DOCTORAL PROGRAMME

Information Management

OBJECT DETECTION IN MEDICAL IMAGING

Carina Isabel Andrade Albuquerque

A thesis submitted in partial fulfillment of the requirements for the degree of Doctor in Information Management

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

NOVA Information Management School

Instituto Superior de Estatística e Gestão de Informação

Universidade Nova de Lisboa

OBJECT DETECTION IN MEDICAL IMAGING

by

Carina Isabel Andrade Albuquerque

Doctoral Thesis presented as partial requirement for obtaining the PhD in Information Management

Supervisor / Co Supervisor: Professor Roberto Henriques, PhD

Co Supervisor*:* Professor Mauro Castelli, PhD

February 2023

STATEMENT OF INTEGRITY

I hereby declare having conducted this academic work with integrity. I confirm that I have not used plagiarism or any form of undue use of information or falsification of results along the process leading to its elaboration. I further declare that I have fully acknowledge the Rules of Conduct and Code of Honor from the NOVA Information Management School.

Lisboa, February 7, 2023

ACKNOWLEDGEMENTS

This dissertation could not have been accomplished without all the support and guidance of my advisors, Professor Roberto Henriques and Professor Mauro Castelli. Their contributions exceeded mere words of advice and extended to a belief in my work, even when I could not see its potential myself. Their encouragement and kind words during the most challenging moments emphasized the significance of perseverance and tenacity and for that, I could not have hoped for better mentors.

I am grateful to Fundação Champalimaud for providing me with the needed resources to undertake my initial study. Without it, I would not have embarked on this academic journey.

To my parents, who offered me multiple opportunities to redirect my life journey without any judgment and provided me with their crucial support. To my beloved father, my source of stubbornness and the one that shows me that nothing is impossible. To my beloved mother, my source of wisdom and understanding, and for loving me unconditionally.

To my brother and sister-in-law, who stood by me during the most trying moments of my journey. A special thank you to my brother for reminding me that every obstacle can be overcome, and for making me see further. I would be proud and aspire to be as supportive to little Henrique, as my brother has been to me.

To Gonçalo, for the belief in me and for being a constant source of inspiration and wisdom. Though it came later in my journey, it arrived at the perfect time. Thank you for your knowledge and patience, for bringing me peace and happiness at the precise moment it needed to happen. I am thankful for having you in my life. Your unconditional faith in me has allowed me to believe again.

To Vera and Fábia, my unofficial family. No matter how far apart we may be, our bond remains unbreakable, and I never feel estranged from you. I miss you deeply. Thank you for reminding me many times of my true self and for supporting me on these crazy paths with no judgments.

I am deeply grateful and blessed for the people mentioned, as well as many other friends I am lucky to have, and people I know who crossed my life and walked with me, leading me to the path I am on today. I thank each and every one of you for your support, belief in me, and insightful words of wisdom.

This dissertation is the output of several convolutions merging data and information gathered over the course of 36 years of personal experiences, innumerous mistakes, fortunate events, and enlightening encounters with remarkable individuals. It has been an extraordinary journey, often challenging, yet incredibly inspiring.

Lastly, I am thankful for life and every chance it has afforded me. I could not have asked for a more fulfilling path.

ABSTRACT

Artificial Intelligence, assisted by deep learning, has emerged in various fields of our society. These systems allow the automation and the improvement of several tasks, even surpassing, in some cases, human capability. Object detection methods are used nowadays in several areas, including medical imaging analysis. However, these methods are susceptible to errors, and there is a lack of a universally accepted method that can be applied across all types of applications with the needed precision in the medical field. Additionally, the application of object detectors in medical imaging analysis has yet to be thoroughly analyzed to achieve a richer understanding of the state of the art.

To tackle these shortcomings, we present three studies with distinct goals. First, a quantitative and qualitative analysis of academic research was conducted to gather a perception of which object detectors are employed, the modality of medical imaging used, and the particular body parts under investigation. Secondly, we propose an optimized version of a widely used algorithm to overcome limitations commonly addressed in medical imaging by fine-tuning several hyperparameters. Thirdly, we develop a novel stacking approach to augment the precision of detections on medical imaging analysis.

The findings show that despite the late arrival of object detection in medical imaging analysis, the number of publications has increased in recent years, demonstrating the significant potential for growth. Additionally, we establish that it is possible to address some constraints on the data through an exhaustive optimization of the algorithm. Finally, our last study highlights that there is still room for improvement in these advanced techniques, using, as an example, stacking approaches.

The contributions of this dissertation are several, as it puts forward a deeper overview of the state-of-the-art applications of object detection algorithms in the medical field and presents strategies for addressing typical constraints in this area.

KEYWORDS

Object Detection; Medical Imaging; Deep Learning; Computer Vision; Good Health

Sustainable Development Goals (SGD):

RESUMO

A Inteligência Artificial, auxiliada pelo deep learning, tem emergido em diversas áreas da nossa sociedade. Estes sistemas permitem a automatização e a melhoria de diversas tarefas, superando mesmo, em alguns casos, a capacidade humana. Os métodos de detecção de objetos são utilizados atualmente em diversas áreas, inclusive na análise de imagens médicas. No entanto, esses métodos são suscetíveis a erros e falta um método universalmente aceite que possa ser aplicado em todos os tipos de aplicações com a precisão necessária na área médica. Além disso, a aplicação de detectores de objetos na análise de imagens médicas ainda precisa ser analisada minuciosamente para alcançar uma compreensão mais rica do estado da arte.

Para enfrentar essas limitações, apresentamos três estudos com objetivos distintos. Inicialmente, uma análise quantitativa e qualitativa da pesquisa acadêmica foi realizada para obter uma percepção de quais detectores de objetos são empregues, a modalidade de imagem médica usada e as partes específicas do corpo sob investigação. Num segundo estudo, propomos uma versão otimizada de um algoritmo amplamente utilizado para superar limitações comumente abordadas em imagens médicas por meio do ajuste fino de vários hiperparâmetros. Em terceiro lugar, desenvolvemos uma nova abordagem de stacking para aumentar a precisão das detecções na análise de imagens médicas.

Os resultados demostram que, apesar da chegada tardia da detecção de objetos na análise de imagens médicas, o número de publicações aumentou nos últimos anos, evidenciando o significativo potencial de crescimento. Adicionalmente, estabelecemos que é possível resolver algumas restrições nos dados por meio de uma otimização exaustiva do algoritmo. Finalmente, o nosso último estudo destaca que ainda há espaço para melhorias nessas técnicas avançadas, usando, como exemplo, abordagens de stacking.

As contribuições desta dissertação são várias, apresentando uma visão geral em maior detalhe das aplicações de ponta dos algoritmos de detecção de objetos na área médica e apresenta estratégias para lidar com restrições típicas nesta área.

INDEX

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

AI	Artificial Intelligence
AP	Average Precision
API	Application Programming Interface
CSPNet	Cross Stage Partial Network
CNN	Convolutional Neural Network
CRC	Colorectal Cancer
DL	Deep Learning
FCOS	Fully Convolutional One-Stage Object Detection
GB	Gradient Boosting
IGH	Institute of Gastroenterology and Hepatology
LR	Logistic Regression
mAP	Mean average precision
NMS	Non-Maximum Suppression
NMW	Non-Maximum Weighted
R-CNN	Region Based Convolutional Neural Networks
ResNet	Residual Neural Network
RFCN	Region Based fully convolutional Neural Networks
RF	Random Forest
ROI	Region of Interest
RPN	Region Proposal Network
SSD	Single shot detector
SVM	Support Vector Machine
WBF	Weighted Boxes Fusion
YOLO	You Only Look Once

"If I have seen further, it is by standing on the shoulders of giants."

(Sir Isaac Newton)

“The ultimate answer to life, the universe and everything is … 42!”

(Douglas Adams, The Hitchhiker's Guide to the Galaxy)

1. INTRODUCTION

1.1. COMPUTER VISION

Computer vision is a specialized field within Artificial Intelligence (AI) that aims to replicate the complexity of human vision by developing techniques that allow computers to extract meaningful information from digital images or videos. Despite being a research field for several decades (Prince, 2012), the limited capabilities of computers prevented significant improvements in the field. However, with recent advancements in deep learning (DL), the field has undergone significant advancements, even surpassing human performance in certain tasks such as classification and object detection. While the origins of deep learning can be traced back to 1943 with the proposal of neural networks by Walter Pitts and Warren McCulloch (McCulloch & Pitts, 1943), it was only after the 2010s, with the increase in the speed of GPUs (Pandey et al., 2022; Voulodimos et al., 2018), that deep learning began to demonstrate its potential in a wide range of fields.

In computer vision, the system learns by identifying patterns from a training dataset, much like humans do through repeated exposure. However, to a computer, an image is simply a collection of pixels, with each color comprising three channels: red, green, and blue. By converting these values into an array, computer vision tasks can recognize patterns and associate them with specific shapes or objects.

The field of computer vision encompasses a variety of tasks, such as classification, object detection, and segmentation (Khan & Al-Habsi, 2020). All those techniques involve the recognition and understanding of objects; however, they differ in their specific goals and the information they provide.

In image classification, an image is taken as input and the goal is to assign a label to the image. In object detection, the system can distinguish between objects in an image or video, by locating and classifying each object using rectangular bounding boxes. For instance segmentation, the classification is done at the pixel level, providing a detailed understanding of the image by defining and classifying each pixel.

1.2. OBJECT DETECTION

Object detection is a visual recognition problem in computer vision that involves identifying objects with precise localization and labeling in an image. The advent of deep learning in recent years has led to a significant expansion of this technique, resulting in remarkable breakthroughs in a wide range of real-world applications such as face detection (W. Chen et al., 2021; H. Jiang & Learned-Miller, 2017), autonomous driving (Fujiyoshi et al., 2019; Yurtsever et al., 2020), security monitoring (Ahmed et al., 2021; Ashraf et al., 2022), and text detection (Z. Zhang et al., 2016; Zhong et al., 2019), among others.

The history of object detection applications dates back over twenty years (Zou et al., 2019). This field was triggered by the emergence of Viola-Jones Detectors (Viola & Jones, 2001), which aimed to address the problem of face detection. Since then, numerous methods and techniques have emerged, and the history of object detection can be broadly divided into two periods: the traditional object detection period before 2014 and the deep-learning-based detection period after 2014 (Zou et al., 2019). The traditional detectors encompass not only the Viola-Jones Detectors (Viola & Jones, 2001) but also the Histogram of Oriented Gradients (HOG) (Dalal & Triggs, 2005) and the Deformable Part-based Model

(DPM) (Felzenszwalb et al., 2008). The first period reached its peak in the 2010s, but progress slowed down between 2010 and 2012, with only small improvements achieved by ensemble techniques, as noted by R. Girshick (R. B. Girshick, 2012).

The new era of object detection was ushered in with the proposal of Region-based Convolutional Neural Networks (R-CNNs) (R. Girshick et al., 2014). The authors presented two-stage detectors based on convolutional neural networks (CNNs). This marked the beginning of a new era of object detection that has been driven by advancements in deep learning.

The architecture of two-stage object detectors is composed of two distinct phases, which separate the task of object location from that of classification. Within this category of detectors, we find such examples as the R-CNN (R. Girshick et al., 2014), Fast Region-based Convolutional Network (Fast R-CNN) (R. Girshick, 2015a), Faster Region-based Convolutional Network (Faster R-CNN) (Ren et al., 2015), Region-based Fully Convolutional Network (RFCN) (Dai et al., 2016a), and Mask Region-based Convolutional Network (Mask R-CNN) (K. He et al., 2017). The first stage of these detectors is primarily focused on generating region proposals, while the second stage is responsible for classifying those regions.

One-stage detectors emerged with Single Shot MultiBox Detector (SSD) (W. Liu et al., 2016a) and You Only Look Once (YOLO) (Redmon et al., 2016). These detectors possess a simpler architecture than two-stage detectors and simultaneously locates and classifies each object without the need for a region proposal process. In posterior years, other one-stage approaches appeared, such as RetinaNet (T.-Y. Lin et al., 2017), DetNet (Z. Li et al., 2018), CenterNet (K. Duan et al., 2019), and EfficientDet (Tan et al., 2020), among others.

Figure 1.1 showcases a roadmap of object detection, highlighting traditional methods, one-stage and two-stage deep learning-based approaches, and their corresponding years of advent.

These different architectures have undergone extensive examination, and their performance has been compared in several studies (Aziz et al., 2020; Yang Liu et al., 2021; Srivastava et al., 2021; Tong et al., 2020; Xiao et al., 2020; Zaidi et al., 2022; Zhao et al., 2019; Zou et al., 2019) on several studies. In general, there is a compromise between speed and accuracy. While two-stage detectors tend to yield higher detection accuracy, they also exhibit a slower training phase. Limitations and future directions in the field include the need for optimized object detectors for small objects (Yang Liu et al., 2021; Tong et al., 2020; Zhao et al., 2019; Zou et al., 2019), improved localization accuracy on objects with partial occlusions (Zhao et al., 2019), lightweight object detection versions adapted for mobile devices (Zou et al., 2019), weakly supervised detection methods that do not depend on a large amount of well-annotated images (Tong et al., 2020; Zhao et al., 2019; Zou et al., 2019), object detection methods for high-resolution images and videos (Xiao et al., 2020), network optimization for balancing speed, memory, and accuracy (Zhao et al., 2019), and 3D object detection (Aziz et al., 2020; Zaidi et al., 2022), among others.

