists of a preprocessing phase, followed by an adaptive tiling phase, and concludes with an object detection phase. Preprocessing sharpens object boundaries by segmentation techniques, while Connected Component Analysis (CCA) selects regions that guide the process of tile creation for optimal detection and reduced computational overhead. Extensive experiments are conducted that empirically compare SegTiling with the traditional and no-tiling methods, demonstrating significant performance gains in mean Average Precision (mAP) and computational efficiency, especially over the DOTA dataset. Moreover, larger sizes of images during inference show an improvement mAP on Mainblades dataset. On the other hand, the sensitivity of SegTiling to image quality in the context of multi-scale object detection underlines a number of limitations, which are mainly due to its dependence on the accuracy of segmentation. SegTiling, by nature, acts effectively depending on the quality of the segmentation as poor-quality images may lead to less accurate segmentation and hence reduced detection performance. Despite these limitations, this thesis shows the efficiency of SegTiling for improving object detection, especially for small and oriented objects in high-resolution images. The study also points out some possible future research directions, including but not limited to applying SegTiling to other datasets different from aerial and high-resolution images and embedding SegTiling into object detection architectures to make these latter architectures even more robust and adaptable. Overall, the results show that adaptive tiling using segmentation techniques greatly enhances the detection accuracy and computational efficiency for high-resolution image analysis.

Contents

List of Figures

List of Tables

1 Introduction

The highly dynamic research area of computer vision still faces challenges in various object detection tasks. This is especially true for tasks related to detecting small, multi-scale, and oriented objects. A main challenge for high-resolution images is the requirement to downscale the images to reduce the resolution size so computers can handle them. The obvious solution to that problem carries its own risk like losing context information about the objects depicted in the images, especially for small objects. Traditional methods, while effective to some extent, are inadequate when faced with the complexities inherent in high-resolution images.

Tiling techniques as presented in the paper [28] are employed to work with high-resolution images. Tiling is a way of partitioning an image into multiple smaller images(tiles) so that it becomes easier to work. Using tiling can accomplish less computational load. Processing benefits by simply tiling for any size of image. However, this technique can become challenging and can lead to a loss of image integrity if the tiles are too small or when there are too many tiles. The fixed-size tiles may not align optimally with the objects of interest, leading to fragmented and less accurate detection results.

In this thesis, adaptive tiling [24] is presented as a powerful technique for handling high-resolution imagery in the context of oriented object detection. Standard tiling, in general, is usually done with fixed-size tiles that are placed side by side to cover the area of interest without overlapping and without leaving any space between them. In many cases, fixed-size tiles [28] work more effectively compared to using images without tiles. Rather than relying on the constraints of traditional downscaling, adaptive tiling leverages the inherent diversity and complexity, splitting it into properly sized tiles. These tiles are processed separately, preserving the original image's integrity and quality while keeping the required image context for accurate detection. Once analyzed, these tiles are seamlessly reconstructed, resulting in a complete image with the object detection results.

This thesis presents a novel method with the goal of improving object detection models that go beyond similar tiling methods like SAHI [1] by integrating segmentation techniques into the adaptive tiling process. Unlike existing methods that rely on randomly generated tiles as used in "The Power of Tiling for Small Object Detection" [28] paper, SegTiling, which stands for Segmentation-Based Adaptive Tiling, leverages image-specific features obtained by using segmentation strategies to guide the creation of contextually meaningful tiles that preserve object boundaries and spatial coherence. This novel approach enhances the accuracy of object detection as demonstrated on datasets like DOTA and the industrial dataset provided by the company Mainblades which specializes in aircraft drone inspections. SegTiling improves the precision of object detection while keeping computational efficiency by reducing the number of generated tiles. The flexibility observed in practical applications, especially within industrial inspections, underlines its robustness and potential impact it will have on a wide range of object detection applications.

The rest of the paper is organized as follows: In Section 2 describes the problem statement, which comprises motivation and a detailed problem definition. Section 3 reviews relevant literature and puts the approach into a larger perspective with regard to object detection. In Section 4, the metrics are introduced that are used to evaluate the performance of SegTiling. In Section 5, the various baseline tiling approaches used to which SegTiling is compared as introduced. Section 6 presents the SegTiling method in detail, along with the benchmark and industrial data. Section 7 presents the experimental setup. Section 8 describes the experiments. Section 9 reports the results. Findings are discussed in Section 10 along with a discussion on its limitations and possible future scope. The contributions and the implications of the research performed in this thesis are summarised in Section 11.

2

Problem Statement

2.1 Motivation

Recent breakthroughs in computer vision and deep learning have really revamped object detection from autonomous driving to medical diagnostics applications [11] [7]. Despite all these advances, multi-scale object detection remains challenging for high-resolution images. Multi-scale detection has a great demand for managing huge variations of scales and designing anchor boxes that can catch objects of different sizes without adding computational complexity to the overall process. High-resolution images have very rich detailed information that is very helpful for accurate detection, while size puts great demand on computation. This most often requires downscaling the image to reduce computational demands which subsequently brings loss of information, especially in the details which are very valuable, leading to a decrease in accuracy of the detections. Various systems that demand high accuracy, like satellite imagery [2], medical imaging [16], and surveillance [3], as even minor inaccuracies may result in big consequences. Companies dealing in aircraft drone inspections, such as Mainblades, apply this multi-scale object detection technology to perform aircraft inspections using UAVs, commonly referred to as drones. This application uses high-resolution cameras mounted on drones to intricately inspect aircrafts for damages with refined computer vision techniques to make a both precise and reliable assessment. This motivates the search for innovative approaches that can handle high-resolution data effectively without compromising on detail and computational efficiency.

2.2 Problem Definition

The focus of this thesis is to solve the challenge of how maintaining the balance between high accuracy of detection for multi-scale and small objects and high-resolution images without sacrificing computational efficiency. The case study of detecting lightning strikes

on aircraft surfaces was very prominent with the introduction of drone imagery in high resolution by the Mainblades company. Detection definitely requires an accurate and detailed inspection since it normally involves a very small and inconspicuous strike on the aircraft. The general size of lightning strike marks is only a few millimeters, while high-resolution cameras used in drones capture images with very fine spatial resolutions, often in the range of micrometers per pixel. Conventional object detection models normally optimize the input size and require efficient methods to handle high-resolution images without excessive computational demands.

These few strike marks have to be found in the presence of issues such as resolution reduction, detail loss associated with high-resolution image downscaling, the need for robust multi-scale detection associated with the presence of features at different scales, requirements for sophisticated techniques associated with finding and locating oriented marks, and computation-efficient processing methods. In this framework, the current thesis proposes the SegTiling adaptive tiling approach. This work combines edge detection, color segmentation, and Connected Component Analysis (CCA) into a unified framework with advanced models such as the Vision Transformer and Spatial Transform Decoupling [34] by processing high-resolution images in computationally feasible tiles, each at its optimal resolution.

3

Related Work

3.1 Oriented Object Detection

Oriented object detection refers to all the methods used in detecting objects of various orientations in images. This is comparatively more complex than the classic object detection, assuming that the objects in an image are axis-aligned [19].
Traditional approaches towards object detection started with classification schemes where models are trained to classify images into categories. These were quickly adapted to frameworks of object detection like the R-CNN family [27] that joined region proposal methods with CNNs to carry out object detection in images. However, these models struggled to detect objects which are not aligned to the image-axes.
Oriented object detection specifically addresses this limitation by developing methods that can detect objects accurately irrespective of their orientation. For this, traditional methods are used to rely on handcrafted features [23] in handling objects with different orientations. While being innovative in this respect, these approaches again suffered from limitations relating to flexibility. Their ability to generalize and adapt to a wide variety of object orientations and shapes is restricted. As a result, they often struggled to maintain accuracy when confronted with diverse or unpredictable conditions in real-world scenarios.
Recent advances in oriented object detection have led to efficient and accurate frameworks being set up, targeted at dealing with objects of arbitrary orientations. Among these, there exists a framework based on Oriented R-CNN [30], which is two-stage in nature. This utilizes an oriented Region Proposal Network, RPN [27] and an oriented R-CNN head to achieve state-of-the-art detection accuracy on datasets such as DOTA [29] and HRSC2016 [8] at the time of their respective releases. It was followed by the Learning RoI Transformer [10], which addressed the problems in aerial object detection by applying spatial transformations on RoIs, leading to enhanced performance on datasets like DOTA

and HRSC2016 by a large margin. Besides, it has inspired revisiting the classification-based approach to arbitrary-oriented object detection [33], where some new techniques involved, such as Circular Smooth Labeling and Densely Coded Labels, can effectively reduce model parameters while solving boundary problems found in existing regression-based detectors. Very recently, there is increasing interest in using transformers for oriented object detection. Models like Oriented Object Detection with Transformer (O2DETR) [21] and Arbitrary-Oriented Object Detection Transformer (AO2-DETR) [9] extend the capability toward direct processing of oriented objects with competitive improvements over the traditional detectors while providing much-simplified pipelines without any hand-designed components. Recently, Vision Transformers have emerged as a powerful backbone for various computer vision tasks, oriented object detection included. Unlike the traditional CNNs, ViTs intrinsically capture global context better by treating images as sequences of patches. This attribute, thus, inherently makes them more suitable for arbitrary-oriented object detection. Combination with Spatial Transform Decoupling(STD) [34] further enhances its capability to a state-of-the-art on the benchmark datasets including DOTA-v1.0 [29] and and HRSC2016 [8], as of the latest available evaluations. STD is a technique to decouple the spatial transformation from feature extraction to let the model handle complex spatial variations. This increases the detection accuracy of oriented objects because it can now generalize and adapt more effectively with a diverse range of orientations and scales that high-resolution images may have. Their improvements present a potential capability for transformer-based models to excel in oriented object detection tasks, thus allowing for better accuracy and efficiency in detecting objects that come in different orientations within the images. This thesis focuses on building upon these advancements applying a preprocessing approach using segmentation techniques.

3.2 Multi-Scale Detection

Multi-scale detection becomes important to precisely locate objects of different sizes in an image, since objects may appear at different scales due to perspective, distance, or occlusion. A couple of attempts have been made to approach this challenge by using techniques such as multi-scale feature fusion and scale-adaptive feature extraction. One may refer to the following models utilizing multi-scale feature pyramids for effective object detection across resolutions and scales, for example, Feature Pyramid Networks (FPN) [17] and Single Shot MultiBox Detector (SSD) [20]. In [26], an improved version of one-stage object detectors is proposed based on the YOLOv5 method, which is called

Multi-scale Feature Cross-layer Fusion Network (M-FCFN). By fusing shallow and deep features in PANet structure using cross-layer feature fusion, outputs can be provided at different feature scales, improving the accuracy of detecting objects of diverse sizes. For object detection in unmanned aerial vehicles (UAVs) taken images, [38] proposes Self-Attention Guidance and Multiscale Feature Fusion (SGMFNet). This approach leverages global-local feature guidance and parallel sampling feature fusion to solve some challenges of UAV images effectively, including complex backgrounds and remarkable scale differences. Solving the problems caused by multi-scale objects and arbitrary orientations is a very crucial issue in remote sensing imagery. A unified framework that includes a feature-fusion architecture to improve the feature representation for objects of different sizes, is presented in the paper [14]. This framework also proposes a rotation-aware object detector with oriented boxes for accurate object localization in remote sensing images.

However, transformer-based models, such as DEtection TRansformer (DETR) [5], have demonstrated limitations related to efficiently processing multi-scale detection tasks. While they process the input in parallel, transformers are very likely to be insensitive to fine-grained details and spatial relationships at different scales. Recent variants of DETR and other transformer-based models have been proposed to overcome the challenges brought about by multi-scale detection in transformers such as DETR. The Iterative Multi-scale Feature Aggregation (IMFA), as proposed in [36], provides a paradigm for efficiently exploiting multiple-scale features in transformer-based detectors. The IMFA leverages sparse multi-scale features and rearranges the encoder-decoder pipeline, with very minimal additional computational cost, yielding considerable performance gains. Recent attempts at improving multiscale DETR detection include DETR++, which proposes a new architecture with a Bi-directional Feature Pyramid, known as BiFPN, for the effective integration of multi-scale features. DETR++ [35] also gives notable improvements in the accuracy of detections against previous baselines. Meanwhile, a lightweight object detection framework was also proposed for Lite DETR [15] to alleviate some of the computational inefficiencies in multi-scale feature fusion in DETR variants. In that work, an interleaved encoder block and key-aware deformable attention were leveraged; Lite DETR achieved much lower computational costs while retaining most of the detection performances. Vision Transformers (ViTs) [12] have also promised improvement in multi-scale detection capabilities, especially when working together with STD [34]. The main intuition behind the approach used in ViT lies in splitting the image into patches and providing the sequence of linear embeddings of these patches as

input to the model, allowing them to consider the whole input image with full contextual information, capturing effective details at several scales. In this case, the main reason of splitting the image is to fit to transformer architecture.

3.3 Tiling Strategies for Object Detection

It is common practice to use the tiling approach in order to alleviate computational and memory burdens while processing large-size images [28]. The tiling approach divide the input image into smaller tiles so the object detection models can process the images effectively without reaching hardware limitations. Traditional object detection models utilize a tiling approach [28] by divide the input image into fixed-size tiles and processing tiles independently. This setup provides a way to use the model for large images, but can have problematic settings leading to incomplete object detections or object fragmentation at tile boundaries, and also inefficiencies in capturing contextual information across tiles. Other works explore other tiling aspects for object detection, such as by Plastiras et al. (2020) [25], where the authors investigate several ways of developing efficient pipelines for resource-constrained devices in the areas of combined pre-processing mechanisms with quantization and further exploration of different tiling approaches. Another paper [6] proposes a CNN processor with hierarchical pipelining and multicore reconfigurable computing for the effective detection of objects on FPGAs with optimal use of computing units and on-chip memory utilization. A different method called SAHI [1] presents a critical improvement in the domain of small object detection, since one of the well-acknowledged problems is detecting far and low-resolution objects in surveillance imagery. It proposes slicing-aided inference to improve the detection capability of existing object detectors without any further fine-tuning. This method serves as the baseline for comparison in this thesis where the proposed SegTiling technique demonstrates superior mAP scores. Contrary to SAHI, SegTiling employs segmentation techniques to generate tiles in an optimized and meaningful manner that lets SegTiling perform object detection more accurately and handle multi-scale and oriented objects better.

Adaptive tiling is a refinement of the traditional tiling approach, whereby tile sizes and positions are dynamically adjusted according to the characteristics of the input image and the objects contained therein. This adaptive strategy allows the model to direct more resources to the regions of interest, therefore improving both the accuracy and efficiency in detection. Several different approaches have explored the performance and capability of various adaptive tiling techniques toward better object detection. A

Dynamic tiling approach is proposed by S.Nguyen et al. [24], for small object detection that uses non-overlapping tiles with dynamic overlapping rates to achieve improved efficiency and accuracy. A paper leveraging model uncertainty and tiling [22] also tackles domain adaptation for object detection. It seeks a good balance between adversarial feature alignment and class-level alignment to enhance adaptation performance.