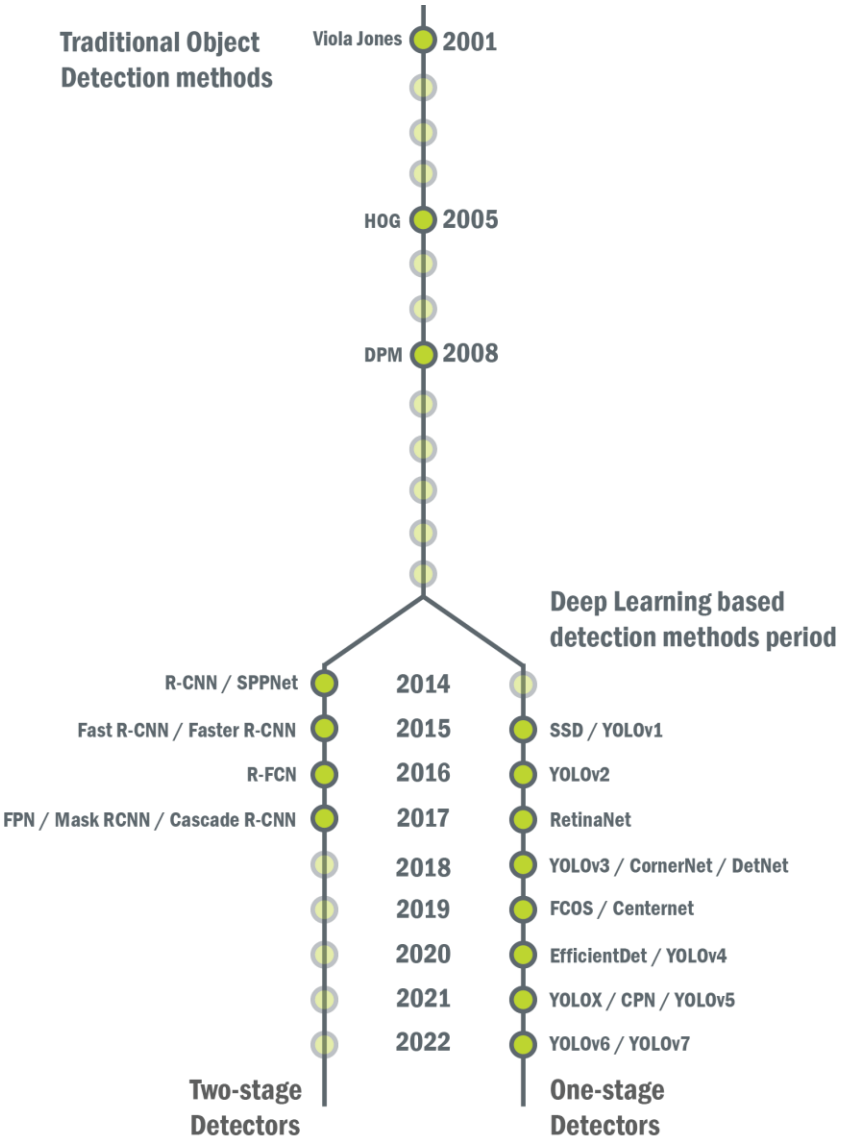

Figure 1.1 – Evolution of object detection methods

1.2.1. Metrics in Object Detection

To evaluate the performance of an object detector, we use three of the four available measures on the confusion matrix: (1) the true positive (TP) that is defined as a correct identification of a ground-truth bounding box; (2) the false positive (FP) that identifies all the incorrect detections where no object exists; and the (3) false negative (FN), where there is no detection of the model when in reality we have. The concept of true negative (TN) is not used in object detection, as there are infinite bounding boxes that should not be detected within any given image. As so, traditional metrics such as accuracy, specificity, ROC Curve, and false positive rate are not applicable.

The output of object detectors typically is comprised of a list of bounding boxes, confidence levels and classes, but the standard format can vary depending on the specific algorithm being used. A bounding box is usually represented by its top-left and bottom-right coordinates, denoted as $(x_{ini}, y_{ini}, x_{end}, y_{end})$, with some exceptions, such as in the YOLO algorithm (Padilla et al., 2021).

Although there is a lack of consensus among various object detection works regarding the metrics that should be used to evaluate the performance of an object detector, the average precision (AP), based on the concept of Intersection Over Union (IoU), and the mean Average Precision (mAP) stand as the most widely used metrics (Padilla et al., 2021).

The AP, introduced in the VOC2007 dataset, is defined as the average detection precision under different recalls for each individual class.

The concept of IoU is applied to distinguish what is a correct identification from an incorrect one. Based on the Jaccard index, the IoU measures the overlapping area between the ground-truth bounding box BB_{GT} and the predicted bounding box BB_P , divided by the area of the union of both (Padilla et al., 2021), as seen in Equation 1.1:

$$IoU = \frac{BB_{GT} \cap BB_P}{BB_{GT} \cup BB_P} \tag{1.1}$$

IoU ranges from 0 to 1, where 0 means no overlap and 1 stands for a perfect overlap. A threshold is defined previously to determine whether a prediction is a TP, FP, or FN, as shown in Figure 1.2.

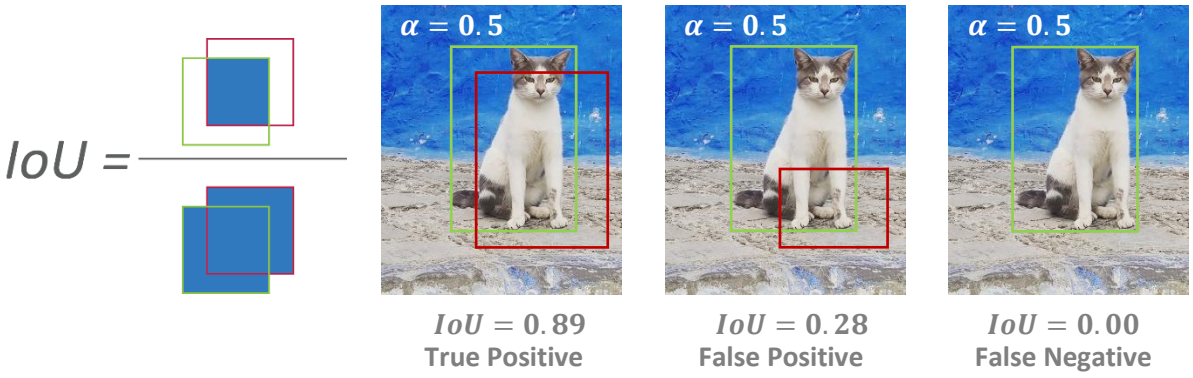

Figure 1.2 – The concept of IoU, and the representation of a TP, FN and FP, using a threshold of 0.5.

Note. The green bounding box corresponds to the ground truth, while the red bounding box is associated with the prediction. Predictions with an IoU value higher than the predefined threshold are considered TP, while predictions with a lower IoU are considered FP. If the IoU value is equal to 0, it implies a case of FN.

The AP is defined as the area under the precision-recall curve on different confidence thresholds, where precision is (see Equation 1.2):

$$Precision = \frac{TP}{TP + FP} \tag{1.2}$$

And recall is given by (see Equation 1.3):

$$Recall = \frac{TP}{TP + FN} \tag{1.3}$$

The mAP is determined by taking the average of AP scores obtained among all possible classes (Padilla et al., 2021), as shown in Equation 1.4,i.e.,

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i \tag{1.4}$$

where N stands for the total number of classes, and AP_i being the AP in the ith class.

1.3. MEDICAL IMAGING

In the past decades, medical imaging became an essential element of healthcare. Imaging is widely used not only to detect and diagnose diseases (Habuza et al., 2021) but also as a tool for therapy planning and assessment (Ritter et al., 2011).

Medical imaging refers to a range of techniques used in medicine where images of the human body's structure and intrinsic mechanisms, such as organs, are used to aid the diagnosis, monitoring, and treatment of various medical conditions. Medical images encompass different techniques, including computed tomography (CT), ultrasounds, mammograms, X-rays, Magnetic Resonance Imaging (MRI), endoscopy, and positron emissions tomography (PET), among others (Chowdhary & Acharjya, 2020; Kaur et al., 2021; Yang & Yu, 2021).

Medical imaging modalities are defined based on how the images are acquired. X-rays and CT scans, for instance, utilize ionizing radiation. X-rays are frequently employed to detect fractures, and dislocations, but also lung and heart pathologies. CT scans provide detailed images of bones, soft tissues, and blood vessels in different parts of the body (Kaur et al., 2021; Puttagunta & Ravi, 2021).

Magnetic Resonance Imaging (MRI), which uses magnetic waves, is widely used to diagnose aneurysms, tumors, and strokes (Kaur et al., 2021), playing an essential role in the diagnosis of brain, heart, and soft tissues conditions (J. Li et al., 2021).

Ultrasound techniques, which use high-frequency sound waves to produce images of organs and structures of the body, are frequently used to monitor pregnancy and imaging of the heart in real-time (J. Li et al., 2021).

Pathology imaging can include techniques where a microscope is employed on stained tissue slides to display cell-level features. It is widely used for the diagnosis of cellular diseases, including cancers and tumors (J. Li et al., 2021).

Medical image interpretation was traditionally carried out by medical specialists such as radiologists and physicians (H. Yu et al., 2021). However, this can prove to be a challenging task due to the substantial differences in existing pathologies and the likelihood of expert fatigue (Shen et al., 2017), leading to the development of computer-assisted tools to aid with the process (Tsuneki, 2022; Yang & Yu, 2021).

1.4. Object Detection in Medical Imaging

Deep learning can be employed in medical imaging to carry out various functions such as classification, localization, detection, segmentation, and image analysis registration (Kaur et al., 2021).

Object detection, in particular, has received significant attention in the field, with various studies exploring the capabilities of distinct object detection algorithms on diverse medical imaging modalities across several anatomical application areas. Despite the various proposals of distinct object detectors by researchers, there is a lack of a universally applicable method that can be used across all types of applications with the needed precision in the medical field.

The arrival of deep learning-based detection methods was marked by the emergence of Faster R-CNN as a leading system. At the same time, techniques such as YOLO and SSD also arose, but Faster R-CNN was quickly recognized for its higher accuracy compared to one-stage methods (J. Huang et al., 2017), making it suitable for the demanding requirements of the medical field. Over time, one-stage detectors have evolved towards more accurate techniques. The ongoing improvement of algorithms such as YOLO along several versions, prioritizing both speed and performance, has led to a significant shift from two-stage to one-stage object detectors, with YOLO becoming the most widely used algorithm, and producing comparable or even superior results to those of two-stage detectors.

However, as suggested by the No Free Lunch Theorem, it is asserted that all optimization algorithms perform equally well when evaluated across all feasible problems. This implies that a generic, universal optimization strategy is not viable, and the only way for one strategy to outperform another is through specialization to the unique structure of the specific problem at hand (Ho & Pepyne, 2002). Furthermore, the nature of the data and its preprocessing are just as important, if not more so, as the algorithm itself in determining the outcome (Gómez & Rojas, 2016).

1.4.1. Real-time object detection on medical imaging

Endoscopy is widely used for colon polyp detection, and many studies in this area favored one-stage object detectors for their faster inference speeds, a needed condition during a colonoscopy. For an object detection algorithm to detect an object in real time, it must provide at least 30 FPS (I Pacal et al., 2022). Some studies have utilized SSD-based versions with speeds of 50 FPS (Ozawa et al., 2020) and 62.5 FPS (Souaidi & Ansari, 2022), while a study using YOLO reached 100 FPS (I Pacal et al., 2022). RefineNet, a two-stage detector used for the same purpose, had a slower speed of only 32 FPS (K. Li

et al., 2021). However, depending on the changes made to the network employed, one-stage detectors can become slower than optimized versions of two-stage detectors. This is seen when comparing two cases of endoscopies applied in the stomach, where the demand of high inference time is also required, one of those using Faster R-CNN and the other a YOLOv3, both of them with adaptations from the original algorithm. In the former, a detection speed of 23 FPS was achieved (Xia et al., 2021), while YOLO achieved only 6 FPS (L Wu et al., 2022). However, as mentioned in the first of those studies, a clinician was only able to have an inference speed of around 3 FPS.

1.4.2. The need of large data sets

As a rule, larger data sets tend to yield superior outcomes. The feeding of an algorithm with a greater number of data points generally enables a higher capability for generalization, and enhances the model's ability to make accurate predictions (Cho et al., 2015). However, it is imperative to note that the expansion of data size must be accompanied by maintenance in data quality, which includes consistency and completeness, among other factors.

In a study aimed to evaluate breast cancer detection and classification (Y Liu et al., 2022), both Faster R-CNN and Mask R-CNN-based algorithms were applied to two distinct data sets composed of mammograms. The first data set, the Digital Database of Screening Mammography (DDSM) (PUB et al., 1996), contained 2620 mammograms, while the second was an in-house data set with 40,000 images. The results showed that Faster R-CNN achieved an mAP of 75.3% on the DDSM data set and 82.3% on the largest data set. The same pattern was identified with the proposed algorithm, achieving an mAP of 82% on the smaller data set and 87.6% on the larger one. The authors additionally noted that other public data sets, such as INBreast (Moreira et al., 2012) and MIAS (Suckling, 1994) were not considered due to their insufficient size for training a detection model.

Other studies suggest the implementation of data augmentation to artificially increase the size of the data set available on different topics, such as malaria detection in blood smear microscopic imaging (Abdurahman et al., 2021), diabetic retinopathy detection on eye fundus imaging (Ju et al., 2021), and melanoma diagnosis on photography (Jojoa Acosta et al., 2021).

1.4.3. Leading with overcrowded situations

In medical imaging tasks, challenges arise from overlapping objects, such as cell counting in zebrafish xenografts (C Albuquerque et al., 2021) and cell tracking in bacterial colonies, where a lack of accuracy in models is clear in overcrowded regions (Balomenos et al., 2015). A study of white blood cell detection (Kutlu et al., 2020) enhances the success of deep learning methods, against the challenges of overlapping or partial visibility of the object of interest, from the classical image processing methods. However, the detection of multi-scale cells in bone marrow aspirate smear (Su et al., 2021) still faces difficulties due to small overlapping cells, despite advancements in object detection models. The authors highlight that current detectors are not capable of accurately counting cells in these conditions.