In oriented object detection, standard tiling and adaptive tiling play an important role in handling objects of arbitrary orientations with complex spatial distributions. The adaptive tiling techniques that were incorporated into oriented object detection frameworks enhanced the robustness of the model toward correct object detection of multiple orientations and scales in large-scale images. Recent papers addressed challenges related to detecting arbitrarily oriented tiny objects. In fact, one paper presents a method for tackling the challenges of detecting arbitrarily oriented tiny objects [31]. Employing dynamic priors and a coarse-to-fine labels assigner, "Dynamic Coarse-to-Fine Learning" mitigated mismatch issues and provided balanced supervision. Experimental results have shown state-of-the-art performance on datasets such as DOTA. Adaptive tiling enables the model to dynamically adjust tile size and position with object attributes, hence focusing computational resources on regions of interest for improving detection.

Although the mentioned methodologies present very noticeable advances in adaptive tiling and object detection, SegTiling enjoys this singular advantage of bringing in segmentation techniques right within the tiling process. Unlike other methods focused on the placement of tiles or domain adaptation, SegTiling focuses on the correct identification of areas of interest through segmentation before the actual execution of the tiling phase. By embedding segmentation into the adaptive tiling framework, SegTiling not only increases detection accuracy but also generalizes effectively across different datasets like DOTA [29] and object detection scenarios.

4

Fundamentals

This chapter introduces the metrics that help in assessing the efficiency and accuracy of the proposed approach in this thesis. These metrics provide a quantitative measure of the accuracy, efficiency, and overall performance of the object detection models used. Key concepts related to evaluation, such as Precision, Recall, Intersection over Union (IoU), Average Precision (AP), and mean Average Precision (mAP), are also covered.

4.1 Evaluation Metrics

Evaluation metrics have prime importance in assessing the entire process and ensuring the correctness of the predictions made by the model against the ground truth annotation. The following steps and methodologies will obtain common metrics such as precision, recall, average precision, and mean average precision. In this thesis, besides the standard VOC AP method [13], the VOC 2007 metric [13] is also used as an extra evaluation measure. These metrics are used to compare SegTiling to existing methods.

4.1.1 Precision and Recall

First of all, precision (4.1) and recall (4.2) are the very core metrics here. Precision tells how well the model was able to predict by finding the ratio of true positive detections to the overall amount of positive detections by the model. Recall, on the other hand, tells how well a model detected all instances by finding the ratio of true positives to the total actual number of positives in the ground truth dataset.

$$\text{Precision} = \frac{TP}{TP + FP} \tag{4.1}$$

Where:

- TP is equal to the number of True Positives.
- FP is equal to the number of False Positives.

$$\text{Recall} = \frac{TP}{TP + FN} \tag{4.2}$$

Where:

- TP is equal to the number of True Positives.
- FN is equal to the number of False Negatives.

4.1.2 Intersection over Union

Therefore, one of the essential elements in assessing object detection models is whether a predicted bounding box correctly matches a ground truth box. This is typically done using the Intersection over Union (IoU) metric (4.3), which provides how much overlap is between the predicted and ground truth bounding boxes. A detection is considered a true positive if the IoU exceeds a predefined threshold which is set to 0.5.

$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}} \tag{4.3}$$

Where:

- Area of Overlap: The region where the predicted bounding box and the ground truth bounding box intersect.
- Are of Union: The total area covered by both the predicted bounding box and the ground truth bounding box.

4.1.3 Average Precision

Average Precision provides a single number that summarizes the precision-recall trade-off of a detection system and characterizes overall performance. A variety of methods exists for computing AP, and in this context, two approaches based on commonly used VOC (Visual Object Classes) challenges [13] are considered. The VOC 2007 [13] 11-point metric calculates AP by averaging precision values at 11 equally spaced recall levels. Specifically, precision at recall values from 0.0 to 1.0 in steps of 0.1 is calculated. In this regard, for every recall level, the precision is interpolated to be the maximum precision attained for that recall or higher. Then, the AP is obtained by taking the average of the respective interpolated precision values. It represents the precision in discreet form, hence simplifying the process of evaluation and comparison across models.

Integral Method:

The integral method given in (4.4) is the default metric in this thesis and is used in later VOC challenges so is selected as the most recent metric. This approach computes AP as the area under the PR curve. First, in this method, the precision values are interpolated to create a precision envelope where the precision does not decrease as recall increases. Precision is then integrated over the range of recall values to compute the AP, effectively giving the area under the PR curve. The continuous approach yields more nuance towards the model's performance.
The AP in this case is given by:

$$\text{AP} = \int_0^1 \text{Precision}(r)dr \tag{4.4}$$

Where:

- Precision(r) represents the precision value at a specific recall level r.
- r is the recall value, which ranges from 0 to 1.
- dr represents a small change in the recall value, used in the integral to compute the area under the Precision-Recall (PR) curve.

Each of these methods has its merits and finds its application depending on the exact needs of the evaluation protocol. The VOC 2007 11-point metric provides a simple fixed recall level approach, whereas the integral method allows for an overall evaluation considering the whole PR curve.

4.1.4 Mean Average Precision

Mean Average Precision (mAP) (4.5) is calculated as the average of the APs across all object classes. This single metric provides a compact representation of model performance across multiple classes. It simultaneously conveys overall accuracy and robustness.

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i \tag{4.5}$$

Where:

- N is the total number of object classes in the dataset.
- AP_i is the Average Precision for the i^{th} object class.

Summarizing, these metrics are intended to systematically validate the performance of the object detection model. Due to the use of standardized metrics and methodologies,

rigorous and reliable assessment of the model predictions against the ground truth data is guaranteed. All this may be considered very important in light of finding strengths and weaknesses to improve further, but also to make sure the model has the standards for desired performance. Further on, it will provide an opportunity to evaluate the SegTiling approach.

5

Baselines Tiling Methods

In this section, the various baseline tiling approaches for object detection in high-resolution images are presented. Each method offers a well-known strategy for splitting the images, with its own strengths and limitations. These approaches serve as reference points for evaluating the performance of the proposed SegTiling method in experiments in this thesis.

5.1 Standard Tiling

Standard tiling is among the simplest tiling methods for object detection. In this process, an image is divided into small tiles of equal size, overlapping tiles, without any segmentation or preprocessing procedures. The main idea behind standard tiling is to divide a big image into more manageable segments that might be further processed through an object detection model.

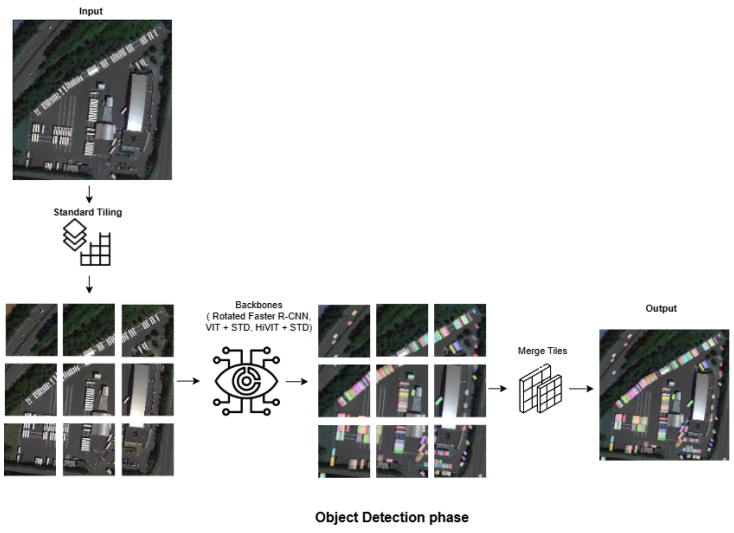

Figure 5.1: *Illustration of the Standard Tiling process.*

Figure 5.1 demonstrates the standard tiling process, where the input image is divided directly into fixed-size tiles without considering the specific features or objects present

within the image subsequently the tiles go through the object detection model for detection.

The naming convention applied in the standard tiling approach for this thesis encodes the position of each tile within the initial image, thereby enabling the straightforward reconstruction of the complete image in the pre-processing phase. Although this technique is uncomplicated and easy to implement, it frequently encounters difficulties in identifying small objects situated along the edges of tiles, which leads to fragmented detections and generates more tiles than necessary due to its simplified approach. For evaluation purposes these models are used: Rotated Faster R-CNN [32], Vision Transformer (ViT) with Spatial Transform Decoupling (STD) [34], Hierarchical Vision Transformer (HiViT) with STD [37].

5.2 Slicing Aided Hyper Inference (SAHI)

The Slicing Aided Hyper Inference (SAHI) [1] towards tiling is much flexible when compared to other traditional tiling methods. It is designed with the goal of improving the detection of small objects by refining the process of generating tiles with respect to dimensional size and positional information of objects that may exist within an image.

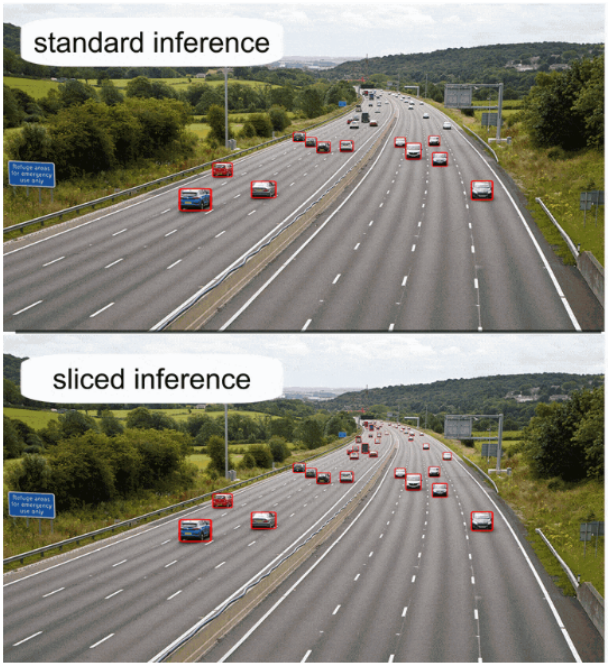

Figure 5.2: *This figure shows the performance of SAHI when sliced inference or standard inference is applied and how the small object detection is improved. Adapted from article "SAHI: A vision library for large-scale object detection & instance segmentation" [1]*

Instead of using regular tiles, SAHI slices the image into pieces that are intentionally centered around the locations where small objects are likely to appear. This process

makes sure that small objects fall in the center of a slice, hence increasing the chances that the object detection model will identify them correctly. The sliced inference approach used by SAHI as shown in Figure 5.2 modifies the dimensions and placement of the individual tiles according to the distribution of objects within the image. Such adaptive tiling methods are supposed to balance the scale between the preservations of the spatial context of a scene and the emphasis on fine details to improve detection precision on smaller objects. This involves the continuous division and subdivision of the image, where required, to enable the detection model to focus on the more limited areas of interest while retaining the overall context of the image. For evaluation of this method the model Rotated Faster R-CNN [32] is used.

While SAHI relies on the splitting of the image into overlapping patches both during fine-tuning and during the inference to increase the relative size of the small objects within the tiles, SegTiling follows a more concrete approach. It uses segmentation techniques to identify the regions of interest from the image before tiling, ensuring that only regions likely to have objects are tiled. This reduces unnecessary computations on empty regions and reduces object fragmentation.

5.3 Multi-scale tiling

This approach is used by the state-of-the-art model on the DOTA dataset [34] to obtain the final results. In this approach, images are divided into multiple overlapping tiles of multiple sizes, allowing the model to process multiple scales. Each image is cropped into patches of different sizes with varying overlaps of 500 pixels in the multi-scale setting. For the multi-scale approach, images are split into tiles by factors of 0.5×, 1.0×, and 1.5× of the image's original size. This multi-scale resizing ensures that objects of various sizes are captured effectively within the tiles. Finally, each tile fits to the model for object detection resizing to 1024x1024 fixed size.

5.4 Without Tiling

In the no tiling approach, the object detection model is presented with the entire image in one inference, which means that the entire image is fed into the object detection model at once, without breaking it into smaller sections. This method maintains the overall context of the image, enabling the model to assess the complete scene while generating predictions. However, a prominent limitation is its failure to detect small

objects in high-resolution images. The model's focus on the larger context suggests that fine-grained details may be sacrificed, and the related computational costs are significantly higher compared to the other tiled approaches. For the evaluation of this method the three object detection models are used same with the standard tiling method, Rotated Faster R-CNN [32], Vision Transformer (ViT) with Spatial Transform Decoupling (STD) [34], Hierarchical Vision Transformer (HiViT) with STD [37].

6

SegTiling

This chapter describes the proposed method, SegTiling which has the potential of enhancing oriented object detection. It consists of a pipeline involving a series of preprocessing, tiling, and object detection stages as depicted in Figure 6.1. SegTiling uses different image processing techniques for determining candidate regions of objects, adaptive tiling strategies, and potentially enhancing detection accuracy. This methodology mitigates some of the problems with the detection of objects with irregular orientations and sizes in large-scale images by incorporating segmentation techniques before the tiling phase. The following sections will provide a detailed explanation of each phase of the SegTiling framework.

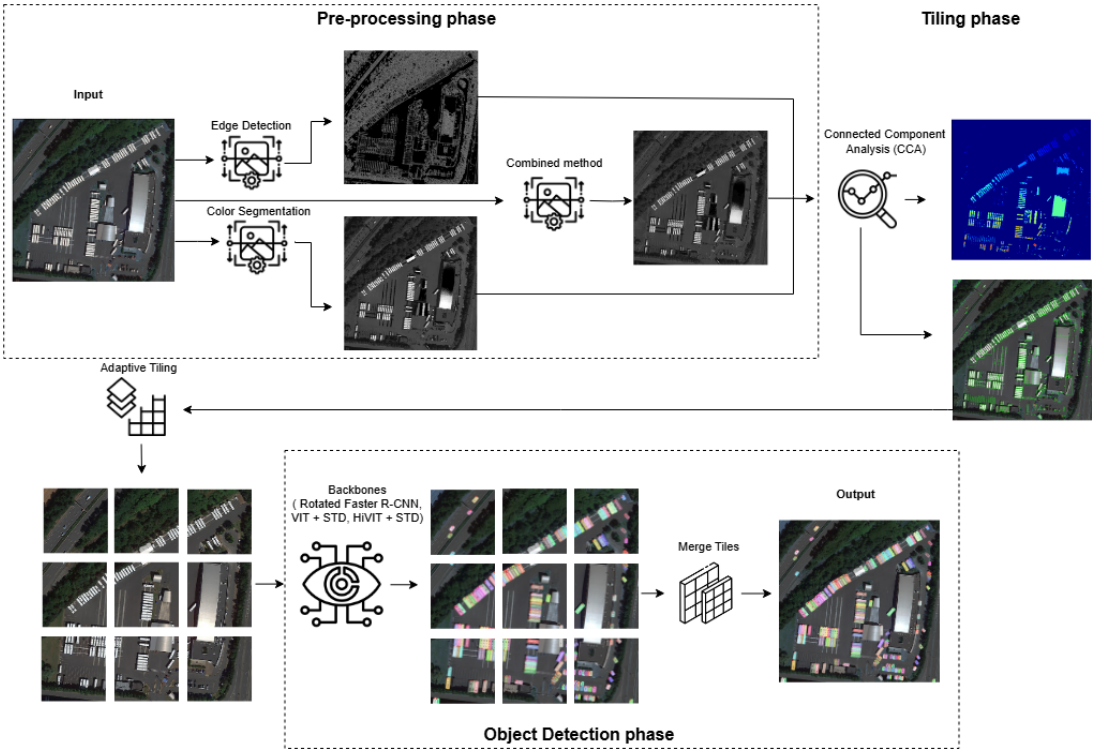

Figure 6.1: *The framework of SegTiling approach*

6.1 Pre-processing phase

Pre-processing is one of the crucial steps for preparing images for efficient object detection. It seeks to enhance areas in an image where objects might most likely appear without necessarily identifying them. An example of an input image is shown in Figure 6.2.

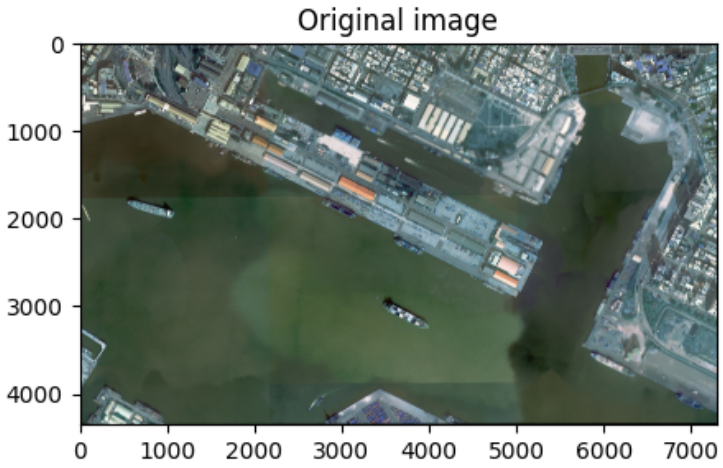

Figure 6.2: *Example of an image from DOTA dataset*

In the pre-processing phase a stepwise process is followed to develop the input image features so that the tiling phase is able to identify the area of objects more easily. In this respect, the pre-prosessing methods used here are edge detection, color segmentation, and a method combining both edge detection and color segmentation.

6.1.1 Edge Detection

Edge detection is an image processing technique used to determine boundaries in images. It focuses on finding edges, crucial changes in intensity that normally relate to object borders.

To apply this technique, the Canny edge detection algorithm [4] through OpenCV library is used since it recognizes a substantial amount of edges. The image is taken through several stages by the Canny algorithm. First, it smoothes out the image through a Gaussian filter to reduce the noise. Then it calculates the gradients of intensity of the image to identify places with big differences. Non-maximum suppression is applied, thinning the edges down to retain only the most important edges.

Finally, the edges are classified as strong and weak using a double threshold mechanism. Strong edges are those that have gradient values above a set high threshold, while weak edges have gradient values between the high and low threshold values. The edge tracking by hysteresis then ensures that weak edges are retained only if they are adjacent

to strong edges, eliminating any responses from isolated noise.
The Canny algorithm returns an edge map, indicating the most important structures inside the image, as illustrated in Figure 6.3. This edge map is used in further steps, guiding the adaptive tiling process to focus on regions with structural variabilities and features ensuring that areas with potential objects are accurately highlighted.

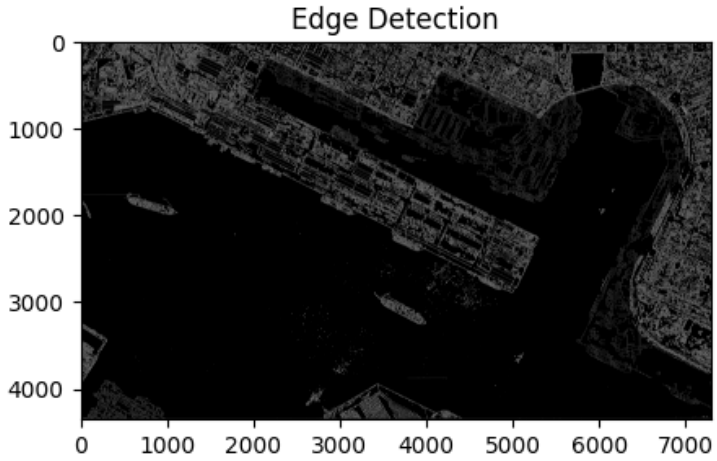

Figure 6.3: *Segmented image after edge detection is applied.*

6.1.2 Color Segmentation

Color segmentation called the separation of an image based on color regions that help in improving the contrast between objects and backgrounds. The technique is very effective in cases where the objects of interest have distinctive colors.

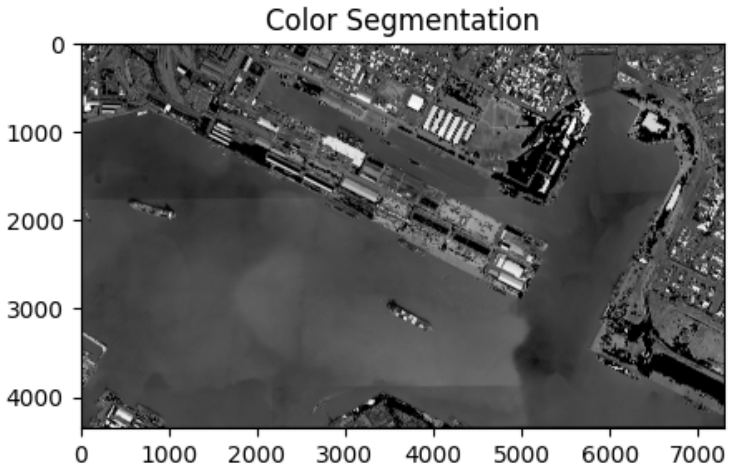

Figure 6.4: *Segmented image after color segmentation is applied.*

With the help of OpenCV library [4] it starts with converting the image from RGB color space to HSV(Hue, Saturation, Value) color space, in which color information is separated from intensity information. Then, a sequence of masks is applied in order to segment the specified colors of the image, targeting different hues of blue, green,

red, gray, white, and yellow. These colors are demarcated by their ranges within the HSV: blue is identified by hue values in the range of 100 to 140, green by values falling between 35 and 75, and red for a range close to 0-10 due to the cyclical nature of the HSV spectrum. Gray is defined by very low saturation and medium brightness so that it depends more on intensity rather than hue, whereas white has been identified by low saturation along with high levels of brightness. Yellow is outlined by applying hue values between 20 and 30.

These masks are then combined to form a composite that highlights areas with the corresponding color information, as shown in Figure6.4. A grayscale form of this image then serves to enhance the display of structural information and object boundaries within the scene. This segmented image is important for the tiling process, ensuring that regions with important color features are preserved and helped to highlight the objects for the next phase.

6.1.3 Combined Method

This combined approach integrates the edge detection and color segmentation methods to provide a complete view of the image. It collects the most probable areas with objects in it by using structural information defined by edge detection and color segmentation.

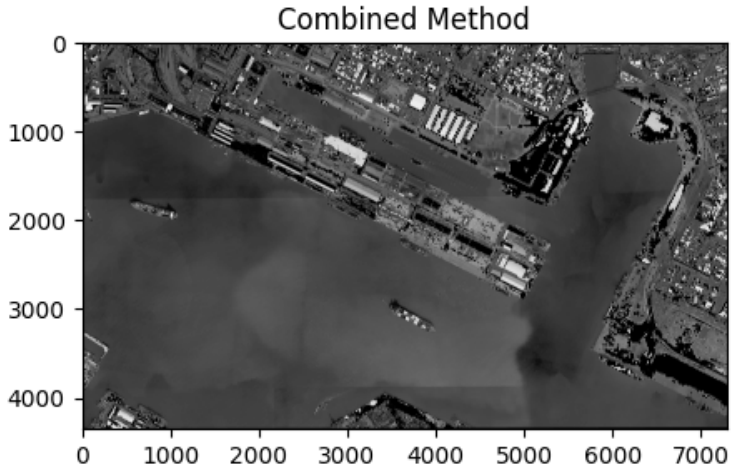

Figure 6.5: *Segmented image after a combination of edge detection and color segmentation is applied.*

It first converts the segmented image to grayscale. Both the grayscale segmented image and the edge-detected image are used to compute their histograms, with 256 bins a parameter that represents the intensity levels ranging from 0 to 255, to obtain their intensity distributions. For these histograms, the entropy is calculated, which measures the level of uncertainty or randomness in the image data. A higher entropy value indicates a more complex image with greater detail and variation in pixel intensities,

while a lower entropy suggests a simpler image with more uniform areas. Based on the value of entropy, weights are assigned for both images. The images are then blended using a weighted sum, where the color-segmented image and the edge-detected image contribute proportionally to their respective weights, resulting in a composite image that highlights regions with both significant structural details and distinct color variations. This combined image with both edge and color information offers a detailed and accurate representation of the areas in the image containing potential objects. From this Figure 6.5 we can see that color segmentation is more prominent than the edge detection information which means for this particular image subsequently used was weighted higher than the color segmentation information. This comprehensive representation is for guiding the adaptive tiling process to generate tiles around the most information regions.

6.2 Tiling phase

The tiling phase involves dividing a pre-processed image into smaller, manageable tiles for the object detection models. In this phase, Connected Component Analysis (CCA) will identify the area occupied by the objects. Based on this, an adaptive tiling approach will be applied to create efficient tiles. One of the main challenges in this phase is the possible fragmentation of the objects between the boundaries of the tiles. The adaptive tiling approach plays a crucial role in facing this challenge.

6.2.1 Connected Component Analysis

Connected Component Analysis (CCA) is the method used for finding the regions in a binary image that are connected by intensity. The CCA function is performed using OpenCV library [4]. This allows individual objects or areas of interest to be segregated from an image. First, thresholding is used to convert the grayscale image into a binary image and assign to each pixel either the maximum or minimum value of intensity based on a threshold of 100. It enhances the image further, providing sufficient contrast between the regions occupied by objects and the background. Moreover, morphological operations are used, specifically dilation and erosion, to refine the boundaries of the detected objects. Dilation extends the borders of objects by adding pixels to their edges, which helps in filling small gaps and connecting nearby regions. In this implementation, dilation is carried out with two iterations in order to ensure the capturing of even minor connections between components. It is followed by erosion that removes the pixels from the edges of objects in order to remove noise and small artifacts that might have merged

during dilation. Erosion here used four sets of iterations to reach a balance that maintains the basic structures of the objects involved while reducing noise simultaneously. The values chosen for the mentioned parameters of CCA were found by trial and error keeping track of the count number of detected bounding boxes, which a higher number, indicating more accurately identified objects within the image.

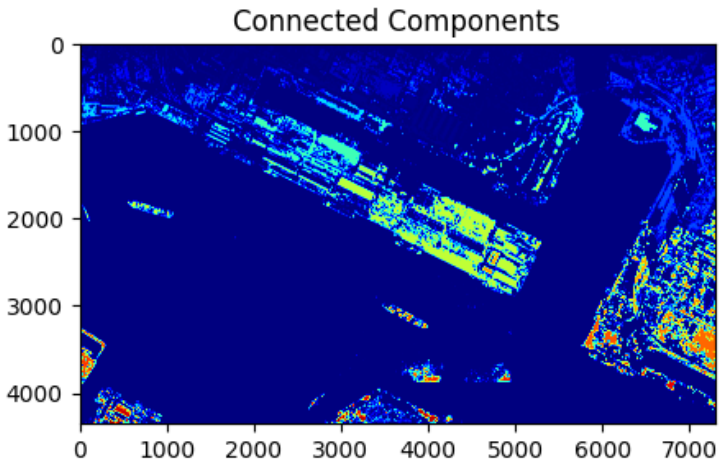

Figure 6.6: *Image visualization after CCA is applied using the segmented image by the combined method.*

Labeling each connected region in this way gives information about the number of objects, bounding box coordinates and areas. The information is used by the adaptive tiling method. The tiling can be adapted to where potential objects are located in the image. The output of CCA visually for Figure 6.6 is given here, using "jet" colormapping to highlight where regions differ. In this color scheme, regions with higher intensity values are represented by warmer colors (such as red and yellow), while lower intensity areas appear cooler (blue and green). The "jet" colormap helps to clearly distinguish between different connected regions in the image, making it easier to interpret the distribution and extent of detected objects.

6.2.2 Adaptive tiling approach

The Adaptive Tiling Approach represents a method aimed at generating tiles, adapted to the location and characteristics of the object-regions detected by the Connected Component Analysis. The strategy tries to optimize the regions covering the objects. This approach offers potential enhancement in the accuracy and efficiency of detection and may reduce object fragmentation.

Firstly, the adaptive tiling process creates a grid overlaying the image so that each cell in the grid corresponds to a potential tile. The number of rows and columns in this grid is determined by the dimensions of the image as well as the size of each tile and the overlap between them. The overlap parameter controls how much adjacent tiles overlap

with each other, which ensures that object boundaries that might lie on the edge of one tile are captured fully in the neighboring tiles. For each detected object, the grid is updated by marking the cells which have to be tiled. In this way, it can be made certain that all regions containing objects are included in the tiling process. As shown in Figure 6.7, the marked cells represent the generated tiles for this image. All grid cells containing at least one marked object will be generated as tiles. Cells without any marked objects will be excluded, this is how this approach optimizes the number of generated tiles.