1.4.4. Creating models with high generalization capability

Research has been conducted to evaluate the ability of object detection frameworks in detecting various organs and objects of interest in medical imaging. A Faster R-CNN was proposed for organ localization using CT scans of eleven body organs and twelve head organs with distinct anatomical structures (X. Xu et al., 2019). The results showed a global mAP of 57.30% for head organs, with AP scores ranging from 44.99% for the left joint to 79% for the oral cavity. The proposed object detector achieved an mAP of 73.01% for body organs, with AP scores ranging from 58.23% for the bladder to 86.88% for the right lung.

Lesion detection is the focus of two studies aimed at constructing a universal detector. The DeepLesion dataset was introduced in 2018 (K. Yan et al., 2018), offering over 32,000 lesions located in various parts of the body across 32,120 CT scans. This comprehensive dataset encompasses a wide range of lesions, including lung nodules, liver tumors, and enlarged lymph nodes, among others. The authors aim to develop a Faster R-CNN-based detector that can be utilized in various medical imaging applications. On average, the model identified 5 false positives in each image. Three years later, the authors (K. Yan et al., 2021) acknowledge the limitations of the dataset, such as heterogeneous label scopes, and address these issues. After implementing a Mask R-CNN on the newer version of the dataset, now named LENS, the results of this evaluation show an average sensitivity of 47.6%. Despite these results, the authors also identified certain limitations of the universal lesion detector proposal, such as its tendency to identify smaller parts on a single lesion or irregular lesions with a large bounding box.

1.4.5. Ensemble methods on medical imaging

Ensemble methods are a group of techniques that leverage the combination of multiple models to create a final model with improved performance. These methods have been widely adopted in both traditional Machine Learning and deep learning fields, with notable applications in medical imaging. In classification tasks, researchers have used ensembles of fine-tuned convolutional neural networks to achieve promising results on the ImageCLEF (Kalpathy-Cramer et al., 2015) medical image dataset, with 30 categories (Kumar et al., 2016). In segmentation, the DivergentNets algorithm has been proposed, which combines multiple well-known segmentation models into one (Hossain et al., 2022).

Despite the widespread usage of ensemble methods in various fields of study, their application in the field of object detection remains relatively uncharted. In medical imaging, a recent study sought to explore the potential of ensemble techniques through the implementation of the concept of weighted box fusion (WBF) in fracture detection within wrist X-rays (Hardalaç et al., 2022). The study combined several popular object detection models, including Faster R-CNN, RetinaNet, and Dynamic R-CNN, and observed a marked performance improvement. Specifically, the ensemble model recorded an mAP of 86.39%, compared to an mAP of 75.4% for the best single model. Recently, a novel stacking-based artificial intelligence framework was developed for the detection of colon polyps with high precision. The proposed model achieves an mAP of 86%, representing a significant improvement of 34.9% compared to the best baseline model and 28.8% compared to the WBF ensemble technique (Carina Albuquerque et al., 2022).

1.5. MOTIVATION

Medical imaging is an indispensable tool widely used in the healthcare industry and medical field. Allied with the advent of deep learning, the analysis of medical imaging has entered a new era, where disease diagnosis and therapy assessment may become faster and less prone to error. Although the replacement of medical experts by artificial intelligence is neither feasible nor desirable, deep learning has enabled various advances in the medical area that aids in the diagnosis, monitoring, and treatment of several medical conditions. However, these techniques are not free of potential biases and errors.

We are aware that, in the medical field, the accuracy of results is of paramount significance, as it helps medical experts to determine the best course of action for each patient. One of the main obstacles to the wide adoption of deep learning techniques in clinical practice is their limited generalization capability caused by the variability inherent in medical data.

Deep learning-based computer vision comprises a range of tasks, such as classification, segmentation, and object detection, where each task has a distinct purpose. Classification is the simplest task and aims to categorize the object or image into one of the several predefined classes. Object detection goes one step further by not only determining the class an object belongs to but also providing the localization of it within an image, given by a bounding box around each object. Finally, instance segmentation provides a mask for each object and segment it out from the rest of the image.

While there is a significant increase in scientific research focusing on classification and segmentation tasks in medical imaging, the concept of object detection still deserves further exploration. Although segmentation can offer a more profound understanding of the boundaries of an object of interest, the process of creating a training data set for semantic segmentation requires a laborious and time-consuming review of the images, during which the boundaries of relevant objects must be carefully drawn. Additionally, within some tasks in the medical field, such detail is neither useful nor necessary. These tasks include, for instance, cell counting, polyp detection, or fracture recognition.

Regardless of the relevant advances in object detection in recent years, some applications are still associated with unacceptable errors in the medical field. To effectively integrate deep learning into clinical practice, it is crucial to improve those algorithms by adapting them to medical image analysis, where constraints such as limited data, the specificity of each problem, and the need for short inference time may be present.

The application of object detectors in the medical field has brought several advantages but has also been overwhelmed by certain limitations. A new challenge has emerged: given the massive amount of available data produced daily, it is imperative to understand and investigate how we can adapt techniques designed for big data to fields where data is typically scarce and precision is of paramount importance.

For these reasons and given these considerations, there was a clear opportunity to investigate the significance of deep learning object detection methods in medical images nowadays, as also to develop various techniques that may potentially improve and enhance the performance of object detection algorithms within the context of medical image analysis.

1.6. RESEARCH QUESTIONS

As previously suggested, the primary focus of this work is to assess the current state of the art of object detection in the medical field and to apply object detection algorithms in medical imaging analysis. Additionally, further efforts are made to adapt and improve those frameworks to typical challenges in medical imaging, such as constraints in the data set and the need for high-performance models.

In particular, the main research questions are as follows:

- To what extent is the application of deep learning object detection methods being explored in the medical imaging analysis field?
- What are the most common object detection algorithms employed, the modality of medical imaging used, and the specific body parts under study?
- How can object detection algorithms be adapted to handle overlapping objects and small objects in overcrowded scenes in a limited data set?
- How can the precision of detection, a crucial requirement in various medical imaging analysis tasks, be improved?

1.7. DISSERTATION STRUCTURE

This dissertation is structured into five chapters. The first chapter briefly introduces computer vision, object detection algorithms, medical imaging, and the application of object detection in medical imaging. It also presents the dissertation motivation, research questions, and the scientific production involved. The second chapter presents a quantitative and qualitative analysis of the application of object detection algorithms in medical imaging analysis. In this section, we highlight and interpret several quantitative indicators and analyze the top cited publications in a qualitative approach, focusing on the object detection algorithms applied, the modality of medical imaging used, and the exact body parts under investigation, including comparisons in terms of performance. The third chapter presents a fine-tuned and optimized version of Faster R-CNN developed to handle a data set with constraints such as its small size and the presence of small cells, among others. The strategies applied to deal with those limitations are also defined. Chapter 4 presents the development of a novel stacking-based AI framework designed to detect colon polyps in endoscopic images with superior precision compared to standalone models and other commonly used ensemble techniques. Finally, chapter 5 summarizes the most relevant findings of the research, considers the contributions of this work, points out limitations, and provides directions for future research.

1.8. SCIENTIFIC PRODUCTION

1.8.1. Published journal articles

- Albuquerque, C., Vanneschi, L., Henriques, R., Castelli, M., Póvoa, V., Fior, R., & Papanikolaou, N. (2021). Object detection for automatic cancer cell counting in zebrafish xenografts. *Plos one*, *16*(11), e0260609.

- Albuquerque, C., Henriques, R., & Castelli, M. (2022). A stacking-based artificial intelligence framework for an effective detection and localization of colon polyps. *Scientific Reports*, *12*(1), 17678.

1.8.2. Submitted journal articles

- Albuquerque, C., Henriques, R., & Castelli, M. (2022). Object detection in medical imaging: a quantitative and qualitative analysis. Manuscript submitted for publication in a first quartile journal.

2. OBJECT DETECTION IN MEDICAL IMAGING: A QUANTITATIVE AND QUALITATIVE ANALYSIS ON THE SCIENTIFIC RESEARCH

2.1. INTRODUCTION

Medical imaging is an essential tool and plays a crucial role in the diagnosis and treatment planning in the medical field. Various approaches, such as Magnetic resonance imaging (MRI), computed tomography (CT), digital mammography, X-ray, ultrasound, positron emission tomography (PET), and digital pathology, are used to generate images that aid in a variety of tasks such as pathology localization, the study of anatomical structure, treatment planning and even computer-integrated surgery, among others (Pham et al., 2000; Tsuneki, 2022; S. K. Zhou et al., 2021).

Medical image analysis and interpretation have traditionally been performed by humans for several years. However, with the rapid advancement of artificial intelligence (AI), the medical field is increasingly embracing computer-assisted tools as a resourceful approach to solve diagnostic problems with precision and efficiency. This allows for real-time prediction of various diseases and in-depth examination of different treatment options (Tsuneki, 2022; Yang & Yu, 2021), avoiding limitations such as inter-observer and intra-observer variability (De Boodt et al., 2013) and error resulting from large variations in pathologies and the possible exhaustion of human experts (Shen et al., 2017).

Bibliometric analysis allows for uncovering emerging trends and patterns in the scientific knowledge of a specific domain in a quantitative approach (Donthu et al., 2021), by combining disciplines such as information science, mathematics and statistics (De Bellis, 2009). The topic of deep learning application in medical image analysis has been significantly exploited in recent years, due to its current relevance. Many bibliometric analysis works have focused on various fields of research including the usage of artificial intelligence in healthcare (Yuqi Guo et al., 2020), medical image segmentation (B. Zhang et al., 2021), the application of deep learning in the medical field (Egger et al., 2022), medical data mining research (Y. Hu et al., 2020), and machine learning-based disease diagnosis (Ahsan et al., 2022). To the best of our knowledge, there has been bibliometric and qualitative analysis research on the application of object detection algorithms in the medical field published.

In this work, we propose to conduct a comprehensive quantitative and qualitative analysis on the literature related to the application of deep learning object detectors in medical imaging, and to reveal potential trends in this field. Our contributions include:

- A through bibliometric analysis on the application of deep learning object detectors, conducted using two of the most widely used databases: Scopus and the Web of Science (WoS).
- The reporting of several relevant quantitative indicators, such as annual publication analysis, research area analysis, journal publication analysis, publications by country and author, and keyword analysis.

- In the qualitative analysis, the top publications were filtered based on annual citation rates and analyzed in greater depth. Additionally, we propose to evaluate the presence of various object detectors available in the research conducted, and to identify the imaging modalities and the anatomical application areas where object detectors are used with more incidence.

The structure of this work is as follows: in the first section, a brief introduction, background to the problem and the purpose of this work is provided. In the following section, the research methodology, data collection procedures, bibliometric tools and screening methods are outlined. In the third section, a quantitative and descriptive view, supported by a bibliometric analysis of the filtered information, is presented. In the fourth section, a qualitative analysis of the most cited papers is conducted, and an exploratory analysis is performed on the incidence of different object detectors applied and the medical fields where object detection is more prevalent. In the final section, the conclusions of this study and the overall findings are presented.

2.2. MATERIALS AND METHODS

With the objective of constructing a bibliometric analysis that is complemented by a qualitative analysis, two well-known academic databases were utilized. The Web of Science (WoS) Core Collection provides researchers with a substantial amount of pertinent publications dating back to 1900, including six Citation Indexes such as the Science Citation Index Expanded (SCIE), in addition to journal citation reports (JCR) and essential science indicators (ESI). The second database is Scopus, which encompasses peer-reviewed journals in the fields of life sciences, physical sciences, social sciences, and health sciences.

The framework for gathering and filtering data was based on Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) guidelines. The query search, conducted on January 20th, 2023, was centered around three main topics that align with our purpose, excluding the concept of segmentation on keywords or title, and limiting the results to articles written in English, as depicted in Table 2.1. Those results comprise articles up until the end of 2022.

The number of records retrieved from the WoS was 673 and 760 from Scopus. Following the PRISMA methodology, as illustrated in Figure 2.1, duplicates were identified, and exclusion criteria were applied.

Table 2.1 – Article selection on Scopus and WoS, based on strings and queries

Database	N	Keywords	Strings	Field	Results
WOS	#1	Deep learning and synonyms	(“deep learning”) OR (“convolutional neural network*”) OR (“convolution”) OR (“deep neural network*”)	Topic (TS)	299,122
	#2	Object Detection and synonyms	(“object detection”) OR (“object recognition”) OR (“lesion detection”)	Topic (TS)	65,125
	#3	Medical Imaging or synonyms	(“medical imaging”) OR (“diagnostic imaging”) OR (“diagnosis”) OR (“computer aided diagnosis”) OR (“disease*) OR (“medical computing”) OR (“medical”)	Topic (TS)	7,608,363
	#4	Segmentation	(“segment*”)	Key (AK) OR Title (TI)	228,918
	#1 AND #2 AND #3				1,321
	(#1 AND #2 AND #3) AND NOT #4				1,077
	(#1 AND #2 AND #3) AND NOT #4, English, Document type as “Article”				673
SCOPUS	#1	Deep learning and synonyms	(“deep learning”) OR (“convolutional neural network*”) OR (“convolution”) OR (“deep neural network*”)	TITLE-ABS-KEY	448,227
	#2	Object Detection and synonyms	(“object detection”) OR (“object recognition”) OR (“lesion detection”)	TITLE-ABS-KEY	116,390
	#3	Medical Imaging or synonyms	(“medical imaging”) OR (“diagnostic imaging”) OR (“diagnosis”) OR (“computer aided diagnosis”) OR (“disease*) OR (“medical computing”) OR (“medical”)	TITLE-ABS-KEY	15,611,842
	#4	Segmentation	(“segment*”)	Key OR Title	479,319
	#1 AND #2 AND #3				2,502
	(#1 AND #2 AND #3) AND NOT #4				1,863
	(#1 AND #2 AND #3) AND NOT #4, English, Document type as “Article”				760

Note. The search was conducted using both WoS and Scopus data bases. Records that contained the topic of deep learning, object detection, and medical imaging in the title, abstract, or keywords were combined. Publications where segmentation was on the topic were excluded. The criteria for inclusion was limited to articles written in the English language.

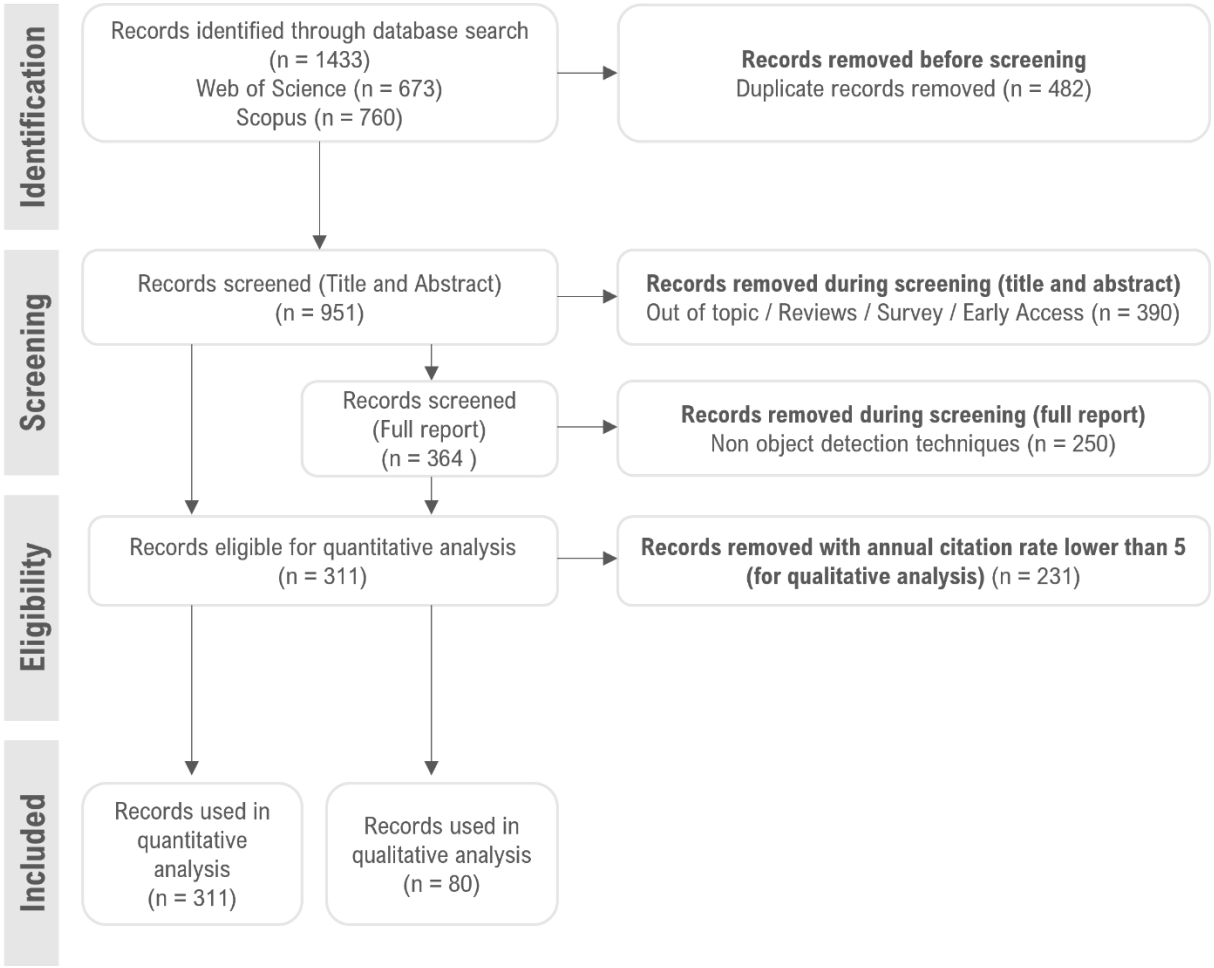

Figure 2.1 – Data collection and filtering process for object detection bibliometric analysis, following PRISMA methodology

The removal of duplicated articles resulted in a database of 951 reports that underwent a screening process for the title and abstract for enhanced filtering. Out of those, 390 were excluded, 197 were included in the final analysis, and 364 were subjected to a comprehensive report screening as the information provided in the title and abstract was not conclusive. Of the latter, 250 reports were rejected as they were not related to object detection, and the remaining were added to the final inclusion list. This resulted in a total of 311 papers that pertain to the topic of research. Those 311 documents are object of a quantitative analysis. From those, the records where the annual citation rate was lower than 5 (n = 231) were removed, and the remaining (n = 80) were further analyzed in a qualitative approach.

The tools used during this analysis was VosViewer (Van Eck & Waltman, 2010) version 1.6.18, ScientoPy (Ruiz-Rosero et al., 2019) version 2.1.0, and additional exploration using Python was carried out. The application of these advanced analytical tools enabled a thorough examination of the literature and provided valuable insights into the current state of the field.

2.3. Quantitative Analysis

In the current section, a quantitative analysis was undertaken on a total of 311 documents. This analysis was developed into five distinct perspectives, including an annual publication analysis, research area analysis, journal publication analysis, country publication analysis, author publication analysis, and keyword analysis.

2.3.1. Annual Publication Analysis

According to the data available in WoS and Scopus, a total of 311 publications were focused on object detection in medical imaging.

The most basic indicator of publications trends is the number of academic articles on the subject. As shown in Figure 2.2, the first publications in this field were published in 2018, with a consistent upward trend in the number of publications each year thereafter.

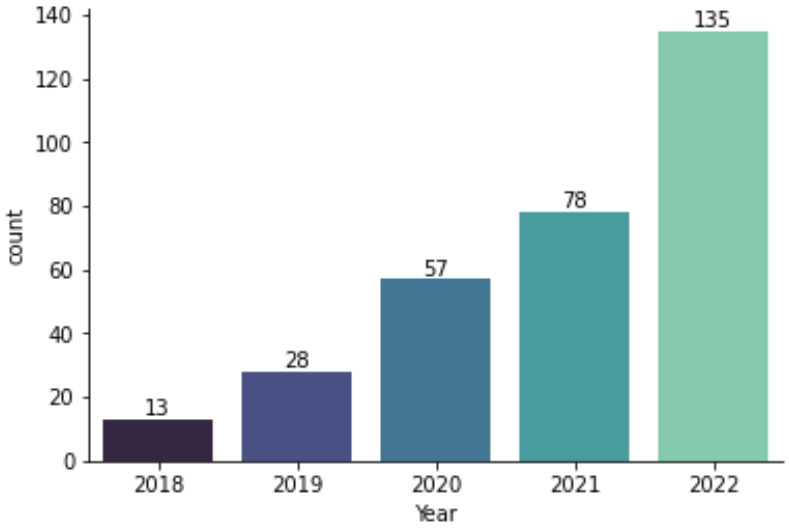

Figure 2.2– Growth of literature related to object detection in medical imaging until 2022

In 2018, only thirteen papers were published in the area. In 2019, more than twice as many articles as the previous year were published and in 2022 the number of publications was around 10 times higher than the first year. This change reflects the growing acceptance of this topic in the research community and highlights the growing interest and significance of this field among researchers and academics.

The citations of these 311 publications were analyzed as a whole, as shown in Table 2.2. In total, those articles were cited 5,052 times, and only 5.79% (n = 18) of the publications have more than 50 citations. On the other hand, 77.81% (n = 242) of the total publications have at least one citation.

Table 2.2 – Article papers and citations per year of object detection in medical imaging field

Year	TP	TC	AC	Total publications with citations				
				>=50	>=20	>=10	>=5	>=1
2018	13	1062	81.69	6	9	12	13	13
2019	28	799	28.54	6	14	22	23	27
2020	57	2358	41.37	6	18	36	47	57
2021	78	593	7.60	0	7	19	42	73
2022	135	240	1.78	0	0	2	14	72
%	100.00%			5,79%	15.43%	29.26%	44.69%	77.81%

Note. TP stands for total papers, TC for total citations, and AC for average article citations. An analysis of the number of publications according to their quantity of citations (higher or equal to 50,20,10,5 and 1) is performed at each year.

With the emergence of this field in 2018, the total number of publications in this year was significantly low. However, the average number of citations for those articles stands at a significant value of around 82 citations per publication. Despite a significant decrease to an average of 28 citations per publication in 2019, the publications in 2020 achieved a peak of 41 average citations per year. This, combined with the increase in the number of publications in recent years, demonstrates the potential growth in this field.

2.3.2. Research Area Analysis

According to Scopus, the selected publications are associated with 17 distinct fields, where publication can be included in more than one area. In total, 656 links have been established between the 311 publications and the 17 research areas. As expressed in Figure 2.3, the treemap clearly illustrates that most of the publications are included in the fields of Computer Science, Medicine, and Engineering, with 152, 139 and 107 records respectively.

When viewed from a broader medical perspective, the areas of “Medicine” (n = 139), “Biochemistry, Genetics and Molecular Biology” (n = 58), “Health professions” (n = 33), “Dentistry” (n = 12), “Neuroscience” (n = 7) and “Immunology and Microbiology” (n = 8) correspond to approximately 39,30% of the total associations.

From an engineering-oriented perspective, “Computer Science” (n = 152) and “Engineering” (n = 107) account for 39,60% of the total links.

In a multidisciplinary view, publications from “Materials Science” (n = 32), “Multidisciplinary” (n = 24), “Mathematics” (n = 25), “Physics and Astronomy” (n = 22), “Chemistry” (n = 13), “Chemical Engineering” (n = 12), “Agricultural and Biological Sciences” (n = 4), “Environmental Sciences” (n = 4), account for around 20,80% of the total number of links. Two of the publications were not found available on Scopus and were classified as “Others”.

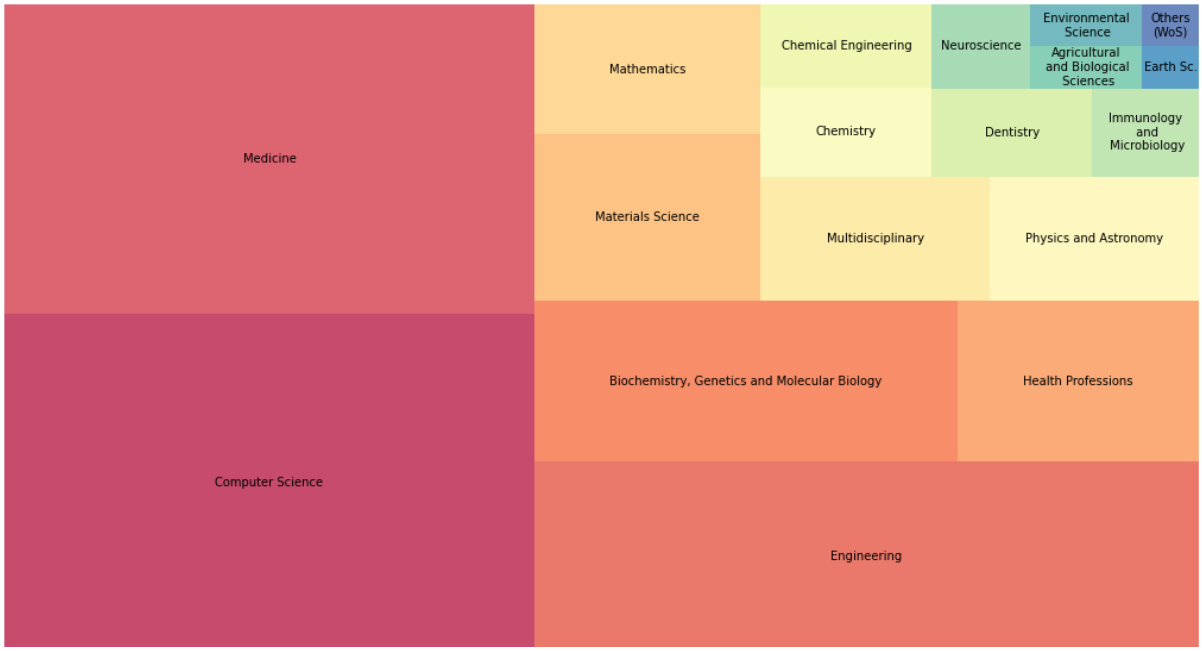

Figure 2.3 – Treemap with the distribution of publications across various research areas provided by Scopus

2.3.3. Source Analysis

The 311 publications obtained were published in 161 journals from different fields of study. There were 111 journals (68.94%) with only one publication associated, followed by 17 journals (10.60%) with two articles published. Notably, only two journals have more than ten publications, namely IEEE Access (n = 17) and Scientific Reports (n = 13). All journals with five or more publications, as shown in Table 2.3, account for 33.12% of the total publications and belong to the first quartile, excluding Diagnostics and Applied Sciences (Switzerland) Journals who belong to the second quartile.

Among these journals, those with the highest average number of citations based on the total number of publications were “Computers in Biology and Medicine” with an average of 159 citations, Scientific Reports with approximately 53 citations in average, and PLoS One, with an average of 24 citations.