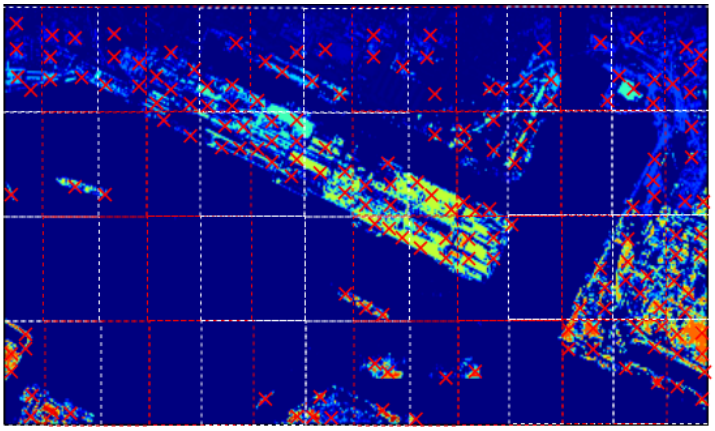

Figure 6.7: *Image visualization to show how the tiling approach works in this thesis.*

Once the grid is filled with marked cells, tiling has to be done by iteration through the grid. One tile is created for each marked cell by adjusting the coordinates and dimensions of its start so that the tile size is as specified, but completely inside the image. This aim to ensure that objects in an image, even at the edges, fall completely within at least one tile. The number of generated tiles depends on both the dimensions of the input image and also the overlap parameter. This implies it is generated dynamically during the process instead of being predefined. Specifically, the overlap and tile size are used to calculate the grid's density for determining the overall number of tiles generated in order to provide appropriate coverage of object regions.
In this setup, several parameters are crucial for achieving the best results in terms of detection performance and computational efficiency. These parameters include:

- **Non-Maximum Suppression (NMS):** NMS is a technique adopted to minimize unnecessary bounding boxes resulting from CCA by eliminating those with high inter-overlaps. Further refinement with an overlap threshold parameter ensures that only the boxes of relevance remain. It helps in reducing false positives and generally enhances the precision for object detection.
- **Min/Max area ratios:** Min/Max area ratios define what detected regions should be considered depending on their size with respect to the whole image. Ratios

are especially important for such cases that allow avoiding missing small salient objects while rejecting large irrelevant areas. An idea using these ratios allows a model to be more selective and focus on such areas which have high probability of containing an object of interest. These parameters work along with NMS.

- **Overlap size:** The overlap size between tiles is tuned as a trade-off, to find a good balance between complete coverage of the image and unnecessary computational processing. If the overlap of the tile's edges is more, then very few objects around the edges will be missed, but this is at the cost of extra computations. The appropriate trade-off is determined through experimental evaluations using mAP score, which assesses the model's performance. By analyzing how changes in overlap size affect the mAP, the system can be optimized for a detection rate that minimizes redundancy while ensuring effective coverage of all object regions.
- **Tile size:** The tile size is directly proportional to how much of the image is processed in a single run. Small-sized tiles are great for closely monitoring small objects. However, these inflate the number of tiles and raise the computational cost. In contrast, larger tiles decrease the number of tiles but may fail in the detection of small objects. The system explores different tile sizes through a process of trial and error, assessing performance via the mAP score for a better trade-off between accurate detection and efficient computation.

In a nutshell, the adaptive tiling phase finds out the regions of interest through Connected Component Analysis and dynamically adapting the tiling grid to completely cover the located object regions. This approach aims to improve both efficiency and accuracy in the object detection process and is particularly suitable for large-scale image processing tasks with varied distributions of objects.

6.3 Object detection phase

During the object detection phase, the goal is to detect objects within tiles generated during the tiling phase, and then those results for each tile are combined in order to produce the final result for the complete original image. This phase leverages the advantages of smaller sections of an image being processed and ensure that object detection is both precise and complete, particularly for small objects, across the entire image.

The adaptive tiling strategy divides the image into tiles which are independently processed with an object detection model. After each tile is passed through the detection

model, a set of predictions would be obtained consisting of bounding boxes, class labels, and confidence scores for each of the detected objects.
The tiles are saved with specific names that encode their location within the original image. This information is very important, as this is used by the merge functionality to get the correct positions of the objects and reassemble the detected objects in the context of the full image. After all the tiles have been processed by the detection model, the function merges all the results and builds the final results for the complete original image. This process is described in detail in Section 6.3.2

6.3.1 Object detection models

Several object detection models are used for the performance evaluation of SegTiling, and in order to understand the impact of the tiling strategy on the object detection models when detecting small objects and challenging objects such as in high-resolution images. The selected object detection models are Rotated Faster R-CNN, ViT with STD, and HiViT with STD. ViTs are selected for this thesis as the state-of-the-art models on DOTA for oriented object detection and Rotated Faster R-CNN as a light and computationally efficient model which is ideal for the experiments in this thesis.

Rotated Faster R-CNN

Rotated Faster R-CNN [32] expands the traditional Faster R-CNN model [27], targeted at oriented object detection. This model is very useful when dealing with oriented objects that are not aligned with the horizontal or vertical axes of the input image, especially in tasks where the orientations of objects may differ a lot, e.g. aerial imagery or scenes that have oblique perspectives as found in the DOTA dataset. While in general are less accurate, compared to transformer models such as ViT and HiViT, the rotated Faster R-CNN is relatively lightweight and computationally efficient, hence, it is faster to train and more convenient for experiments and applications where the computational resources are limited. In this thesis, the adaptive tiling approach is evaluated using rotated Faster R-CNN and is provided as the baseline. This evaluation helps to develop an understanding of how tiling methods are affecting the performance in oriented object detection.

ViT with STD

The Vision Transformer (ViT) with Spatial Transform Decoupling (STD) [34] is a transformer-based model that excels in capturing fine details and long-range depen-

dencies in images—an advantage when detecting small objects within high-resolution images. The STD component enhances the model's ability to handle variations in object orientation and positioning, making it particularly effective in scenarios requiring precise localization. In this thesis, ViT with STD is used to process the tiles generated by the adaptive tiling approach, offering insights into how transformer architectures can leverage the benefits of tiling for improved object detection.

HiViT with STD

HiViT stands for Hierarchical Vision Transformer [37], an advanced approach primarily for object detection, particularly oriented objects and their multi-scale variations. It adopts a spatial transformer decoupling (STD) method [34] which make it a state-of-the-art model. The concept of HiViT is to handle images hierarchically in order to extract features from an image at many different levels of granularity. It therefore works very efficiently when it comes to the complexity of large-scale high-resolution images. Despite its higher computational demands, HiViT is expected to outperform CNN-based models in accuracy, particularly in detecting challenging objects. HiViT is employed along with STD in this study to unlock the full potential of an adaptive tiling approach when highly accurate detection is critical.

Integration into the object detection phase

These models are applied to the tiles generated by the SegTiling approach and the other baseline tiling methods. By comparing the performance of each, the present work evaluates how each model interacts with the tiling strategy, especially in relation to the detection of small and complex objects. This is vital in realizing the strengths and limitations of adaptive tiling across different detection architectures and gives insights into possible further improvements using transformer-based models compared to traditional CNNs.

6.3.2 Merging tiles

Merging the object detection results of all the tiles is done by the following main steps:

1. **Extracting Coordinates:**This step involves extracting coordinates for each tile from its corresponding filename. This extraction has to be done so that for all the detected objects the coordinates in the original image can be calculated.

2. **Adjusting Bounding Boxes:** The predicted bounding boxes of the objects are projected to the original image so that the correct position of the objects with respect to the entire image is obtained
3. **Merging Detections:** The modified bounding boxes together with their class labels and confidence scores will be combined to make the final set of detections for the entire image. This will be achieved by aggregating all tile results, which may include overlapping parts where certain objects could have been detected in more than one tile. The combined outputs are then subject to duplicate removal and refinement of the final detections to have each object detected only once with its highest confidence score. Techniques such as non-maximum suppression are therefore involved in refining these results by removing redundant detections.

7

Experimental Setup

This section presents the experimental setup in terms of hardware and software configuration. The chapter begins by presenting the used datasets. Next is an elaboration on the hardware and software tools used in the experiments, and custom pre-processing scripts. Subsequently, the experimental setup ranged from data preparation and model training to performance evaluation. The goal of the experimental setup is to obtain an effective evaluation of the proposed method and its performance in real-world applications.

7.1 Dataset

For the performance evaluation of SegTiling the public benchmark dataset DOTA is used, which has been widely used in related research papers, and the real world dataset of Mainblades. The two dataset are further described in the next subsections.

7.1.1 DOTA Dataset

DOTA is a benchmark designed for evaluating object detection tasks in aerial imagery. The dataset of DOTA-v1.0 [29] has been chosen for this thesis because of its distinctive features that make it suitable for evaluating small objects in high-resolution images and it has been widely used in related research papers.
Key Characteristics of the DOTA Dataset:

1. **High-Resolution Images:** DOTA-v1.0 comprises large-scale, high-resolution images. This could be highly resourceful for the detection and analysis of small objects. High resolution will, therefore, be helpful to observe features that might be of utmost importance to be detected and classified accurately.
2. **Diverse Object Categories:** The ground truth involves object categories from the more general ones, such as cars and planes, to those that could be considered

specialized, like ships and storage tanks. The effect of this kind of diversity is to make sure that the model sees enough variations of real-world scenarios that would contribute to improving the generalizability of the model.

3. **Complex Backgrounds:** Given the fact that these images have an aerial perspective, objects generally show up against complex and cluttered backgrounds. This acts as a challenge to object detection algorithms to effectively distinguish the objects from their surroundings by placing a rigorous test of the model's capabilities.
4. **Varied Object Sizes and Orientations:** The objects in DOTA-v1.0 vary by a large margin in size and orientation, so it is very useful to test the robustness of the detection methods, especially for small objects in different orientations.
5. **Comprehensive Annotations:** It also contains more detailed oriented bounding box annotations for each object. That kind of annotation is quite ideal for understanding object positioning and orientation in precise object detection tasks which this study requires.

Dataset	Number of Classes	Train-set	Validation-set	Test-set
DOTA-v1.0	15	1411 images	458 images	937 images

Table 7.1: *Summary of DOTA dataset*

The DOTA-v1.0 dataset is concisely summarized in Table 7.1 and consists of 15 classes. The total data has 1,411 images in the train-set, 458 images in the validation-set and 937 images in test-set. There is a balance between the train-set images and val-set images that ensures the model evaluation will be fair. The experiments using DOTA dataset in this thesis were conducted on DOTA v1.0, where the model is trained on the train-set and tested on the validation-set because the test-set is without annotations provided by DOTA.

7.1.2 Mainblades Dataset

In addition to the DOTA dataset, this thesis utilizes a proprietary dataset provided by a private company Mainblades specializing in aircraft inspection using drones. This dataset shares common characteristics with DOTA in terms of image size and object diversity sizes. This dataset is selected to evaluate SegTiling to real-world images. The quality of images and the annotations are based on real-world scenarios and this makes it more challenging compared to more research-oriented public benchmarks like DOTA. Key Characteristics of the Mainblades Dataset:

1. **High-Resolution Images:** The dataset includes a wide variety of high-resolution

images encompassing a wide range of scales and resolutions. This kind of diversity enables evaluations of object detection algorithms under many conditions, from close-up detail to broad, wide-angle views.

2. **Diverse Sources:** The images in the dataset were taken from a variety of sources, such as drones, cameras, and smartphones. This variation in source can introduce variation both in image quality and angle, which more accurately reflects how data is actually captured in the real world under the parameters of differing settings and situations.
3. **Variability in Image Quality:** Some of the images have common problems of being blurry or of poor lighting, which impairs clarity. The variability tests the robustness of detection models and poor-quality imaging.
4. **Wide Range of Classes and Object Sizes:** The dataset involves many classes, all related to aircraft inspection, including the different types of damage, text, and marks from lightning strikes. Objects vary significantly in size, from very small and hard-to-detect features to larger, more prominent objects.
5. **Complex Annotations and Ground Truth:** The annotations provided are detailed and cover multiple objects within the same image, often overlapping or in close proximity. This complexity simulates the intricate and densely packed scenarios that are typical in aircraft inspection imagery and generally in real-world datasets.

Dataset	Number of Classes	Train-set	Validation-set
Mainblades	23	1199 images	514 images

Table 7.2: *Summary of Mainblades dataset*

Mainblades dataset as described in Table 7.2 It is divided into 23 different classes. The train-set consists of 1,199 images, while the validation-set contains 514 images. The structural setup of this dataset offers a large number of labeled examples meant for training and testing object detection models.

7.2 Implementation and Training settings

7.2.1 Hardware Setup

All the experiments in this thesis were performed on a system running Ubuntu 22.04.2 LTS, where two NVIDIA RTX A6000 GPUs with 50GB memory were available. This setup was employed during training, while for the inference phase, one of these GPUs was involved, providing enough computational power to handle the high demands of training and evaluating such a complex model as ViT with STD and HiViT with STD.

7.2.2 Tools and Frameworks

In this thesis, the training and evaluation were done using the MMRotate framework [39]. MMRotate is a part of the OpenMMLab projects focused on oriented object detection. The code provides full functionality for dealing with models like Rotated Faster R-CNN, Vision Transformers, and HiViT detectors with STD.
For working with the DOTA dataset, the DOTA Devkit was used. This toolkit already has all the utilities necessary to handle the DOTA data, among which are utilities that split images and process annotations. Based on DOTA Devkit for the purpose of this research, some custom script modifications were implemented, allowing it to execute the different image tiling and preprocessing described in Sections 6.1 and 6.2.

7.2.3 Experimental Procedure and Workflow

The experimental workflow is designed to systematically investigate the proposed adaptive tiling approach and its effect on object detection performance. More specifically, the process unfolds as follows:

- **Data Preparation and Preprocessing:** DOTA-v1.0 and Mainblades datasets are used for the training and evaluation processes. It includes parsing of annotations of both the datasets and preprocessing according to the input requirements of object detection models. Conve This step ensures that the data is correctly formatted for further processing in the subsequent workflow stages.
- **Model Training:** For Mainblades the two models, Rotated Faster R-CNN and HiViT with STD, are trained using pre-trained weights in both tiled and non-tiled versions. Regarding the DOTA dataset, only the Rotated Faster R-CNN was trained from scratch using the train-set for training and the validation-set for testing in all the experiments. HiViT with STD and ViT with STD are utilized using the pre-trained checkpoints from the released paper [34] which are pre-trained on trainvalidation-set and tested on validation-set in the experiments. This happens because the test-set on DOTA dataset is provided without annotations so it is not feasible to evaluate them.
- **Modification and Integration:** The DOTA Devkit script is extended to integrate the DOTA dataset and Mainblades dataset into the MMRotate framework. The ImgSplit tool available in the DOTA Devkit has been used to efficiently do image tiling. A custom script was also used to convert Mainblades data into the DOTA format so that the MMRotate framework could be applied. Models are compared

based on the DOTA Task1 script which relies on the mean Average Precision as its main metric as described in Section 4.1.

- **Inference and Performance Evaluation:** Inference is conducted on DOTA-v1.0 and Mainblades validation-sets using the different object detection models explained in Section 6.3. Tiling methods as explained in Section 5 are applied to the images, and performance evaluation is done through the DOTA Task1 evaluation framework, where the mean Average Precision (mAP) score is used. This metric is important because it is used in most of the related papers.

8

Experiments

The experiments are divided as follows:

- The first experiment starts with a comparison of the performance of existing baseline tiling approaches, no-tiling, standard tiling, and SAHI Tiling on the DOTA dataset.
- SegTiling is further analyzed with respect to its parameter optimization, using object detection models.
- Then the performance of the proposed method SegTiling is evaluated.
- Evaluation of the object detection models used for this thesis is presented using mAP scores.
- An experiment to evaluate SegTiling on real-world Mainblades dataset.
- Experimentation on training data and how tiling affects the mAP.
- The impact of modification of inference image size on mAP is analyzed and presented.
- The Final experiment shows the impact of SegTiling on different sizes of objects on the DOTA dataset.

These experiments are described in their respective subsections.

8.1 Comparison of baseline tiling approaches

This experiment is intended to evaluate the relative effectiveness of different tiling approaches in the context of the thesis. In this work, the impact of different tiling strategies on the detection performance of object detection models for small objects is investigated in the high-resolution aerial images of the DOTA dataset. This experiment will discuss the three approaches: Standard Tiling, SAHI Tiling, and a method without tiling. For this experiment, the Rotated Faster R-CNN model is used for the detection

because of its computational efficiency although has lower performance than ViTs. The goal in this experiment is not a higher mAP score so this model is the ideal. The mAP score after comparing different methods shows how much accuracy and efficiency have been improved by the various methods in this very application.

8.2 Optimization of SegTiling parameters

This experiment focuses on fine-tuning the parameters involved in the SegTiling method for further improvement of its object detection performance on the DOTA dataset using Rotated Faster R-CNN. The key objective is to investigate and adjust some of the key parameters that influence adaptive tiling and segmentation setting to find out which setting achieves the optimum result.
The parameters that have been optimized in this experiment include:

- Segmentation techniques (edge detection, color segmentation and combined method)
- Non-Maximum Suppression (NMS) with its overlap size.
- Overlap size between tiles.
- Min/max area ratios for bounding box selection.

Each of these parameters has to do with a trade-off between object detection precision and computational efficiency. The concept of this experiment is to see how changes in the settings could impact the generation of tiles and eventually alter the performance of object detection. Model fine-tuning consists of performing several sets of testing parameters by trial and error using the mAP score to find an optimal setting that maximizes object detection accuracy.

8.3 Segmentation-Based Adaptive Tiling (SegTiling)

In this experiment, the SegTiling methodology is evaluated on the DOTA dataset to test its performance. More specifically, the focus is on how segmentation techniques affect the behavior of the adaptive tiling mechanism. These different segmentations serve as drivers for the tiling mechanism to dynamically adjust, based on the image content, in order to enhance the capability of the system to focus with higher efficiency on portions of interest. Once the tiles are generated, they are fed into the object detection model, Rotated Faster R-CNN in this case to evaluate the potential benefits that this approach can bring in terms of performance of the detection. This experiment presents the analysis based on how different segmentation techniques may affect the efficiency of

the adaptive tiling approach. For comparison reasons, the mAP score after applying the standard tiling approach using no segmentation at all is presented in this experiment too.

8.4 Evaluation of object detection models

This experiment investigates the performance of the three examined object detection models: ViT with STD, HiViT with STD, and Rotated Faster R-CNN, applied to the DOTA dataset using SegTiling approach with the combined segmentation method. For comparison reasons standard tiling and no-tiling approaches are presented too. In the no-tiling approach, full-sized images ranging between 800x800 and 4000x400 pixels are down-scaled to 1024x1024 pixels in the final processing for detection due to computational limitations.
The objective is to evaluate the performance of these models coupled with the fine-tuned SegTiling method from previous experiments. Each of these models will be applied to the tiled images and assessed in their performance for object detection.
Mean Average Precision (mAP) scores are used for performance, enabling the comparison of how each model handles both the challenges of the dataset and the enhanced preprocessing method. This evaluation underlines the strengths and weaknesses of each model within the proposed tiling approach.

8.5 Application to real-world dataset

The following experiment is designed to confirm the practical applicability and generalizability of SegTiling by its application on a real-world dataset provided by Mainblades. This dataset, focused on aircraft drone inspection, has all the peculiar difficulties of different image resolutions, various object sizes, and poor-quality of the images.
The experiments are done with the application of SegTiling on this dataset in order to check how this technique including standard tiling and no-tiling methods will transfer from the DOTA dataset into a real-world scenario. This involves, testing the robustness of the approach in object detection like damage, text, and other features within images captured under practical conditions.
It also demonstrates, through such a validation process, the performance of the proposed method in real application scenarios outside the controlled experimental environment where aircraft inspection can be performed. That would also further validate the SegTiling approach to a wide variety of situations concerning how flexible and robust

the model is. For this experiment, the Rotated Faster R-CNN and HiViT with STD models are used to check the performance both on CNN and a transformer-based model.

8.6 Experiment with training data

This experiment evaluates the impact of applying a tiling approach during the training phase of object detection models. Unlike SegTiling used during inference, this approach focuses on tiling based on annotations (which are known in this case) to prepare the training and validation data.
In this setup, a script from the DOTA Devkit is employed to divide the images from DOTA and Mainblades datasets into tiles according to their annotations. This method ensures that each tile contains complete annotations, thereby reducing the risk of object fragmentation across tile boundaries. By doing so, the dataset is processed into smaller, manageable tiles while preserving the integrity of object annotations.

Two scenarios are compared in this experiment:

1. **Training on tiled images:** Here, each dataset is divided into tiles using an annotation-based approach before training. This approach attempts to help the object detection model learn from images that are divided in a way that is better for finding smaller objects without running into fragmentation issues or incomplete annotations.
2. **Training on full images:** In this scenario, the goal is to train the model directly on the high-resolution, full images without any tiling. For computational constraints, a downscaled version of them is trained eventually. The idea of this approach is to allow the model to learn from the complete context of the images, which might be beneficial for understanding larger spatial relationships.

This experiment, therefore, compares the performance of Rotated Faster R-CNN model chosen due to its computational efficiency, under two conditions on both DOTA and Mainblades datasets and it will investigate whether tiling the training data confers a significant advantage over training with full-resolution images using mAP score.

8.7 Impact of inference image size on mAP

In this experiment, the goal is to analyze how varying the size of input images during the object detection phase affects the performance of the mAP. Rotated Faster R-CNN is used for this experiment. The model is trained on both the DOTA and Mainblades

datasets on image tiles. However, during the object detection phase, the size of the input images is adjusted to assess the sensitivity of the model's performance, measured through mAP, to changes in image resolution.

This experiment investigates how reduction or increase in the size of the input image during inference affects the accuracy of object detection, given the different resolutions and contexts of the DOTA and Mainblades datasets. The mAP for each dataset is calculated across different image resolutions for further comprehension of how image size during inference affects model performance. It is aimed at establishing whether the model generalizes across different resolutions for these datasets. These different sizes of the inference images, allow for an analysis of whether what size of images provides a more accurate detection. This experiment hence serves the dual purpose of testing both the robustness of the SegTiling approach and its impact due to different image sizes during inference.

8.8 Impact of SegTiling on multi-scale objects

This experiment presents the effect of SegTiling on the detection performance for objects of varying sizes in high-resolution images from the DOTA dataset. The aim is to investigate the impact of SegTiling on the relative detection of smaller versus larger objects. In this experiment, the Rotated Faster R-CNN model was used and trained on the DOTA dataset in both training scenarios on image tiles and without tiles. This experiment compares two different image processing strategies. The first strategy utilized the original full-resolution images, simply downscaled without tiling, while the second applied SegTiling and thus divided the images into smaller tiles. No-tiling investigates how well the model is able to detect objects directly from full images. SegTiling tries to reduce the fragmentation problem for smaller objects with the goal improve the detection accuracy.
This experiment focused at object classes that are varied a lot in image size. The small objects, such as "small vehicle" and "plane", occupy comparably smaller portions in the high-resolution images, while the large objects of "roundabout" and "baseball-diamond" are generally in extensive areas. By comparing the detection performance for both small and large objects across the two tiling approaches, the goal is to demonstrate the extent to which SegTiling affects performance when taking into consideration different object sizes.

9

Results

In this section, the results of the experiments are evaluated through the mean Average Precision (mAP) metric. Each of the experiments conducted in this thesis is assessed based on how effectively the object detection models perform when combined with different baseline tiling approaches, SegTiling, and different pre-processing techniques or using different datasets. The performance results of these techniques are obtained after training has fully been completed using the validation set. It's worth mentioning that the tile size of 1024x1024 is selected as the optimal size, as the pre-trained models are trained on tile images of this dimension. In cases where no tiling is applied full images are used but due to computational constraints, a down-scaled version of them is used during inference in the end. Downscaling is one of the techniques used to process high-resolution images to fit the object detection models under limited computational resources.

9.1 Comparison of baseline tiling approaches: Results and Analysis

The following results provide insight into the use of tiling in handling large-scale images for object detection. Three tiling approaches were being compared: standard tiling, SAHI tiling, and without tiling. Table 9.1 gives the performance of these three different tiling approaches when applied to the DOTA dataset using the Rotated Faster R-CNN model. The metric used for comparison is the mean Average Precision (mAP), with higher values showing better object detection performance.

As Table 9.1 shows, **standard tiling** gives the highest mAP score, 69.10%. Standard tiling involves dividing an image into smaller tiles of fixed size, 1024 × 1024, with a significant overlap of 512 pixels to avoid fragmentation of objects at tile borders. Such high overlap ensures better object detection for small objects, which are abundant in DOTA.

Although **SAHI tiling** is an adaptive tiling technique, it lags behind Standard Tiling,

which has an mAP of 60.07%. This may also show that while SAHI tries to make a better optimization in tiling with the help of some search heuristic, it may be unable to keep object integrity as well as standard tiling does and detect multi-scale objects on the particular benchmark. It can be observed that **No Tiling** has the lowest mAP score with 33.69%. It gives an idea of how difficult it might be for models to identify objects in large-scale and high-resolution images without partitioning them. Without tiling, the model is not be able to catch the subtlety of the image, which will lead to poor detection, specially for the small-size objects that may spread over a large portion of the image. For this experiment, the full images are used at the time of inference, downscaled to 1024 × 1024 pixels in the final processing while the model is trained on tiles of this resolution.

Tiling Approach	Tiling Exec. Time	Images size (pixels)	Overlap size	mAP (%)
Standard Tiling	382.0 s	1024x1024	512	**69.10**
SAHI Tiling	201.8 s	1024x1024	512	60.07
No Tiling	N/A	800x800-4000x4000	N/A	33.69

Table 9.1: *Comparison of mAP scores and tiling execution times (in seconds) for different tiling approaches on the DOTA dataset using Rotated Faster R-CNN model. N/A in the table stands for not-applicable. Higher mAP means better performance.*

The results show the importance of tiling in large images when performing object detection tasks. The overall best performance was from Standard Tiling since it has a balanced approach in dividing images while maintaining the integrity of objects even if it is a simple tile approach without any complex procedures. Standard tiling takes somewhat longer to execute than SAHI, however, the benefit outweighs this. In this case, the execution time refers to the generation time of the tiles. This performance drop using no tiling shows heavy processing is involved when an image is being processed as a whole and not with its pieces. While SAHI tiling does try to optimize this, the Standard Tiling approach still outperforms due to its consistent tile size and overlap.

9.2 Optimization of SegTiling Parameters: Results and Analysis

This experiment investigates the sensitivity of the SegTiling approach, using the Rotated Faster R-CNN, towards various parameters using the three different segmentation techniques combined method, edge detection, and color segmentation, that will lead to an optimal configuration for maximum detection accuracy. The best configuration is sought by fine-tuning the parameters that control tile size and its overlap, Non-Maximum Suppression (NMS) parameters, and area ratio thresholds. Experiments are conducted and their results presented on how these changes in parameters affect the mAP scores

to provide some insight into the sensitivity and efficiency of SegTiling.
Results in the table 9.2 are based on fixed **tile size** of 1024x1024 pixels. **Overlap sizes** were manipulated in an attempt to study their impacts on model performance. Small Overlap size of 100 pixels yielded comparable mAP as compared to an overlap size of 512 pixels. This indicates that although the overlap is critical in dealing with objects spanning across tile borders, too large overlaps do not contribute to an improvement in the detection performance.

Segmentation Technique	Overlap Size	NMS/overlap size	Min Area Ratio	Max Area Ratio	mAP (%)
Combined Method	100	No	N/A	N/A	69.72
Combined Method	512	Yes/0.1	0.00001	0.01	68.60
Combined Method	512	Yes/0.4	0.00001	0.01	68.59
Combined Method	256	No	N/A	N/A	69.71
Combined Method	512	Yes/0.1	0.000001	0.01	69.22
Combined Method	512	No	N/A	N/A	**69.79**
Edge Detection	512	Yes/0.1	0.00001	0.01	67.69
Edge Detection	512	No	N/A	N/A	69.76
Color Segmentation	512	Yes/0.1	0.00001	0.01	58.01
Color Segmentation	512	No	N/A	N/A	69.54

Table 9.2: *Impact of different parameters on the mAP using the Rotated Faster R-CNN model on different segmentation techniques on the DOTA dataset. N/A in the table stands for not-applicable. Higher mAP means better performance.*

NMS should clean and refine bounding boxes by suppressing overlapping boxes. In theory, this is supposed to give better detection accuracy. For this experiment, though, the NMS did nothing to improve the performance and actually reduced mAP in some cases. Because of this, for this kind of tiling approach, keeping more bounding boxes despite the possibility of overlaps improved results. That would mean NMS could be less useful in the case of adaptive tiling trying to capture all possible object instances for tile division.

The **Min Area Ratio** and **Max Area Ratio** parameters control the selection of bounding boxes based on their relative area to the total image. Experiments demonstrated that the variation of these ratios had negligible influence on mAP. This suggests that the size of the bounding boxes, relative to the total area of the image, did not have a significant effect on the detection performance in this framework.
The highest mAP of 69.79% was achieved with no NMS and an overlap size of 512 pixels using combined method as the segmentation technique. More experiments are done for combined method because it shows promising results. A high mAP score was achieved

using edge detection too but it is slightly lower than with combined method. This configuration suggests that for the Rotated Faster R-CNN model with the DOTA dataset, avoiding NMS and using a moderate overlap size resulted in the best performance. It emphasizes that the inclusion of as many bounding boxes as possible, rather than refining them, can be advantageous for adaptive tiling.

9.3 Performance of SegTiling: Results and Analysis

In this experiment, the performance impact of different segmentation techniques, namely, Edge Detection, Color Segmentation, and Combined Method, will be considered within the proposed adaptive tiling approach. The goal is to investigate how these segmentation methods perform in enhancing object detection accuracy by fine-tuning the process of tile generation with minimal object fragmentation. Table 9.3 presents the mAP scores for the Rotated Faster R-CNN model when different segmentation techniques are applied in the context of adaptive tiling.