Table 2.3 – Journals with five or more publications in the area

Rank	Journal Title	Country	SJR 2021	No. of Citations	No. of Articles	AC	H Index	IF Scopus	IF WoS
1	IEEE Access	United States	0.93	111	17	6.53	158	4.342	3.367
2	Scientific Reports	United Kingdom	1.01	683	13	52.54	242	4.543	4.379
3	PLoS ONE	United States	0.85	240	10	24.00	367	3.582	3.24
4	Sensors	Switzerland	0.80	29	9	3.22	196	4.352	3.576
5	Computers in Biology and Medicine	United Kingdom	1.31	1431	9	159.00	102	7.469	4.589
6	Diagnostics	Switzerland	0.66	28	7	4.00	35	3.912	3.706
7	IEEE Transactions of Medical Imaging	United States	4.05	89	7	12.71	233	12.018	10.048
8	Computer Methods and Programs in Biomedicine	Ireland	1.33	107	6	17.83	115	7.639	5.428
9	Biomedical Signal Processing and Control	Netherlands	1.21	53	5	10.60	84	5.861	3.88
10	Applied Sciences (Switzerland)	Switzerland	0.51	72	5	14.40	75	3.143	NA
11	Multimedia Tools and Applications	Netherlands	0.72	19	5	3.80	80	3.158	2.757
12	Medical Image Analysis	Netherlands	4.17	12	5	2.40	143	15.243	8.545
13	Medical Physics	United States	1.17	16	5	3.20	189	4.448	4.071

Note. The SJR, or SCimago Journal Rank, is a metric that ranks journals based on the number of citations received by their articles over a period of three years. The AC stands for average number of citations, a ratio between the number of citations and the number of publications. The H-Index measures the number of publications and how often they are cited, reflecting both the quantity and quality of a journal's publication. IF Scopus and IF WoS reflect the journal Impact Factor in each database.

2.3.4. Countries publication Analysis

In this analysis, each publication has been linked to countries with each author is associated with. Since each publication can have more than one author, and an author can be associated with more than one country, a paper can be associated with one or further countries. However, if a paper has two authors from the same country, it is only considered once, to avoid the duplication of the same article in the same country.

At the time of research, 51 countries have published at least one paper in the field. In table 2.4, the countries are sorted by the number of citations, and those where the number of citations is higher than one hundred is shown.

Table 2.4 – Countries with higher number of citations.

Rank	Country	P	C	AC
1	Japan	36	1791	49.75
2	United Kingdom	12	1465	122.08
3	Turkey	14	1451	103.64
4	Taiwan	17	1404	82.59
5	Singapore	2	1328	664.00
6	China	124	1273	10.27
7	United States	48	828	17.25
8	Hungary	1	364	364.00
9	South Korea	22	356	16.18
10	Australia	10	119	11.90
11	Germany	9	105	11.67
12	Hong Kong	6	116	19.33

Note. P stands for number of publications, C for Citations, AC is the average number of citations per publication.

Japan stands as the country with more citations (c = 1791) associated to their publications, followed by United Kingdom (c = 1465) and Turkey (c = 1451). However, when concerning to average number of citations, Singapore is at front with 664 citations per publication, Hungary in second with 364 and United Kingdom stands in the third place with around 122 publications.

Figure 2.4 illustrates the top ten most productive countries, with a supplementary analysis of co-authorship, which indicates collaboration among authors from different countries. China stands as the leading country in this field, with 124 publications, followed by the United States with 48 publications, and Japan with 36 publications. In terms of international collaboration, the United Kingdom displays the highest percentage of affiliations, with all of its articles being the result of international partnerships. In contrast, Australia has only one out of ten papers with international collaboration. In absolute terms, China has published 86 papers with international collaboration out of 124 total papers,

the United States has published 30 out of 48 papers with international affiliations, and Japan has published 26 papers out of 36.

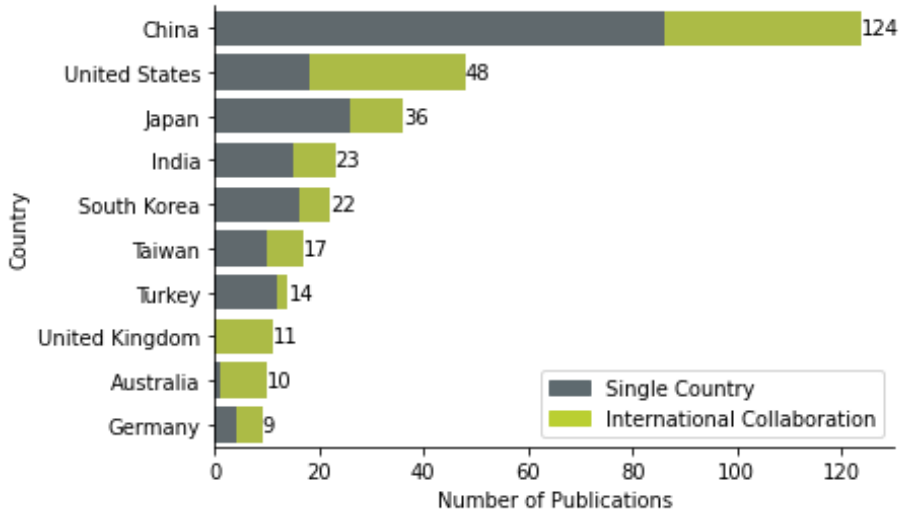

Figure 2.4 – The top ten most productive countries and their international co-authorship

When we assess the average number of citations of these three leaders in terms of publications, China presents a value of 10.27, United States has 17.25 and Japan has 49.75, which represents the 25th place, 15th place, and 7th place respectively among the 51 countries available.

2.3.5. Authors publication Analysis

In the set of 311 publications chosen, the author analysis was done through the SCOPUS ID of the authors. This approach was adopted to prevent the association of papers for authors with the same name, even if they are not the same researcher.

From this analysis, it was found that there were a total of 1870 related authors. Of those, 1752 authors were associated with only one paper, while only eight authors are associated with four or more papers, as can been seen in Table 2.5 which lists the most productive authors on the field.

Of the 1870 related authors, nine have three publications, 100 have two publications, and the remaining have just one single publication. Notably, all the most productive authors (authors with four or more publications) belong to Japan or China.

Table 2.5 – The most productive authors

Rank	Author	Country	P	C
1	Ariji, Eiichiro	Japan	7	195
2	Aniji, Yoshiko	Japan	7	195
3	Katsumata, Akitoshi	Japan	6	182
4	Fukuda, Motoki	Japan	5	165
5	Liang, Yixiong	China	5	81
6	Fujita, Hiroshi	Japan	5	176
7	Muramatsu, Chisako	Japan	4	118
8	Kuwada, Chiaki	Japan	4	107

Note. Authors with four or more publications on the field. P stands for the number of papers and C for the number of citations on those papers.

With regard to the number of citations, out of all the authors, 32 are cited more than 100 times, 15 have received more than 200 citations, and 401 authors have no citation associated with their work. Table 2.6 reveals the identities of all authors with more than 200 citations.

Table 2.6 – Authors with more than two hundred citations associated

Rank	Author	Country	P	C
1	Baloglu, Ulas Baran	United Kingdom	1	1306
2	Ozturk, Tulin	Turkey	1	1306
3	Rajendra Acharya U.	Singapore, Taiwan, Japan	1	1306
4	Talo, Muhammed	Turkey	1	1306
5	Yildirim, Eylul Azra	Turkey	1	1306
6	Yildirim, Ozal	Turkey	1	1306
7	Csabai, István	Hungary	1	364
8	Horváth, Anna	Hungary	1	364
9	Pollnar, Péter	Hungary	1	364
10	Ribli, Dezso	Hungary	1	364
11	Unger, Zsuzsa	Hungary	1	364
12	Lu, Le	United States	3	242
13	Yan, Ke	United States	3	240
14	Wang, Xiaosong	United States	1	228
15	Summers, Ronald M.	United States	1	228

Note. The authors with more than 200 citations are referred, as such as their country, the number of publications associated to them, and the total number of citations

As shown in Table 2.6, 14 out of the 15 authors with 200 or more citations associated have published only one paper on the field, and only two are associated with the publication of three papers, namely Le Lu and Ke Yan from the United States.

2.3.6. Keyword analysis

To gain insight into the central focus of the research, the commonly utilized words in the keyword section of the 311 selected articles were also evaluated. The analysis was conducted using the software ScientoPy. The word cloud in Figure 2.5 illustrates the most frequently employed terms of the 311 selected articles. It is evident, as the size of the concepts in a word cloud is determined by their frequency, that concepts such as deep learning (n = 138), object detection (n = 84), convolutional neural networks (n = 54), and artificial intelligence (n = 34) are among the most frequently used terms.

Concerning the algorithms used, Faster R-CNN appears on 19 articles as keyword, and the YOLO algorithm (regardless of the version) is shown as keyword in 31 records.

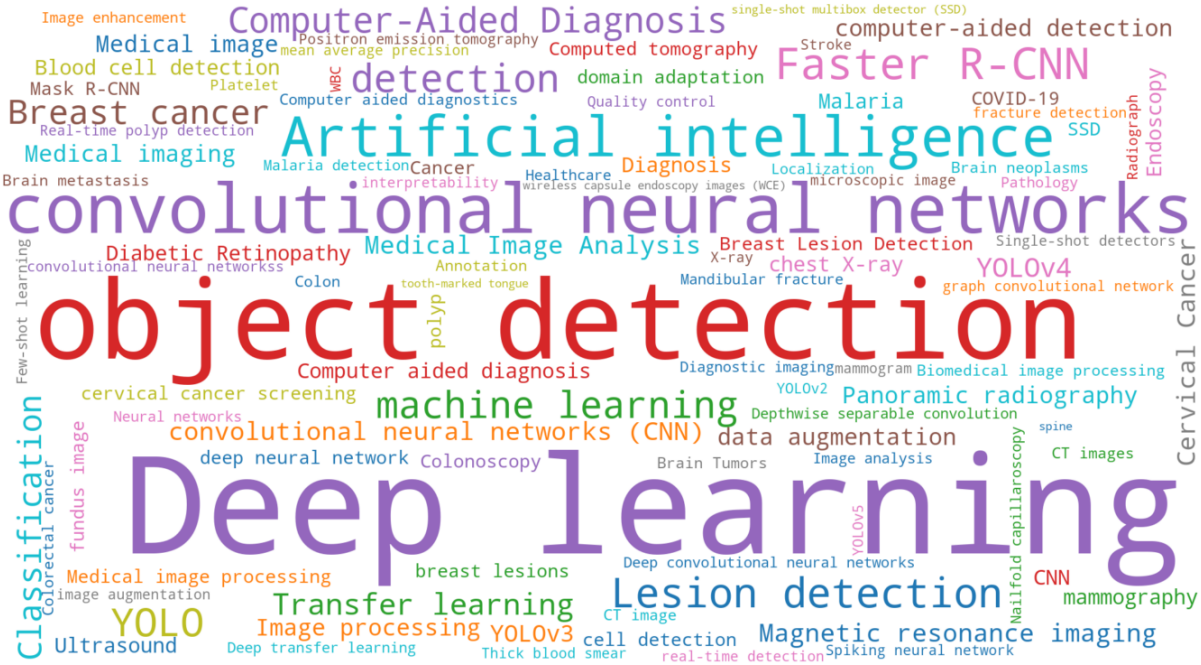

Figure 2.5 – Wordcloud with author's keyword analysis

Note: The size of a word or concept correspond to the frequency of occurrence.

In Figure 2.6, we visualize the network of co-occurrence of index keywords provided by VosViewer. Deep Learning appears 181 times, object detection has 123 appearances, and convolutional neural network 101 times.

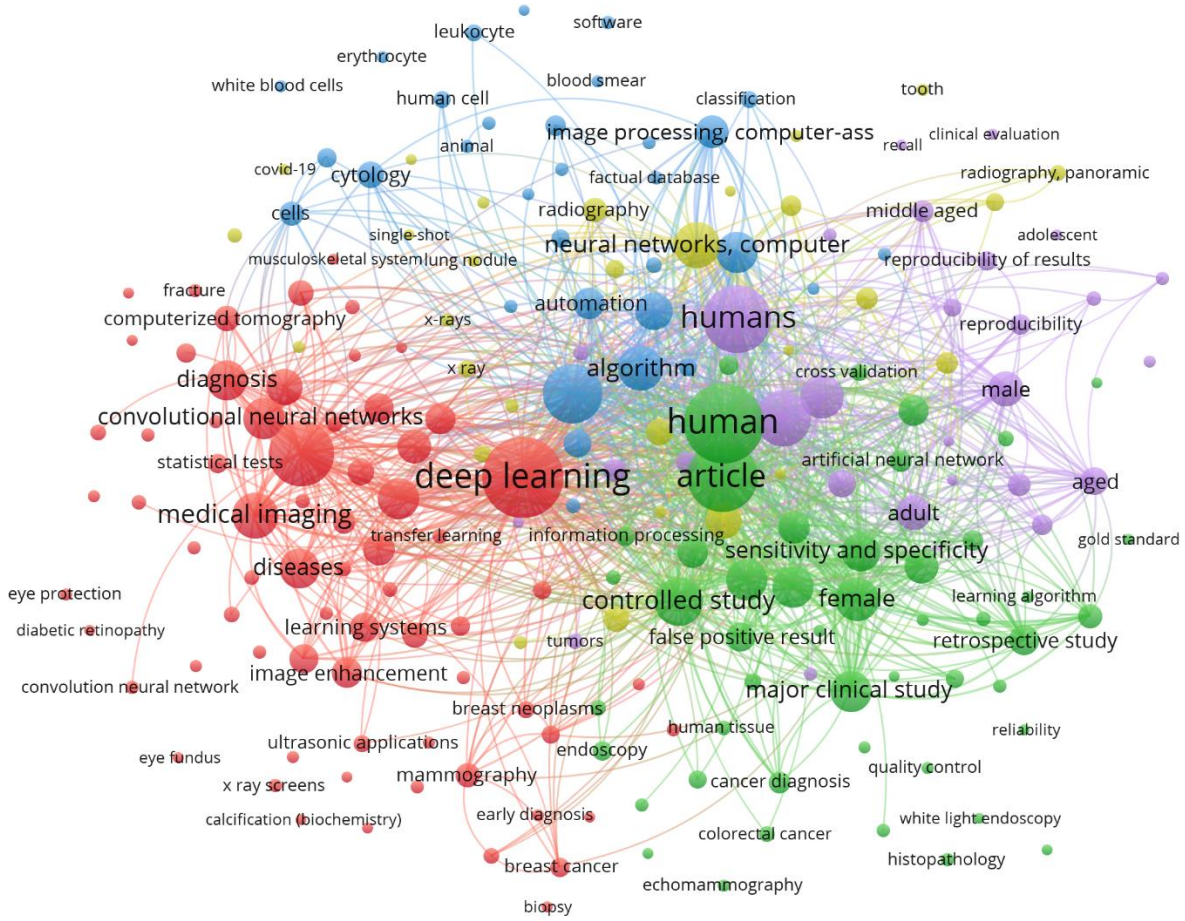

Figure 2.6 – Co-occurrence of index keywords network developed in VosViewer.

Note. Only keywords with 5 or more occurrences are shown.

2.4. QUALITATIVE ANALYSIS

The qualitative analysis was based on the publications where the annual citation rate was higher than 5. This selection resulted in a total of 80 publications, with 30 of them with more than 10 in the annual citation rate, as shown in Table 2.7.

Table 2.7 – Publications with annual citation rate higher than ten

Rank	Title (authors)	Annual Citation Rates
1	Automated detection of COVID-19 cases using deep neural networks with X-ray images (Ozturk et al., 2020)	435.33
2	Detecting and classifying lesions in mammograms with Deep Learning (Ribli et al., 2018)	72.80
3	DeepLesion: Automated mining of large-scale lesion annotations and universal lesion detection with deep learning (K. Yan et al., 2018)	45.60
4	White blood cells detection and classification based on regional convolutional neural networks (Kutlu et al., 2020)	31.00
5	Artificial intelligence using convolutional neural networks for real-time detection of early esophageal neoplasia in Barrett's esophagus (with video) (Hashimoto et al., 2020)	30.67
6	Evaluation of deep learning detection and classification towards computer-aided diagnosis of breast lesions in digital X-ray mammograms (Al-antari et al., 2020)	30.00
7	A deep learning approach to automatic teeth detection and numbering based on object detection in dental periapical films (H. Chen et al., 2019)	29.25
8	An improved deep learning approach for detection of thyroid papillary cancer in ultrasound images (H. Li et al., 2018)	19.60
9	Polyp detection during colonoscopy using a regression-based convolutional neural network with a tracker (R. Zhang et al., 2018)	19.60
10	An experimental study on breast lesion detection and classification from ultrasound images using deep learning architectures (Z. Cao et al., 2019)	19.50
11	Evaluation of an artificial intelligence system for detecting vertical root fracture on panoramic radiography (Fukuda et al., 2020)	19.33
12	Deep learning in diabetic foot ulcers detection: A comprehensive evaluation (Yap et al., 2021)	19.00
13	Melanoma diagnosis using deep learning techniques on dermatoscopic images (Jojoa Acosta et al., 2021)	18.50
14	Artificial intelligence detection of distal radius fractures: a comparison between the convolutional neural network and professional assessments (Gan et al., 2019)	18.25
15	Automated endoscopic detection and classification of colorectal polyps using convolutional neural networks (Ozawa et al., 2020)	17.33

16	Automatic detection and classification of radiolucent lesions in the mandible on panoramic radiographs using a deep learning object detection technique (Ariji et al., 2019)	16.50
17	An efficient real-time colonic polyp detection with YOLO algorithms trained by using negative samples and large datasets (I Pacal et al., 2022)	16.00
18	Malaria parasite detection in thick blood smear microscopic images using modified YOLOV3 and YOLOV4 models (Abdurahman et al., 2021)	15.50
19	Detection of cardiac structural abnormalities in fetal ultrasound videos using deep learning (Komatsu et al., 2021)	15.00
20	Deep learning approach to peripheral leukocyte recognition (Q. Wang et al., 2018)	14.60
21	Addressing class imbalance in deep learning for small lesion detection on medical images (Bria et al., 2020)	14.33
22	Efficient Multiple Organ Localization in CT Image Using 3D Region Proposal Network (X. Xu et al., 2019)	13.50
23	Machine learning approach of automatic identification and counting of blood cells (Alam & Islam, 2019)	13.25
24	Imaging based cervical cancer diagnostics using small object detection - generative adversarial networks (Elakkiya et al., 2022)	13.00
25	Use of artificial intelligence for detection of gastric lesions by magnetically controlled capsule endoscopy (Xia et al., 2021)	12.50
26	Detection of masses in mammograms using a one-stage object detector based on a deep convolutional neural network (Jung et al., 2018)	12.20
27	Tooth detection and classification on panoramic radiographs for automatic dental chart filing: improved classification by multi-sized input data (Muramatsu et al., 2021)	12.00
28	Early esophageal adenocarcinoma detection using deep learning methods (Ghatwary et al., 2019)	11.00
29	Development of an artificial intelligence system using deep learning to indicate anatomical landmarks during laparoscopic cholecystectomy (Tokuyasu et al., 2021)	10.50
30	A novel YOLOv3-arch model for identifying cholelithiasis and classifying gallstones on CT images (Pang et al., 2019)	10.25

Note. The publications are arranged in descending order based on their annual citation rate. Only those articles possessing an annual citation rate of greater than 10 are portrayed.

In the selected papers for a qualitative analysis, it is crucial to discern the incidence of algorithms used, the medical imaging modality most frequently associated with publications in the field of object detection, and the anatomical application areas where object detectors are applied. These topics are thoroughly reviewed in the following three subsections.

2.4.1. Used algorithms

Concerning the object detection algorithms applied on the 311 publications, Faster R-CNN stands as the most frequently used algorithm, with 32.5% of the publications, followed by YOLO algorithm (independently of the version) with 30% of the records, and SSD with 8.75%. RetinaNet and DetectNet have both 5 publications associated, and the remaining 13 papers (16.25%) apply other object detectors, as shown in Figure 2.7.

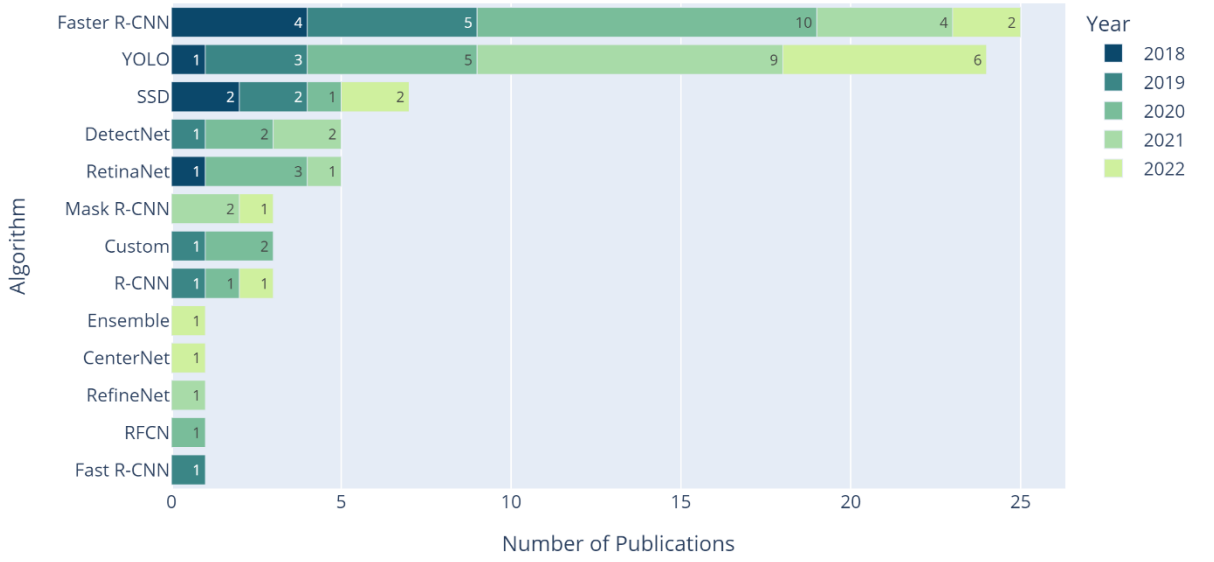

Figure 2.7 – Stacked bar plot on the algorithms explored by year

Faster R-CNN was the most widely utilized algorithm during the emergence of object detection in medical imaging. However, since 2019, the algorithm YOLO has gained significance in the field, with more publications in 2021 and 2022 than Faster R-CNN, which was previously the most popular algorithm. This fact reveals that the YOLO algorithm is gradually replacing the position previously held by Faster R-CNN and has gained increased popularity in recent years.

2.4.2. Modalities

There are several types of medical imaging modalities. As shown in Figure 2.8., the most frequent modalities include Computed Radiography (CR), Pathological Imaging, Endoscopy, and Computed Tomography (CT)

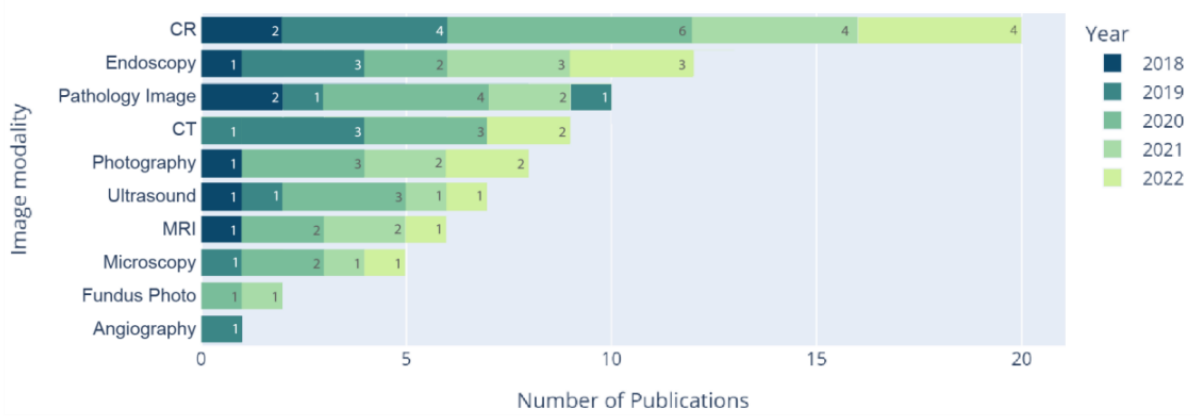

Figure 2.8 – Stacked barplot on the used image modalities by year

Of the 80 papers analyzed, 25% utilized CR as the primary image modality, followed closely by pathology imaging at 16.25%, and Endoscopy with 15% of the publications. Although it is unclear whether there have been significant changes in the type of images used over the years, it is evident that there has been an increase in the use of pathological images in 2021 and 2022 within this field.

2.4.3. Anatomical Application Areas

Over the past five years, the anatomical application areas where object detection has been applied have been diverse, but digital pathology and microscopy have emerged as the most popular, with a total of 16 publications, as illustrated in Figure 2.9. Closely following in popularity is the detection of abnormalities in abdominal organs, with 15 publications.

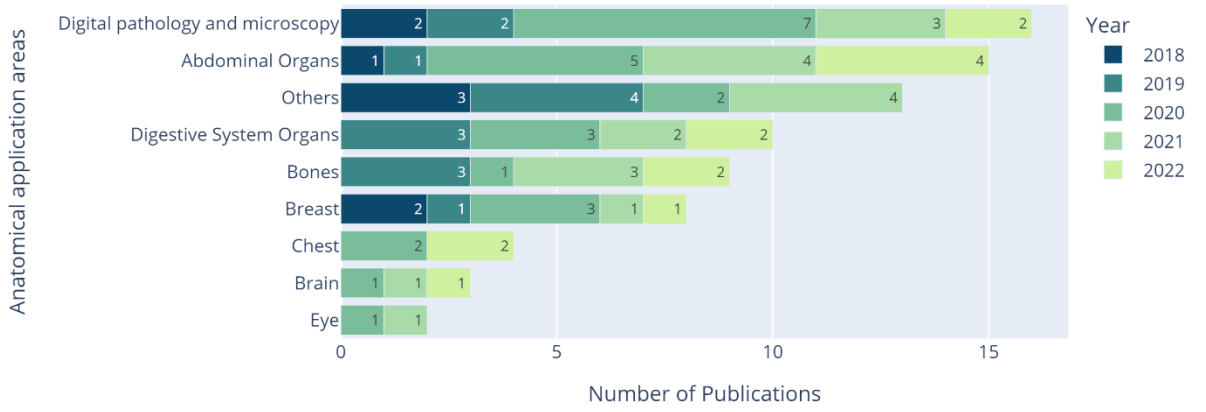

Figure 2.9 – Stacked barplot on the anatomical application areas explored by year

In the following subsections, we will conduct a qualitative analysis of the various publications distributed by their specific anatomical application area.

2.4.3.1. Abdominal organs

As shown in Table 2.8., on abdominal organs, the research is focused on colon, gallbladder, liver, stomach, cervix and uterus.