Segmentation Technique	Tiling Approach	Tiles Size (pixels)	Overlap size	mAP (%)
Edge Detection	SegTiling	1024x1024	512	69.76
Color Segmentation	SegTiling	1024x1024	512	69.54
Combined Method	SegTiling	1024x1024	512	**69.79**
No Segmentation	Standard Tiling	1024x1024	512	69.10

Table 9.3: *Comparison of segmentation techniques within SegTiling approach using Rotated Faster R-CNN on the DOTA dataset. Higher mAP means better performance.*

It can be observed from the results that the **Combined Method** for segmentation gives the highest mAP score of 69.79%, outperforming Edge Detection and Color Segmentation techniques. This suggests that the combination of these two segmentation methods develops the model to give more fine results for object detection.

For **Edge Detection**, the mAP attained is 69.76%, slightly lower than the Combined Method but still higher than what was achieved by **Color Segmentation**. Among the segmentation techniques, the worst-performing one is Color Segmentation, which has an mAP of 69.54%. It is clear from the Table 9.3 that the worst performance even slightly lower has the standard tiling with 69.10% approach using no segmentation, which is expected as a tiling approach using no segmentation teqhniques.

That means the leveraging of both edge information and color information probably does give a more complete understanding of the objects, hence the better detection performance. The time taken to execute this tiling method is approximately **564.9 seconds** among all the segmentation techniques including the combined method with

slight and not important differences, which is slightly longer when compared to the other tiling methods, but here performance gains outweigh the increase.

9.4 Evaluation of Object Detection Models: Results and Analysis

The models that evaluate object detection models under this experiment have been applied to the DOTA dataset processed by the Segtiling approach in order to compare performances among ViT and STD, HiViT and STD, and Rotated Faster R-CNN. Standard tiling and no-tiling techniques have also been included in the table for comparison purposes. The results illustrate how each model performs with the optimized preprocessing approach. For SegTiling the combined segmentation method has been used. In the no-tiling approach, full-sized images down-scaled to 1024x1024 pixels are used in the final processing for detection.

Model	Tiling Approach	# of images/tiles	Images/tiles size	Overlap Size	mAP (%)
ViT with STD	Standard Tiling	14222	1024x1024	512	95.25 *
ViT with STD	**SegTiling**	**9977**	1024x1024	512	**95.43** *
ViT with STD	No tiling	458	800x800-4000x4000	N/A	56.58
HiViT with STD	Standard Tiling	14222	1024x1024	512	95.32 *
HiViT with STD	**SegTiling**	**9977**	1024x1024	512	**95.77** *
HiViT with STD	No tiling	458	800x800-4000x4000	N/A	58.08
Rotated Faster R-CNN	Standard Tiling	14222	1024x1024	512	69.10
Rotated Faster R-CNN	**SegTiling**	**9977**	1024x1024	512	**69.79**
Rotated Faster R-CNN	No tiling	458	800x800-4000x4000	N/A	33.69

Table 9.4: *Comparison of object detection models using tiling techniques and without tiling during inference on the DOTA dataset, trained on tiles of 1024x1024 pixels. N/A in the table stands for not-applicable. Higher mAP means better performance. Images/tile size resolution are in pixels. The results marked with (*) are obtained using the checkpoints of the models which are trained on trainvalidation-set and tested on validation-set this is why the high score.*

Model	Tiling Approach	# of images/tiles	Images/tiles size	Overlap Size	mAP (%)
HiViT with STD	**SegTiling**	19636	1024x1024	512	81.74
HiViT with STD	Multi-scale setting	72000	682x682-2048x2048	500	**82.24**

Table 9.5: *Comparison of SegTiling with state-of-the-art (SOTA) mAP score. The model trained on trainvalidation-set and tested on the test-set. The SOTA used multi-scale setting to split the images into patches/tiles at different scales and processed the tiles with 1024x1024 size during object detection.*

Tables 9.4 and 9.5 demonstrate several key points:

Impact of tiling on model performance: Across all three models (ViT with STD, HiViT with STD, and Rotated Faster R-CNN), using tiling methods-whether standard or adaptive-significantly improved the mAP scores compared to the no-tiling approach as

shown in Table 9.4. This highlights the importance of tiling in handling large images and detecting smaller objects, particularly for high-resolution datasets like DOTA.

Comparison between Standard and SegTiling: SegTiling slightly outperformed standard tiling in all three models in Table 9.4, suggesting that the proposed adaptive tiling approach optimizes object detection performance by better preserving object boundaries and minimizing object fragmentation across tiles.

Comparison of SegTiling with state-of-the-art(SOTA) on test-set: SegTiling achieves an mAP score of 81.74% using tiles size of 1024x1024 pixels and an overlap of 512 pixels, resulting in a total of 19,636 tiles. This approach minimizes the total number of processed tiles while effectively balancing spatial coherence and computational efficiency. SOTA using multi-scale tiling performs a slightly higher mAP of 82.24%, employing a range of tile sizes from 682x682 to 2048x2048 pixels with an overlap of 500 pixels. However, this comes at the cost of a significantly larger number of tiles, approximately 72,000, due to multi-scale splitting.

Computational trade-offs: While the tiling mechanism increases performance, the number of images/tiles that need to be processed by the model also increases. For example, using standard tiling and SegTiling generated approximately 14,222 and 9,977 tiles, respectively, compared to 458 images with no tiling as shown in Table 9.4. Similar to Table 9.5 which test-set is used, SegTiling generated 19636 tiles while SOTA using multi-scale tiling generated a high number of approximately 72000 tiles with different scales. The more tiles, the more computations will be needed, this might lead to longer times for inference.

Effectiveness of SegTiling: Compared to the standard tiling approach, SegTiling produces fewer tiles, 9,977 versus 14,222. Its scores increase with higher mAP, peaking at 95.77% for the HiViT model in Table 9.4. Compared to multi-scale tiling used in SOTA official paper [34] SegTiling generated much fewer tiles and scored the comparable performance of 81.84% which is 0.5% less than state-of-the-art as shown in Table 9.5. It means that SegTiling not only does reduce computational overhead significantly but also scores a comparable performance in object detection tasks.

9.5 Application to real-world Dataset: Results and Analysis

In this experiment, both the Rotated Faster R-CNN and HiViT with STD models are evaluated on the Mainblades dataset to assess how well the SegTiling approach performs during the inference phase. The models were trained on tiles of 1024x1024. The focus of

this comparison is on mAP scores and the number of tiles generated during inference, comparing standard tiling, SegTiling, and no tiling.

As shown in table 9.6, SegTiling demonstrates both higher accuracy and efficiency compared to standard tiling and no tiling. For the Rotated Faster R-CNN, SegTiling reduces the number of tiles (14,753 vs. 20,840) while achieving a slightly higher mAP of 23.94%. In the meantime, HiViT with STD shows an increase in mAP: from 29.43% with standard tiling to 30.95% with SegTiling, underlining the benefits brought by the segmentation-based approach. The less number of tiles decreases the computation burden without performance loss. In contrast, the no tiling approach, which used a down-scaled version of the images too for the inference with a size of 1024x1024 pixels, consistently resulted in the lowest mAP values for both models.

Model	Tiling Approach	# of images/tiles	Images/tiles Size	Overlap Size	mAP (%)
Rotated Faster R-CNN	Standard Tiling	20840	1024x1024	512	23.57
Rotated Faster R-CNN	**SegTiling**	**14753**	1024x1024	512	**23.94**
Rotated Faster R-CNN	No Tiling	514	800x800-8192x5460	N/A	16.00
HiViT with STD	Standard Tiling	20840	1024x1024	512	29.43
HiViT with STD	**SegTiling**	**14753**	1024x1024	512	**30.95**
HiViT with STD	No Tiling	514	800x800-8192x5460	N/A	24.42

Table 9.6: *Comparison of SegTiling, Standard Tiling, and No Tiling using Rotated Faster R-CNN, HiViT with STD on the Mainblades dataset during inference. The models are trained on tiles of 1024x1024 pixels. N/A in the table stands for not-applicable. Higher mAP means better performance. Images/tile size resolution are in pixels.*

Rotated Faster R-CNN achieved only 16.00% mAP, while HiViT with STD performed slightly better at 24.42%, but both results indicate that tiling is essential when working with high-resolution images in object detection tasks.

The overall low accuracy over all cases is expected, since the Mainblades dataset is from real-world scenarios, presenting challenges from image quality to illumination and environmental factors. Despite such inherently challenging conditions, SegTiling gives a slight advantage over standard tiling and even more so from no tiling on both models, hence would thus seem to improve the performance of object detection under such difficult conditions by its capability to generate more contextually aware tiles.

9.6 Experiment with training data: Results and Analysis

The idea behind this experiment is to test the performance of tiling during training, in comparison with the original direct training of full images using an annotation-based tiling scheme from DOTA Devkit. This strategy tries to avoid object fragmentation, and each tile has complete annotations. By using a Rotated Faster R-CNN model, the

performance was measured in terms of mAP, with the intent to determine how the tiling affects the results during training. Originally, by dividing the dataset into 1024x1024 pixel tiles with a 512-pixel overlap, the model is provided with training data that maintains the spatial integrity of the objects. This contrasts with second training scenario used, where the model is trained on the full, downscaled images at 3240x2160 pixels. The latter approach, while retaining the entire image context, faces challenges in effectively detecting smaller objects and managing large-scale images, which can reduce accuracy and increase computational load.

Training Scenario	Dataset	Train. Image Size	Tiling Approach	Inf. Image Size	mAP (%)
Annotation-based tiling	Mainblades	1024x1024	No Tiling	1024x1024	16.00
Full images (downscaled)	**Mainblades**	3240x2160	**No Tiling**	3240x2160	**28.79**
Annotation-based tiling	Mainblades	1024x1024	SegTiling	1024x1024	23.94
Full images (downscaled)	Mainblades	3240x2160	SegTiling	1024x1024	22.22
Annotation-based tiling	DOTA	1024x1024	No Tiling	1024x1024	33.69
Full images (downscaled)	DOTA	3240x2160	No Tiling	3240x2160	35.71
Annotation-based tiling	**DOTA**	1024x1024	**SegTiling**	1024x1024	**69.79**
Full images (downscaled)	DOTA	3240x2160	SegTiling	1024x1024	43.42

Table 9.7: *Comparison of training on tiled images vs. full images using Rotated Faster R-CNN on the Mainblades and DOTA dataset presenting mAP on the inference. For the inference, no tiling or SegTiling is applied. Higher mAP means better performance. Train. Image Size stands for the size resolution of images during training. Inf. Image Size stands for the size resolution of images during inference (object detection phase).*

The experiment's results are noted in Table 9.7, comparing mAP values between the two training scenarios for both Datasets. For the cases where SegTiling is applied the inference image size is the same as the generated tiles size. One of the main observations here could be that without tiling at inference, the performance differences is not important between training on tiles and training on full images. In the specific case of the Mainblades dataset, for instance, mAP is 16.00% when trained on annotation-based tiling and 28.79% when trained on full images.

On the other hand, when tiling is applied in inference, the advantages of training with tiles become more evident, especially for the DOTA dataset. Take for instance the DOTA dataset, which has trained using annotation-based tiling and SegTiling on inference, giving a mAP of 69.79%, compared to training on full images and using SegTiling at inference, yielding only a mAP score of 43.42%. That would mean it is fully aware of the benefits of training with the tiles, only if tiling is applied at the time of inference too, especially on unstructured datasets such as Mainblades, where the full image context may be better preserved.

Interestingly, another trend follows in the case of Mainblades, which includes real-world

images. Although the use of SegTiling during inference improves performance when the model is trained on tiles (achieving an mAP of 23.94%), the highest performance is achieved without tiling during inference but training on full images which reaches a mAP of 28.79%.

9.7 Impact of inference image size on mAP: Results and Analysis

This experiment plots the mAP performance of the Rotated Faster R-CNN model versus the change in input image size during the inference phase. The model was trained on 1024x1024 image tiles from both datasets, DOTA, and Mainblades, and the inference is done at various image sizes resizing the size of images with the goal of seeing how scaling affects object detection performance.

Dataset	Tiling Approach	Inf. Image Size (pixels)	Overlap Size (pixels)	# of images	mAP (%)
DOTA	No Tiling	3240x2160	N/A	458	52.54
DOTA	SegTiling	1152x1152	256	4601	**70.63**
Mainblades	No Tiling	3240x2160	N/A	514	**28.04**
Mainblades	SegTiling	2160x2160	512	2323	27.80

Table 9.8: *Impact of varying inference image sizes on mAP for DOTA and Mainblades datasets using Rotated Faster R-CNN. The model is trained on tiles of 1024x1024 pixels. N/A in the table stands for not-applicable. Higher mAP means better performance. Inf. Image Size stands for the size resolution of images during inference (object detection phase).*

As shown in Table 9.8, although originally trained on smaller tiles, the results show significant increases in mAP scores when larger inference image sizes are applied. The mAP for the Mainblades dataset is almost doubled from 16.00% (as seen in Table 9.7) to 28.04% without tiling when the size of inference images is increased from 1024x1024 to 3240x2160. That means having larger image regions during inference would be helpful and provide enough context to the model. This in fact, helps object detection on difficult real-world datasets such as Mainblades.

Speaking of DOTA, an impressive mAP of 52.54% can be achieved using an inference size of 3240x2160 with no tiling whereas the previous figure was 33.69% as was shown in Table 9.4. These results highlight that the modification of image resolution at inference greatly improves model performance by balancing context and resolution in object detection tasks, although the model was trained on smaller-sized tiles compared to those at inference.

9.8 Impact of SegTiling on multi-scale objects: Results and Analysis

The application of SegTiling demonstrates great improvements in Average Precision (AP) across both small and large objects in the DOTA dataset. As a tiling approach, SegTiling was supposed to perform better on small object detection, however, it can be noticed that larger objects like "roundabout" with 26.53% improvement and "baseball-diamond" with 56.26% improvement have significantly benefited from this approach as shown in 9.9.

In high-resolution aerial images, such as in DOTA, objects like "small vehicle," "large vehicle," and even "plane" can still occupy relatively small pixel areas according to the image resolution. SegTiling helps preserve object integrity during inference by breaking the image down into smaller, more digestible tiles. That avoids loss of detail that normally affects the detection when processing entire high-resolution images without tiling.

Class	No Tiling (AP %)	SegTiling (AP %)	Improvement (%)
small vehicle	35.90	65.04	+29.14
large vehicle	40.21	74.84	+34.63
roundabout	39.50	66.03	+26.53
plane	75.06	91.78	+16.72
baseball-diamond	18.39	74.60	+56.21

Table 9.9: *Comparison of Average Precision(AP) for different sizes of object classes from the DOTA dataset using SegTiling and No Tiling. The "Improvement" column shows the increase in AP for each class. The Rotated Faster R-CNN model used and trained on both training scenarios tiles and without tiles and applied on SegTiling and No Tiling on inference respectively.*

However, the results also highlight that object size alone does not determine the impact of SegTiling, and improvements for larger objects such as "roundabout" and "baseball-diamond" are also important. That would suggest that SegTiling lies in handling high-resolution images well, whether for small or large objects.