Table 2.8 – Publications in which object detection algorithms are applied to abdominal organs

Reference	C	Modality	Organ	Algorithm-based	Task
(R. Zhang et al., 2018)	98	E	Colon	YOLO	Polyp detection
(Ozawa et al., 2020)	52	E	Colon	SSD	Polyp detection
(I Pacal et al., 2022)	16	E	Colon	YOLO	Polyp detection
(Souaidi & Ansari, 2022)	7	E	Colon	SSD	Polyp detection
(K. Li et al., 2021)	14	E	Colon	RefineNet	Polyp detection
(Tokuyasu et al., 2021)	21	E	Gallbladder	YOLOv3	Detection of bile duct injury
(Pang et al., 2019)	41	CT	Gallbladder	YOLOv3	Detection of gall stones
(Xi et al., 2020)	17	US	Liver	Faster R-CNN	Abnormality detection
(Xia et al., 2021)	25	E	Stomach	Faster R-CNN	Detection of gastric lesions
(L Wu et al., 2022)	9	E	Stomach	YOLOv3	Detection of gastric lesions and neoplasm prediction
(Elakkiya et al., 2022)	13	PH	Cervix	R-CNN	Cervical cancer detection
(Xue et al., 2020)	27	PH	Cervix	Faster R-CNN	Cervical cancer detection
(Bai et al., 2020)	18	PH	Cervix	Faster R-CNN	Cervical cancer detection
(Komatsu et al., 2021)	30	US	Uterus	YOLOV2	Detection of cardiac structural abnormalities
(J. Dong et al., 2020)	27	US	Uterus	Custom	Detection of cardiac structural abnormalities

Note. Overview of papers using object detection techniques for the detection of abnormalities in abdominal organs, where E stands for endoscopy, CT for computed tomography, PI for pathology image, PH for photography, and US for ultrasound. In the header of the table, C stands for number of citations.

Five of the selected papers are dedicated to colon polyps detection study, using images obtained through endoscopy. Among those publications, YOLO and Faster R-CNN are applied twice, while RefineNet is applied once. In the first study in which YOLO is implemented, an improved framework is developed to detect polyps in a dataset consisting of more than 17,000 frames, yielding a precision of 88.6% and a recall of 71.6% at a prediction speed of 6.5 frames per second (R. Zhang et al., 2018). In the second study, YOLOv3 and YOLOv4 architectures are applied to two datasets, achieving a precision of 90.6% and a recall of 91.0% at a processing speed of 100 frames per second (I Pacal et al., 2022).

This last publication combines the YOLO structure with CSPNet to obtain a model with higher performance. In the SSD Implementations, one of the studies is validated in 7,077 images, where the model was able to identify correctly 1073 colon polyps out of the existent 1172, with a sensitivity of 90%, and a processing time of 20 frames per second (Ozawa et al., 2020). In the second implementation, a multiscale pyramidal fusion single-shot multibox detector network is applied , resulting in an mAP of 93.4% (Souaidi & El Ansari, 2022), with a testing speed of 62.5 FPS. RefineNet is also employed for polyp detection and applied on a combination of four distinct public data sets. In this study (K. Li et al., 2021), the authors compare RefineNet with different algorithms such as YOLO and Faster R-CNN, achieving the best performance among all the tested approaches with an mAP of 73.5%, while YOLOv3 and SSD achieve the fastest inference with 60 FPS.

Concerning to gallbladder organ, two studies have been conducted using YOLOv3, with distinct objectives. In the first one, endoscopic images were used to detect four landmarks in order to prevent bile duct injury in 23 short videos. The authors of this study recognize the system's suboptimal performance(Tokuyasu et al., 2021). In contrast, the second study uses 223,846 CT images to detect gall stones, achieving an mAP of 89.5% for granular and muddy stones (Pang et al., 2019).

Regarding the liver, one study has been conducted in which cysts and primary hepatic carcinoma detection in ultrasound imaging was accomplished using Faster R-CNN with a ResNet backbone. In this publication (Xi et al., 2020), the authors achieved an mAP of 60%, which represented an improvement over the results obtained with YOLOv2, which had an mAP of 51%.

In two distinct studies, the detection of gastric lesion on endoscopic imaging was approached. In the Faster R-CNN implementation (Xia et al., 2021), more than one million images were employed to train and test the system. The authors of this study enhance the significant difference in detection time between the automatic approach and the performance of clinicians. The results are reported by a value of sensitivity, ranging from 87.2% to 99.3%. In the second publication, where YOLOv3 was applied, the data was obtained from over 100,00 patients and the sensitivity of the final model ranged between 91.7% and 96.9% (L Wu et al., 2022).

Out of the fifteen papers related to abdominal organs, three are focused on cervical cancer detection, obtained through the application of two-stage object detection approaches on photographic images. In a study published in 2022 (C. Xu et al., 2022), a generative adversarial network was developed using an R-CNN, with the goal of creating a system to accurately detect small cervical cells and classify them into normal, precancerous, or cancerous. The remaining two publications employed Faster R-CNN. In the first study, a system was created based on 2853 images labeled as high-risk or low-risk, achieving an AUC value of 0.87 (Xue et al., 2020). In the second study, an improved version of Faster R-CNN was developed (Bai et al., 2020) and tested on 2528 images with 5294 lesion regions, achieving an mAP of 80.37%. This represented better results than those obtained with SSD (mAP = 74.27%), YOLOv3 (mAP = 77.11%), and Faster R-CNN (mAP = 75.89%).

Out of the fifteen papers identified, seven of them use one-stage object detectors.

2.4.3.2. Bones

Nine of the eighty studies selected based on the annual citation rate are employed in bone imaging, as exposed in Table 2.9.

Table 2.9 – Publications in which object detection algorithms are applied to bones

Reference	C	Modality	Algorithm	Task
(Gan et al., 2019)	73	CT	Faster R-CNN	Detection of distal radius fractures
(Ariji et al., 2019)	66	CR	DetectNet	Detection of mandible fractures
(Kuwana et al., 2020)	28	CR	DetectNet	Detection of maxillary sinus lesions
(Q. Huang et al., 2022)	7	US	YOLOv3	Detection of vertebras in spine
(Hardalaç et al., 2022)	7	CT	Ensemble	Detection of wrist fractures
(Rouzrokh et al., 2021)	13	CR	YOLOv3	Detection of hip dislocation
(Tsai et al., 2021)	12	MRI	YOLOv3	Detection of Lumbar Disc Herniation
(Liang et al., 2019)	22	CR	Faster R-CNN	Skeletal bone age assessment
(Y.-C. Li et al., 2021)	10	CT	YOLOv3	Detection of vertebral fractures

Note. Overview of papers using object detection techniques on bones, where CT stands for computed tomography, CR for computed radiography, US for ultrasound and MRI for magnetic resonance imaging. In the header of the table, C stands for number of citations.

On CT imaging, three studies use object detection to locate fractures, namely on the distal radius (Gan et al., 2019), on the wrist (Hardalaç et al., 2022) and on vertebras (Y.-C. Li et al., 2021). In the first one, a comparison is made between the automatic approach using a Faster R-CNN, and professional assessment. The authors conclude that the network was able to achieve a similar performance to that of the orthopedists and a performance superior to that of the radiologists. In a second study, wrist fractures are located with an Ensemble technique based on WBF. Ten base models are tried out, and the highest AP Score was obtained with the ensemble version obtained from the base learners. The authors enhance the potential of ensemble techniques on this field. In the third publication, vertebral fractures are detected using a YOLOv3 architecture, and one remark appointed by the authors is the highest interobserver reliability of the automatic system compared to human observers.

CR imaging is used on object detection studies with different purposes. Two publications, in particular, apply DetectNet (Tao et al., 2016), a network developed in DIGITS. The first study employs a model built on 210 training images to locate mandible fractures, achieving a sensitivity of 88% in the testing data (Ariji et al., 2019). The second study, also utilizing DIGITS, identifies maxillary sinus lesions, with a sensitivity of 100% on healthy and inflamed maxillary sinuses and 98% and 89% on distinct data sets for cyst maxillary sinus region (Kuwana et al., 2020). Another study trains YOLOv3 to assess the risk of hip dislocation on 1490 radiographs from dislocated cases and 91,094 from non-dislocated cases, achieving a sensitivity of 89% (Rouzrokh et al., 2021). A Faster R-CNN is employed to assess skeletal bone age by detecting the ossification centers of epiphysis and carpal bones (Liang et al., 2019), with a performance measured by MAE, achieving values of 0.48 and 0.51 in two tested data sets.

On ultrasound imaging, various object detectors, such as SSD, YOLOv3 and YOLOv4, were trained to detect the vertebrae in the spine (Q. Huang et al., 2022). While SSD achieved the highest mAP with a value of 90.84%, YOLOv3 was chosen as the final model due to its small inference time of around 7 FPS, which is around ten times faster than that of SSD.

An automatic detection system based on YOLOV3 and using MRI was also proposed to assist in the initial lumbar disc herniation exam for lower back pain (Tsai et al., 2021). In this study, an mAP of 92.4% was obtained at 550 images with data augmentation.

2.4.3.3. Brain

An MRI is a powerful tool for producing detailed images of the brain, which can be used to detect a wide range of conditions such as tumors, injuries, strokes, and blood vessel problems. In recent years, some object detection algorithms have been applied to this field, with the goal of automating the diagnostic process and increasing the accuracy of results. Table 2.10 exposes those studies.

Table 2.10 – Publications in which object detection algorithms are applied on brain imaging

Reference	C	Modality	Algorithm	Task
(Shelatkar et al., 2022)	9	MRI	YOLOv5	Brain tumor diagnosis
(Montalbo, 2020)	20	MRI	YOLOv4	Brain tumor diagnosis
(S. Zhang et al., 2021)	11	MRI	Faster R-CNN	Stroke lesion detection

Note. Overview of publications that employ object detection techniques on the brain, where MRI stands for magnetic resonance imaging. In the header of the table, C stands for number of citations.

In the selected publications for qualitative analysis, two studies applied the YOLO architecture to brain tumor diagnosis, while another study applied Faster R-CNN for stroke lesion detection. In the study where YOLOv5 is applied (Shelatkar et al., 2022), the authors compared the performance of different variants of the algorithm on 800 images to brain tumor detection. Unsurprisingly, the larger version of the model achieved the best mAP at 91.2% using YOLOv5x, while the smaller versions of YOLOv5 (YOLOv5n and YOLOv5s) achieved the worst performances with an mAP of 85.2% and 87%. However, the trade-off is that larger versions take more time to train.

In a second study for brain tumor diagnosis and detection, a transfer learning approach and fine-tuning techniques were applied to YOLOv4 to build a model on a data set of 3064 MRI scans, which was able to detect three distinct tumors, namely glioma, meningioma, and pituitary. This improved version of YOLO achieved a final mAP of 93.14% (Montalbo, 2020).

A third study on Brain MRI scans is focused on stroke lesion detection (S. Zhang et al., 2021). In this study, the authors compared the performance of Faster R-CNN, YOLOv3 and SSD, and concluded that SSD performed the best, with the highest mAP (89.77%). When comparing the inference time, YOLO was the fastest approach, but this advantage was reflected in a significant decrease in the mAP, with a value of 74.9%.

2.4.3.4. Breast

Breast cancer is the most common cancer diagnosed worldwide with 685,000 deaths associated in 2020, being estimated that in 2040 this value will raise to 1 million (Arnold et al., 2022). With eight publications associated, as expressed in Table 2.11, object detectors in breast imaging focus on the detection of lesions.

Table 2.11 – Publications in which object detection algorithms are applied on breast imaging

Reference	C	Modality	Algorithm	Task
(Ribli et al., 2018)	364	CR	Faster R-CNN	Breast lesion detection
(Al-antari et al., 2020)	90	CR	YOLO 9000	Breast lesion detection
(Z. Cao et al., 2019)	78	US	SSD	Breast lesion detection
(Bria et al., 2020)	43	CR	Custom	Breast lesion detection
(Jung et al., 2018)	61	CR	RetinaNet	Breast lesion detection
(Adachi et al., 2020)	21	MRI	RetinaNet	Breast lesion detection
(Y Liu et al., 2022)	6	CR	Mask R-CNN	Breast lesion detection
(Baccouche et al., 2021)	10	CR	YOLOv3	Breast lesion detection

Note. Overview of publications using object detection techniques for breast lesion detection, where CR stands for computed radiography, US for ultrasound and MRI for magnetic resonance imaging. In the header of the table, C stands for number of citations.

In these studies, various object detection models are applied, including YOLO, SSD, RetinaNet, Faster R-CNN, Mask R-CNN and custom-designed networks. The most commonly used imaging modality was CR, but ultrasound images and MRI were also used.

With regards to CR imaging, six studies were conducted using different algorithms as a basis. One of the publications (Ribli et al., 2018) applied Faster R-CNN to detect and classify malignant or benign lesions on a mammogram in a data set with 2620 screening exams. The system achieved an AUC of 0.85, with 10% of the malignant lesions being missed. In the same year, RetinaNet was tested in seven different data sets and, according to the authors, outperformed the conventional mass detection models (Jung et al., 2018). In 2020, two publications addressed this task. One of these applied YOLO 9000 on two distinct data sets, comprising 600 images and 360 images respectively. On these data sets, 99.17% of the total cases were predicted correctly, with 0.83% being false detections (Al-antari et al., 2020). In the second case, a custom network was built to address class imbalance for small lesion detection on breast (Bria et al., 2020). Another publication used a YOLO-based model on 235 mammograms publicly available, and on 487 privately collected mammograms, achieving a detection accuracy rate of more than 95% for the distinct datasets used, and an inference time of around 0.55 seconds per image (Baccouche et al., 2021). In 2022, a Mask R-CNN-based framework was developed and implemented on a public dataset and on an in-house dataset. The performance of the model was analyzed only using recall, achieving values of 0.82 for the first data set, and 0.87 for the private data set, with an IoU threshold of 0.5 (Y Liu et al., 2022).