10 Discussion

10.1 Interpretation of results

The experiments performed within this thesis are aimed to evaluate the effectiveness of the proposed SegTiling approach with different tiling methods including standard tiling, SAHI tiling and no tiling. Obtained results within two datasets DOTA and Mainblades using object detection models gives valuable insight into the flexibility of SegTiling in a wide range of object detection tasks applied to both training and object detection phases. This section provides a detailed commentary on the findings presented in the results section focusing on how different tiling strategies, inference image sizes, and training approaches affect the final mAP scores. This is not only necessary to understand the strengths but also the limitations of the SegTiling approach for applicability to a wider range of object detection tasks.

10.1.1 Impact of SegTiling on performance

SegTiling shows a clear advantage over different tiling apporaches including standard tiling, sahi tiling, and no tiling, mainly for the datasets like DOTA and Mainblades. According to the results in Table 9.4, in those structured datasets like DOTA, whose images are composed of clear objects in the aerial views and high quality images, SegTiling is able to show a significant improvement both in mAP and computational efficiency. Compared with the SOTA on the test-set SegTiling performs a comparable score even slightly lower, with high advantages on computational efficiency as shown in Table 9.5. Since SegTiling is adaptive, it generates fewer tiles that are more contextually meaningful, achieving less fragmentation of objects, than other tiling methods do. This maintains the spatial coherence which is so important for accurate object detection.
Regarding the more complex Mainblades dataset, SegTiling again achieved performance gains over the other tiling methods but less dramatically than those reported in DOTA.

These results even with lower improvement reflect its adaptability of SegTiling for handling real-world images, which are more difficult because of the variation in objects size and environmental factors like light conditions or poor quality of images. It may not be much of an improvement, but considering the extra complexity using real-world dataset and resource efficiency gained by reducing the number of tiles during inference, it is important.

10.1.2 Application to real-world dataset (Mainblades)

SegTiling gives a slight advantage over standard tiling and even more so from no tiling under challenging real-world dataset. In Mainblades dataset the detection of objects is more difficult compared to DOTA due to inconsistent lighting, occlusion of objects, and variations in quality. While SegTiling performed well in Mainblades, it was not as good as in DOTA due the complexity of the images. Its robust results as shown in Table 9.6 underpinning its potential in real-world applications. The approach's adaptability and ability to handle imperfect data make it well-suited for industrial use cases where data quality is less controlled.

10.1.3 Training and inference observations

Applying tiling strategies during training and inference is critical for maintaining high mAP score. The results from the experiments investigating training with tiles versus full images provide further insight into the importance of matching the training and inference strategies. Regarding DOTA dataset, tiling at training and applying SegTiling at inference resulted in a significant gain (69.79% mAP as per Table 9.7). This result suggests that the model is better positioned for handling tiled images during inference when tiling maintains spatial structure and annotations in the training data. In contrast, training on full images of size 3240x2160 pixels and using tiling , SegTiling in this case during inference yielded a lower mAP of 43.42%, shown in Table on DOTA, thus indicating that a mismatch between training and inference will degrade performance.

On the contrary, the results using the Mainblades dataset tell a slightly different story. When the model was trained on full images and then tested with larger images during inference (no tiling), the mAP was significantly higher (28.79% as shown in Table 9.7) compared to training and inferring on tiles (16.00% as shown in Table 9.7). This suggests that the nature of the Mainblades dataset, with larger objects like text or markings on planes, is better suited to full-image analysis rather than tiling when focusing on larger objects. Tiling during inference, while beneficial for some objects specifically

small objects, may fragment larger objects that span the entire image, thus negatively impacting of mAP score.

Increasing the size of inference images often enhances detection accuracy, especially for larger objects. Another observation of these experiments is the impact of the size of the inference image on mAP score. The two datasets have a better performance with large inference images compared to the training tiles. So much so that increasing the size of the inference image in Mainblades to 3240x2160 pixels almost doubles the mAP to 28.04%, as shown in Table 9.8, compared to smaller tiles, 16.00%, as shown in Table 9.7. The same trend followed DOTA, where inferring using higher resolution images without tiling in inference yielded a higher mAP of 52.54% as shown in Table 9.8 compared to inferring using less resolution with mAP of 33.69% as shown in Table 9.7. These results suggest that for some datasets, especially those containing larger objects, the performance of object detection for a model improves when increasing the resolution of inference images even beyond their training resolution and without applying tiling.

10.1.4 Impact of SegTiling on multi-scale objects

The analysis of SegTiling allows to extract several observations from the results. One of the main issues in methods with segmentation, such as SegTiling, is their dependence on the visual features of an image, like brightness, contrast, and general resolution. This may affect the detection performance for some object classes. For example, "small-vehicle", or "plane" cannot be detected properly due to different image conditions as shown in figures 10.1 and 10.2.

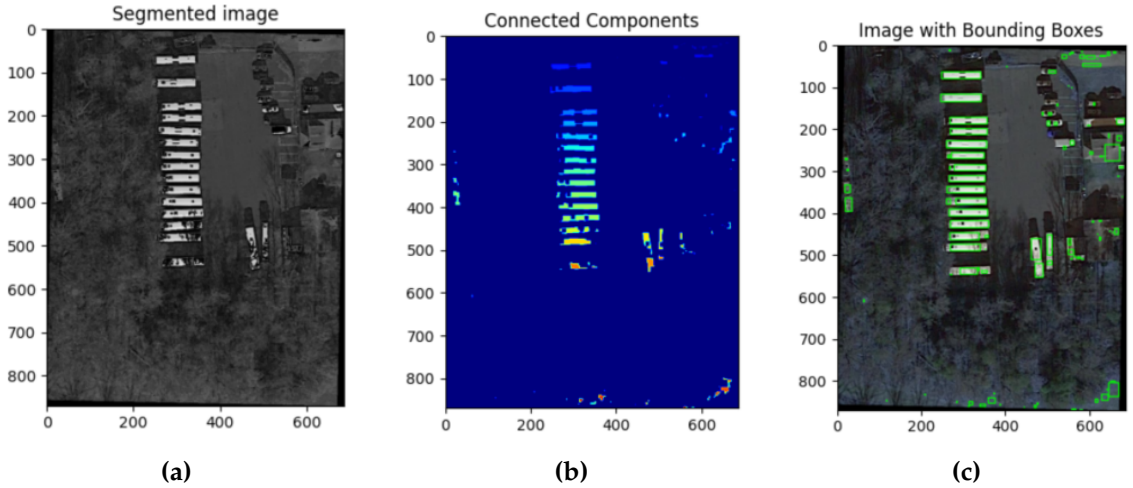

Figure 10.1: *An example of image during preprocessing phase representing two classes "small-vehicle" and "large-vehicle" on DOTA dataset.*

It is clear from the Figures 10.1b and 10.1c that the small vehicles are not fully captured

with compared to the large vehicles. In the case of the class 'plane' as shown in Figures 10.2b and 10.2c, it is also evident that the planes are not clearly captured and separated from other objects. The bounding boxes thus fail to properly fit to planes and often get confused with other objects. High brightness or poor quality may result in problems of segmentation algorithms to identify the boundaries of the objects, therefore tiling lead to poor detection performance. While SegTiling obtained high mAP score, it is still very sensitive to the clarity of the images, which indicates that image quality, lighting conditions can play a key role to its success.

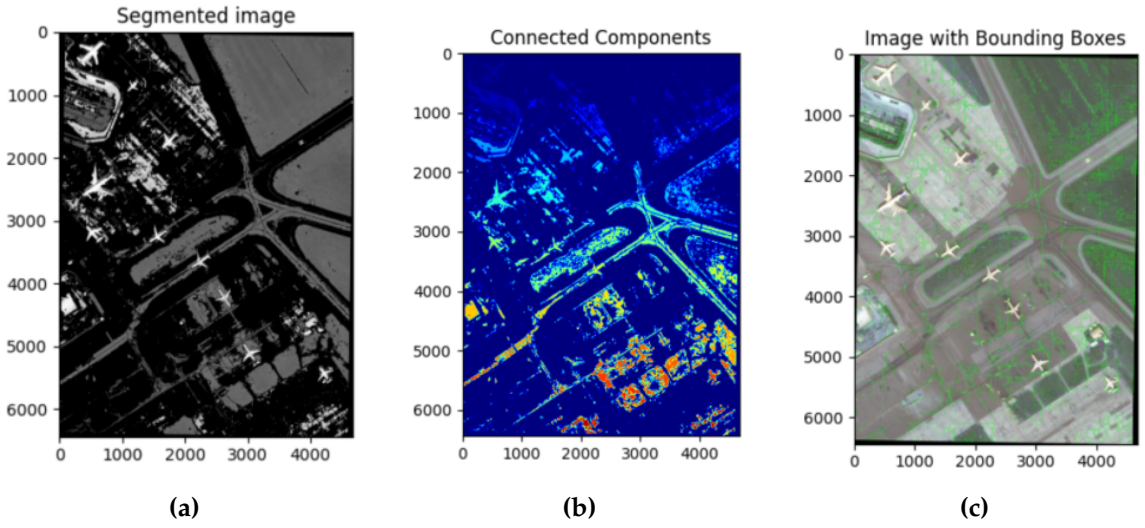

Figure 10.2: *An example of image during preprocessing phase representing some classes including "plane" on DOTA dataset.*

Generally, SegTiling shows a really good performance detecting small object in high-resolution images. The high average precision (AP) improvements for "small vehicle" and "large vehicle" as shown in Table 9.9 further validate the proposed method on small objects in high-resolution images. Sometimes "large" objects like "large vehicle" are still small and take a small area in a high resolution images. Interestingly, SegTiling also benefits larger objects, such as "roundabout" and "baseball-diamond," which have notable AP improvements too. This highlights the extensive utility of SegTiling, as it can preserve object context and spatial integrity for high-resolution images and enabling its effectiveness across variable object scales. This consistent performance gain through different object sizes underlines the robustness of the method in dealing with the challenges on multi-scale object detection.

10.2 Limitations

Although SegTiling has demonstrate promising results with respect to enhancing the mAP performances together with computational efficiency, a number of limitations do exist in this thesis presented in this section.

- **Inference-Time Limitations:** The advantages of SegTiling become visible mainly in inference, especially in generating fewer number of tiles and hence less computation. However, when SegTiling is applied only to the object detection phase instead of both training and inference the performance is not really improved. That means SegTiling works better when i combined with training on tiles too. This limitation it is not very useful for many practical applications when retraining of the model is not feasible.
- **Computational Cost of Training and Inference:** Both phases training and object detection in SegTiling, suffered a considerable rise in computational cost using tiling approaches. Accordingly, the training time almost doubled compared with training using full images that were down-scaled because tiling creates a larger number of samples and requires more processing power. Regarding the use case of inference, splitting high-resolution images into tiles greatly improves mAP but increases the processing time in such cases. This increased computational burden becomes particularly important in real-time or resource-constrained environments, where need to balance the trade-off between improved performance and higher computational costs.

10.3 Future Work

This thesis offers several potential directions for improvement and further exploration. It is already pointed out that SegTiling has some potential for use in well-annotated datasets like DOTA and real-world dataset for multi-scale object detection. However, there are ways in which SegTiling can be improved, and there are certain ways in which SegTiling can be used for certain types of detection tasks that could be part of future work. The following directions are proposed to refine and extend the scope of this thesis.

10.3.1 Generalizing SegTiling to other Datasets:

In future work, the implementation of SegTiling could be expanded to a wider selection of datasets that are not limited to aerial and industrial images. For instance, assessing SegTiling on datasets of medical images, satellite images, or natural scenes would

provide a more thorough appraisal of its robustness and versatility. Understanding how well SegTiling performs in these different domains will help establish it as a general-purpose tiling solution.

10.3.2 Tiling for Large Object Detection:

The issues that run into large object fragmentation, especially in the Mainblades dataset, show that developing more sophisticated tiling methods for detection is needed that can handle large objects. For example, methods that allow tiles to merge or split based on the object's dimensions could help maintain spatial coherence and prevent detection failures caused by fragmented objects. This would be especially useful for datasets featuring large, irregularly shaped objects.

10.3.3 Integration into architectures in Object Detection Models:

Even though the SegTiling method was applied to an existing object detection pipelines , particullary for the Rotated Faster R-CNN, ViT with STD and HiVit with STD, future work could examine integrating SegTiling more closely into the architecture of object detection models. This would entail end-to-end training of the model, where the object detection network itself learns to make the decisions that SegTiling currently makes and generate tiles dynamically based on the input data for optimal results. This could involve modifying architectures like Faster R-CNN or ViT models to internally manage tile creation, allowing for more seamless and optimized processing during both training and inference.

10.3.4 Optimization for Real-Time Applications:

Although SegTiling has been shown to improve object detection performance, it can be computationally demanding, especially for real-time applications. Future researchers could focus to figure out how to make SegTiling less computationally heavy and faster, as a result, more viable for real-time applications. Tile skipping or early exit strategies could be used to avoid processing tiles with are not relevant reducing overhead. The integration of various of these techniques with parallel processing and GPU acceleration would enable fast tile processing, while edge computing frameworks could distribute the computational load to reduce resource demands on real-time applications.

11 Conclusions

This thesis outlines a new approach to elevate the capabilities of object detection on high-resolution images including small, oriented multi-scale objects. Segmentation-based Adaptive Tiling (SegTiling) aims to lessen the limitations of conventional tiling techniques like standard tiling and SAHI tiling by minimizing the object fragmentation and spatial coherence, two important factors that lead to a better performance if are avoided. The proposed SegTiling approach is evaluated against tiling and no-tiling methods through extensive experimentation on the public available DOTA dataset and the real-world confidential Mainblades dataset, a company specializing in aircraft drone inspections.

The results of SegTiling exhibits statistically significant mAP score improvements, besides marked gains in computational efficiency. In well-annotated and high-quality images datasets like DOTA , SegTiling, marks its feasibility and high performance. Even with the Mainblades dataset, wherein real-world images present varying quality and environmental conditions, SegTiling retained robustness. Despite the inherent complexities, SegTiling maintained strong performance, highlighting its potential for practical applications.

The novel contributions of this thesis include:

- Development of SegTiling, an adaptive tiling method that maintains spatial consistency and minimizes object fragmentation.
- Superior mAP performance achieved over standard tiling, SAHI tiling, and no-tiling methods on high-resolution images.
- SegTiling's robust application to Mainblades dataset also illustrates its practicality for handling poor-image quality in the real world.
- SegTiling performs 0.5% lower than state-of-the-art (SOTA) on mAP score on DOTA dataset with a significant reduction of generated tiles compared to multi-scale tiling used by SOTA.

- Significant computational efficiency gains by reducing the number of tiles processed during inference .

Although promising, SegTiling is inhibited by notable limitations on inference time and computational demands, which are common challenges related with tiling strategies. While during the inference phase the given method improves detection accuracy, the best performances are obtained when tiling is simultaneously applied during the training phase too. However, this imposition on computation makes real-time applications harder to realize. Future work could optimize SegTiling for faster inference in order to expand its use to real-time tasks.
In future work, it could be valuable to:

- Generalizing SegTiling to a larger set of datasets besides aerial and industrial images.
- Improving tiling methods for detection in order to facilitate the handling of large objects. Potential methods allowing for merging or splitting of tiles based on an object's dimensions may reduce fragmentation issues in object-detection tasks.
- Integrate SegTiling directly into the architectures of object detection models to allow for end-to-end training. Use SegTiling to generate tiles dynamically based on the input data of the network for optimal results.
- Improve the approach to real-time applications by enabling tile skipping or early exit strategies for non-interesting tiles, rendering it appropriate for tasks including drone inspection and live video surveillance.

This would position SegTiling as a powerful tool for object detection and open the route to further performance improvement regarding wide ranges of practical applications.

Bibliography

[1] Fatih Cagatay Akyon, Sinan Onur Altinuc, and Alptekin Temizel. "Slicing Aided Hyper Inference and Fine-Tuning for Small Object Detection". In: *2022 IEEE International Conference on Image Processing (ICIP)*. 2022, pages 966–970. DOI: 10.1109/ICIP46576.2022.9897990 (cited on pages 11, 17, 24).

[2] Saikat Basu, Sangram Ganguly, Supratik Mukhopadhyay, Robert DiBiano, Manohar Karki, and Ramakrishna Nemani. "DeepSat: a learning framework for satellite imagery". In: *Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems*. SIGSPATIAL '15. Seattle, Washington: Association for Computing Machinery, 2015. ISBN: 9781450339674. DOI: 10.1145/2820783.2820816. URL: https://doi.org/10.1145/2820783.2820816 (cited on page 12).

[3] Ilker Bozcan and Erdal Kayacan. "AU-AIR: A Multi-modal Unmanned Aerial Vehicle Dataset for Low Altitude Traffic Surveillance". In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. 2020, pages 8504–8510. DOI: 10.1109/ICRA40945.2020.9196845 (cited on page 12).

[4] G. Bradski. "The OpenCV Library". In: *Dr. Dobb's Journal of Software Tools* (2000) (cited on pages 28, 29, 31).

[5] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. "End-to-End Object Detection with Transformers". In: *Computer Vision – ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I*. Glasgow, United Kingdom: Springer-Verlag, 2020, pages 213–229. ISBN: 978-3-030-58451-1. DOI: 10.1007/978-3-030-58452-8_13. URL: https://doi.org/10.1007/978-3-030-58452-8_13 (cited on page 16).

[6] Libo Chang, Shengbing Zhang, Huimin Du, Yue Chen, and Shiyu Wang. "A Reconfigurable Neural Network Processor With Tile-Grained Multicore Pipeline for Object Detection on FPGA". In: *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 29.11 (2021), pages 1967–1980. DOI: 10.1109/TVLSI.2021.3109580 (cited on page 17).

[7] Charleen, Cheryl Angelica, Hendrik Purnama, and Fredy Purnomo. "Impact of Computer Vision With Deep Learning Approach in Medical Imaging Diagnosis". In: *2021 1st International Conference on Computer Science and Artificial Intelligence (ICCSAI)*. Volume 1. 2021, pages 37–41. DOI: 10.1109/ICCSAI53272.2021.9609708 (cited on page 12).

[8] Weiming Chen, Bing Han, Zheng Yang, and Xinbo Gao. "MSSDet: Multi-Scale Ship-Detection Framework in Optical Remote-Sensing Images and New Benchmark". In: *Remote Sensing* 14.21 (2022). ISSN: 2072-4292. DOI: 10.3390/rs14215460. URL: https://www.mdpi.com/2072-4292/14/21/5460 (cited on pages 14, 15).

[9] Linhui Dai, Hong Liu, Hao Tang, Zhiwei Wu, and Pinhao Song. "AO2-DETR: Arbitrary-Oriented Object Detection Transformer". In: *IEEE Trans. Cir. and Sys. for Video Technol.* 33.5 (May 2023), pages 2342–2356. ISSN: 1051-8215. DOI: 10.1109/TCSVT.2022.3222906. URL: https://doi.org/10.1109/TCSVT.2022.3222906 (cited on page 15).

[10] Jian Ding, Nan Xue, Yang Long, Gui-Song Xia, and Qikai Lu. "Learning RoI Transformer for Oriented Object Detection in Aerial Images". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2019 (cited on page 14).

[11] Xingshuai Dong and Massimiliano L. Cappuccio. *Applications of Computer Vision in Autonomous Vehicles: Methods, Challenges and Future Directions*. 2024. arXiv: 2311.09093 [cs.CV]. URL: https://arxiv.org/abs/2311.09093 (cited on page 12).

[12] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale". In: *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL: https://openreview.net/forum?id=YicbFdNTTy (cited on page 16).

[13] Mark Everingham, Luc Van Gool, Christopher Williams, John Winn, and Andrew Zisserman. "The Pascal Visual Object Classes (VOC) challenge". In: *International Journal of Computer Vision* 88 (June 2010), pages 303–338. DOI: 10.1007/s11263-009-0275-4 (cited on pages 19, 20).

[14] Kun Fu, Zhonghan Chang, Yue Zhang, Guangluan Xu, Keshu Zhang, and Xian Sun. "Rotation-aware and multi-scale convolutional neural network for object detection in remote sensing images". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 161 (2020), pages 294–308. ISSN: 0924-2716. DOI: https://doi.org/10.1016/j.isprsjprs.2020.01.025. URL: https://www.sciencedirect.com/science/article/pii/S0924271620300319 (cited on page 16).

[15] Feng Li, Ailing Zeng, Shilong Liu, Hao Zhang, Hongyang Li, Lei Zhang, and Lionel M. Ni. "Lite DETR : An Interleaved Multi-Scale Encoder for Efficient DETR". In: *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2023, pages 18558–18567. DOI: 10.1109/CVPR52729.2023.01780 (cited on page 16).

[16] Johann Li, Guangming Zhu, Cong Hua, Mingtao Feng, Basheer Bennamoun, Ping Li, Xiaoyuan Lu, Juan Song, Peiyi Shen, Xu Xu, Lin Mei, Liang Zhang, Syed Afaq Ali Shah, and Mohammed Bennamoun. "A Systematic Collection of Medical Image Datasets for Deep Learning". In: *ACM Comput. Surv.* 56.5 (Nov. 2023). ISSN: 0360-0300. DOI: 10.1145/3615862. URL: https://doi.org/10.1145/3615862 (cited on page 12).

[17] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. "Feature Pyramid Networks for Object Detection". In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pages 936–944. DOI: 10.1109/CVPR.2017.106 (cited on page 15).

[18] Feng Liu, Xiaosong Zhang, Zhiliang Peng, Zonghao Guo, Fang Wan, Xiangyang Ji, and Qixiang Ye. "Integrally Migrating Pre-trained Transformer Encoder-decoders for Visual Object Detection". In: *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2023, pages 6802–6811. DOI: 10.1109/ICCV51070.2023.00628.

[19] Qing Liu and Jian Li. "Orientation Robust Object Detection in Aerial Images Based on R-NMS". In: *Procedia Computer Science* 154 (Jan. 2019), pages 650–656. DOI: 10.1016/j.procs.2019.06.102 (cited on page 14).

[20] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. "SSD: Single Shot MultiBox Detector". In: *Lecture Notes in Computer Science*. Springer International Publishing, 2016, pages 21–37. ISBN: 9783319464480. DOI: 10.1007/978-3-319-46448-0_2. URL: http://dx.doi.org/10.1007/978-3-319-46448-0_2 (cited on page 15).

[21] Teli Ma, Mingyuan Mao, Honghui Zheng, Peng Gao, Xiaodi Wang, Shumin Han, Errui Ding, Baochang Zhang, and David S. Doermann. "Oriented Object Detection with Transformer". In: *ArXiv* abs/2106.03146 (2021). URL: https://api.semanticscholar.org/CorpusID:235358952 (cited on page 15).

[22] Muhammad Akhtar Munir, Muhammad Haris Khan, M. Saquib Sarfraz, and Mohsen Ali. "SSAL: synergizing between self-training and adversarial learning for domain adaptive object detection". In: *Proceedings of the 35th International Conference on Neural Information Processing Systems*. NIPS '21. Red Hook, NY, USA: Curran Associates Inc., 2024. ISBN: 9781713845393 (cited on page 18).

[23] Loris Nanni, Stefano Ghidoni, and Sheryl Brahnam. "Handcrafted vs Non-Handcrafted Features for computer vision classification". In: *Pattern Recognition* 71 (June 2017). DOI: 10.1016/j.patcog.2017.05.025 (cited on page 14).

[24] Son Nguyen, Theja Tulabandhula, and Duy Nguyen. "Dynamic Tiling: A Model-Agnostic, Adaptive, Scalable, and Inference-Data-Centric Approach for Efficient and Accurate Small Object Detection". In: *CoRR* abs/2309.11069 (2023). URL: https://api.semanticscholar.org/CorpusID:261975012 (cited on pages 10, 18).

[25] George Plastiras, Shahid Siddiqui, C. Kyrkou, and Theocharis Theocharides. "Efficient Embedded Deep Neural-Network-based Object Detection Via Joint Quantization and Tiling". In: *2020 2nd IEEE International Conference on Artificial Intelligence Circuits and Systems (AICAS)* (2020), pages 6–10. URL: https://api.semanticscholar.org/CorpusID:216104766 (cited on page 17).

[26] Zhong Qu, Le-yuan Gao, Sheng-ye Wang, Hao-nan Yin, and Tu-ming Yi. "An improved YOLOv5 method for large objects detection with multi-scale feature cross-layer fusion network". In: *Image Vision Comput.* 125.C (Sept. 2022). ISSN: 0262-8856. DOI: 10.1016/j.imavis.2022.104518. URL: https://doi.org/10.1016/j.imavis.2022.104518 (cited on page 15).

[27] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks". In: *IEEE Transactions on Pattern Analysis Machine Intelligence* 39.06 (June 2017), pages 1137–1149. ISSN: 1939-3539. DOI: 10.1109/TPAMI.2016.2577031 (cited on pages 14, 35).

[28] F. Özge Ünel, Burak O. Özkalayci, and Cevahir Çiğla. "The Power of Tiling for Small Object Detection". In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. 2019, pages 582–591. DOI: 10.1109/CVPRW.2019.00084 (cited on pages 10, 11, 17).

[29] Gui-Song Xia, Xiang Bai, Jian Ding, Zhen Zhu, Serge Belongie, Jiebo Luo, Mihai Datcu, Marcello Pelillo, and Liangpei Zhang. "DOTA: A Large-Scale Dataset for Object Detection in Aerial Images". In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2018 (cited on pages 14, 15, 18, 38).

[30] Xingxing Xie, Gong Cheng, Jiabao Wang, Xiwen Yao, and Junwei Han. "Oriented R-CNN for Object Detection". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. Oct. 2021, pages 3520–3529 (cited on page 14).

[31] Chang Xu, Jian Ding, Jinwang Wang, Wen Yang, Huai Yu, Lei Yu, and Gui-Song Xia. "Dynamic Coarse-To-Fine Learning for Oriented Tiny Object Detection". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2023, pages 7318–7328 (cited on page 18).

[32] Sheng Yang, Ziqiang Pei, Feng Zhou, and Guoyou Wang. "Rotated Faster R-CNN for Oriented Object Detection in Aerial Images". In: *Proceedings of the 2020 3rd International Conference on Robot Systems and Applications*. ICRSA '20. Chengdu, China: Association for Computing Machinery, 2020,

pages 35–39. ISBN: 9781450387644. DOI: 10.1145/3402597.3402605. URL: https://doi.org/10.1145/3402597.3402605 (cited on pages 24–26, 35).

[33] Xue Yang and Junchi Yan. "On the Arbitrary-Oriented Object Detection: Classification Based Approaches Revisited". In: *International Journal of Computer Vision* 130 (May 2022), pages 1–26. DOI: 10.1007/s11263-022-01593-w (cited on page 15).

[34] Hong-Xing Yu, Yunjie Tian, Qixiang Ye, and Yunfan Liu. "Spatial Transform Decoupling for Oriented Object Detection". In: *AAAI Conference on Artificial Intelligence*. 2023. URL: https://api.semanticscholar.org/CorpusID:261048888 (cited on pages 13, 15, 16, 24–26, 35, 36, 41, 53).

[35] Chi Zhang, Lijuan Liu, Xiaoxue Zang, Frederick Liu, Hao Zhang, Xi-gang Song, and Jin-Duan Chen. "DETR++: Taming Your Multi-Scale Detection Transformer". In: *ArXiv* abs/2206.02977 (2022). URL: https://api.semanticscholar.org/CorpusID:249431633 (cited on page 16).

[36] Gongjie Zhang, Zhipeng Luo, Zichen Tian, Jingyi Zhang, Xiaoqin Zhang, and Shijian Lu. "Towards Efficient Use of Multi-Scale Features in Transformer-Based Object Detectors". In: *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2023, pages 6206–6216. DOI: 10.1109/CVPR52729.2023.00601 (cited on page 16).

[37] Xiaosong Zhang, Yunjie Tian, Lingxi Xie, Wei Huang, Qi Dai, Qixiang Ye, and Qi Tian. "HiViT: A Simpler and More Efficient Design of Hierarchical Vision Transformer". In: *The Eleventh International Conference on Learning Representations*. 2023. URL: https://openreview.net/forum?id=3F6I-0-57SC (cited on pages 24, 26, 36).

[38] Yunzuo Zhang, Cunyu Wu, Tian Zhang, Yameng Liu, and Yuxin Zheng. "Self-Attention Guidance and Multiscale Feature Fusion-Based UAV Image Object Detection". In: *IEEE Geoscience and Remote Sensing Letters* 20 (2023), pages 1–5. DOI: 10.1109/LGRS.2023.3265995 (cited on page 16).

[39] Yue Zhou, Xue Yang, Gefan Zhang, Jiabao Wang, Yanyi Liu, Liping Hou, Xue Jiang, Xingzhao Liu, Junchi Yan, Chengqi Lyu, Wenwei Zhang, and Kai Chen. "MMRotate: A Rotated Object Detection Benchmark using PyTorch". In: *Proceedings of the 30th ACM International Conference on Multimedia*. 2022 (cited on page 41).