In 2019, ultrasound images were applied on distinct object detectors, including Fast R-CNN, Faster R-CNN, YOLO, YOLOv3 and SSD. YOLO and SSD performed significantly better than the other methods, and SSD was chosen as the most suitable model for this task, achieving an mAP of 96.89%, in a total of 1,041 cases (Z. Cao et al., 2019).

Regarding MRI imaging, only one publication was included in this qualitative analysis (Adachi et al., 2020), using RetinaNet. The authors compared the sensitivity, specificity and AUC between the AI system and human readers, concluding that the former had better diagnostic performance.

2.4.3.5. Chest

In thoracic image analysis of both computed radiography and tomography, the detection of pulmonary tuberculosis and COVID-19 were the most commonly addressed applications, as seen in Table 2.12.

Table 2.12 – Publications in which object detection algorithms are applied on chest imaging

Reference	C	Modality	Algorithm	Task
(Ozturk et al., 2020)	1306	CR	YOLO 9000	COVID-19 detection
(C. Yan et al., 2022)	9	CT	CenterNet	Pulmonary tuberculosis detection
(Mahajan et al., 2022)	7	CR	SSD	COVID-19 detection
(Y. Xie et al., 2020)	19	CR	Faster R-CNN	Pulmonary tuberculosis detection

Note. Overview of publications that apply object detection techniques on chest imaging for COVID and pulmonary tuberculosis detection, where CT stands for computer tomography and CR for computed radiography. In the header of the table, C stands for number of citations.

SSD and YOLO 9000 are applied to COVID detection on CR imaging. Several backbones have been tested with SSD, including VGG, Residual Network, DarkNet and DenseNet and their performance has been compared. DenseNet achieves the highest performance, with a precision of 0.93 and a recall of 0.94 (Mahajan et al., 2022). The most cited publication in our qualitative research focuses on the implementation of YOLO to COVID detection. The proposed model is developed to provide accurate diagnostics for binary classification (COVID vs. No-findings) and multi-class classification, where Pneumonia is added to the former classes. The authors report an accuracy of 98.08% on the binary problem and 87.02% for the multi-class problem (Ozturk et al., 2020).

In pulmonary tuberculosis detection, a Faster R-CNN adapted model has been applied to a multi-class problem on CT images, which encompasses co-existing cases of exudation, calcification, nodules, miliary tuberculosis, and other related conditions. This model achieves an mAP of 53.74%, outperforming the original Faster R-CNN model which achieved 22.66% mAP, and surpassing the results obtained with FPN, which achieved 50.96% mAP (Y. Xie et al., 2020). Additionally, CenterNet has also been used on 892 CT scans for pulmonary tuberculosis detection, yelding an mAP of 68% (C. Yan et al., 2022).

2.4.3.6. Digestive System Organs

Concerning the digestive system, object detection techniques have been applied to various forms of medical imaging, including esophagus imaging obtained through endoscopy, tongue images obtained through photography and teeth images obtained through CR imaging, as seen in Table 2.13.

Table 2.13 – Publications in which object detection algorithms are applied to digestive system organs

Reference	C	Modality	Organ	Algorithm	Task
(Hashimoto et al., 2020)	92	E	Esophagus	YOLOV2	Early esophageal neoplasia detection
(Ghatwary et al., 2019)	44	E	Esophagus	SSD	Early esophageal adenocarcinoma detection
(X. Li et al., 2019)	29	PH	Tongue	R-CNN	Tooth-marked tongue detection
(H. Chen et al., 2019)	117	CR	Teeth	Faster R-CNN	Tooth detection
(Fukuda et al., 2020)	58	CR	Teeth	DetectNet	Root fracture detection
(Muramatsu et al., 2021)	24	CR	Teeth	DetectNet	Tooth detection
(Thanathornwong & Suebnukarn, 2020)	26	CR	Teeth	Faster R-CNN	Periodontal compromised teeth detection
(Estai et al., 2022)	8	CR	Teeth	Faster R-CNN	Permanent teeth detection
(H. Chen et al., 2021)	11	CR	Teeth	Faster R-CNN	Dental disease detection
(M. Liu et al., 2022)	5	CR	Teeth	Faster R-CNN	Marginal bone loss around implants detection

Note. Overview of publications that apply object detection techniques on digestive system organs, where CR stands for computed radiography, E for endoscopy and PH for photography. In the header of the table, C stands for number of citations.

In particular, YOLOv2 and SSD have been applied to endoscopy imaging, with the former being used to detect early stages of esophageal neoplasia (Hashimoto et al., 2020) and the latter being used to detect early stages of esophageal adenocarcinoma (Ghatwary et al., 2019). In the first study, a total of 916 images were used to train the model and 458 images were used to test it, achieving an mAP of 75.33% for narrow bone imaging and 80.19% for near focus images, at a speed of 45 FPS. In the second study, the performance of SSD and Faster R-CNN were compared in terms of average recall rate (ARR), average precision rate (APR), sensitivity (SE), specificity (SP), and F-measure (FM) using a 5-fold cross-validation method. SSD achieved the highest values with 0.7 on APR, 0.90 in SE, 0.88 in SP and 0.88 in FM, being surpassed by Faster R-CNN only in ARR with a difference of 0.04.

In a tongue photography study, the goal was to identify areas on the tongue that were tooth-marked, using a data set containing 641 images, 297 of which were tooth-marked. An adapted version of R-CNN was applied and an average accuracy of 72.7%, a TPR of 69.1%, and a TNR of 76.2% were reported by the authors (X. Li et al., 2019).

Tooth detection appears to be a prevalent subject in the realm of object detection frameworks in medical imaging. Of the 80 studies analyzed, seven were dedicated to tooth detection and related tasks. The Faster R-CNN algorithm was applied in five of these studies. In a first study (H. Chen et al., 2019), the model achieved a precision and recall rate of over 90%, with a mean average precision (mAP) of 91%. These values were only attainable after implementing several postprocessing procedures.

In another study, Faster R-CNN was employed for Periodontal compromised teeth detection (Thanathornwong & Suebnukarn, 2020), resulting in an accuracy rate of 80%, a positive predictive rate of 81%, a sensitivity of 84%, a specificity of 88%, and a F-measure of 81%. The authors of this study concluded that the system exhibited satisfactory detection abilities.

Faster R-CNN was also applied to permanent teeth detection in a study (Estai et al., 2022) where the authors reported results of 0.99 for recall and precision. Additionally, the algorithm was used on another work (H. Chen et al., 2021) to detect dental diseases such as decay, periapical periodontitis, and periodontitis at varying levels of severity. The results showed that decay and periapical periodontitis lesions were detected with precision, recall, and average precision values of less than 0.25 for mild levels, while moderate and severe levels had values ranging from 0.2-0.3 and 0.5-0.6, respectively.

In the more recent study where Faster R-CNN was applied, the goal was to detect Marginal bone loss around dental implants (M. Liu et al., 2022). The system was evaluated using a dataset of 1,670 images, and was measured in terms of sensitivity, specificity, diagnostic error rate, omission diagnostic rate, and positive predictive value. Kappa statistics were also compared between the system and dental clinicians, and the authors concluded that there was a high level of agreement between the automatic system and the clinicians.

On DetectNet, two studies are implemented, both applied to CR imaging. In one of these studies, the primary objective was the detection of root fractures (Fukuda et al., 2020). The study employed 300 images, containing a total of 330 root fractures, and out of these, 267 were successfully detected by the automatic approach, while twenty were falsely detected, resulting in a recall of 0.75, a precision of 0.93, and a F measure of 0.83. In a second study (Muramatsu et al., 2021) DetectNet was applied to tooth detection. However, the authors acknowledged several limitations in their study, such as the small data set used, which was composed of only 75 training cases and 25 test cases.

2.4.3.7. Digital pathology and microscopy

The growing availability of large-scale gigapixel whole-slide images of tissue specimens has made digital pathology and microscopy a highly popular application area for deep learning techniques

(Litjens et al., 2017). These imaging modalities have been applied to various purposes, such as cell detection and malaria detection, among others, as seen in Table 2.14.

Table 2.14 – Publications in which object detection algorithms are applied on digital pathology and microscopy

Reference	C	Modality	Organ	Algorithm	Task
(Kutlu et al., 2020)	93	PI	Blood	Faster R-CNN	White blood cells detection
(Kawazoe et al., 2018)	37	PI	Tissue	Faster R-CNN	Glomerular detection in multistained human renal tissue
(Kassis et al., 2019)	27	M	Others	Faster R-CNN	Human intestinal organoid detection
(Rosati et al., 2020)	16	M	DNA	Faster R-CNN	Detection of DNA damage
(Xiang et al., 2020)	29	PI	Cervical cells	YOLOv3	Cervical cell recognition
(Abdurahman et al., 2021)	31	M	Blood	YOLO	Malaria detection
(Q. Wang et al., 2018)	73	PI	Blood	SSD	Peripheral leukocyte recognition
(Alam & Islam, 2019)	53	PI	Blood	YOLO	Blood cells detection
(Khandekar et al., 2021)	18	PI	Blood	YOLOv4	Blast cell detection
(Manescu et al., 2020)	15	PI	Blood	RetinaNet	Malaria detection
(Koga et al., 2022)	9	PI	Brain tissue	YOLOv3	Alzheimer's diagnose
(J Hung et al., 2020)	22	PI	Blood	R-CNN	Cells detection
(Mochalova et al., 2020)	15	M	Others	Faster R-CNN	Detection of cellular organelles
(W. Yu et al., 2021)	12	PI	Liver and Kidney tissue	RetinaNet	Diatom detection
(Marzahl et al., 2020)	24	M	Others	RetinaNet	Pulmonary Hemosiderophages detection
(A. Chen et al., 2022)	8	M	Others	YOLOv4	Sperm detection

Note. Overview of publications that apply object detection techniques on digital pathology and microscopy imaging, where PI stands for pathological imaging and M for microscopy imaging. In the header of the table, C stands for number of citations.

The YOLO algorithm is regarded as the most prevalent among these studies, with six out of sixteen publications devoted to its application. In particular, three studies have explored the use of YOLO in the analysis of blood samples, specifically for detecting blood cells (Alam & Islam, 2019), blast cells (Khandekar et al., 2021) and malaria (Abdurahman et al., 2021). In the first study, YOLO was trained and tested using 364 images with red blood cells, white blood cells and platelets. The results showed an mAP of 82.58% using VGG16 backbone. Another study (Kutlu et al., 2020) compared YOLOv3 with other popular models, including R-CNN, Fast R-CNN, Faster R-CNN, and SSD. Ultimately, Faster R-CNN model arose as the winner with an mAP of 71% and an inference time of 609 milliseconds per image with VGG16. A third study on blood smears applied 14,700 images of peripheral leukocytes with eleven distinct categories, resulting in an mAP of 93.10% for SSD and an inference time of 53 milliseconds per image. Although YOLOv3 also performed well, it achieved only 92.10% mAP. The blast cell detection study (Khandekar et al., 2021) applied YOLOv4 on blood smears to aid in the early diagnosis of leukemia and achieved an mAP of 95.57% with an inference value of 50 FPS. In the malaria detection study with microscopic images, YOLOv3 and YOLOv4 were compared, with the last being the superior model, achieving an mAP of 96.32% and an inference rate of 29.60 frames per second. Another study (Marzahl et al., 2020) applied RetinaNet to detect malaria in 169 microscopic images for train and 130 for validation, with results reported in terms of sensitivity (0.92), specificity (0.90), accuracy (0.91), positive predicted value (0.92) and negative predicted value (0.90).

The YOLO object detector has also been used for cervical cells recognition (Xiang et al., 2020) and sperm detection (A. Chen et al., 2022). In the first study, YOLOv3 was applied to cervical cytology screening and achieved an mAP of 63.4%, with 95.7% of sensitivity and 67.8% of specificity, on a data set of 12,909 images split into 10 distinct categories.

In the sperm study, with 125,000 objects annotated, YOLOv4 and other models such as YOLOv3, SSD, RetinaNet, Faster R-CNN were compared, with YOLOv4 and SSD achieving the highest mAPs of 40.50% and 41.98%, respectively. However, YOLOv4 was selected due to a higher AP on the class impurity.

The last study on YOLO was on Alzheimer's diagnose (Koga et al., 2022). The YOLOv3 was trained to detect five tau lesion types, including neuronal inclusions, neuritic plaques, tufted astrocytes, astrocytic plaques and coiled bodies, on 2,522 immunostained slides images of the motor cortex, achieving a maximum mAP of 74.4%.

The Faster R-CNN algorithm was the second most used among the studies, appearing on five out of the sixteen work available. In the study of glomerular detection in multistained human renal tissue (Kawazoe et al., 2018), Faster R-CNN was used to automate the task, and was trained on 33,000 images. The results were reported in terms of recall, precision, and F measure for four different classes. Detection of human intestinal organoids (Kassis et al., 2019) was performed using Faster R-CNN on a data set of 1,750 images patches with 14,242 ground truths, resulting in an mAP of 80%. The authors argue that the quality detection of the automatic detection was similar to human capability, but substantially faster. Another study, examining DNA damage detection (Rosati et al., 2020), reported an mAP of 74% with Faster R-CNN, while SSD and YOLOv3 achieved mAPs of 22% and 30%, respectively. The detection of cellular organelles (Mochalova et al., 2020) was also considered using Faster R-CNN, with results reported solely for classification purposes.

The RetinaNet algorithm was used in three studies. Excluding the malaria study also reported in this subsection, this algorithm was used on liver and kidney tissue for diatom detection (W. Yu et al., 2021), and to detect Pulmonary Hemosiderophages (Marzahl et al., 2020). In the first study, an average precision of 0.82 and an average recall of 0.88 was obtained. In the second study, a RetinaNet was trained on seventeen completely annotated cytology whole slide images (WSI) containing 78,047 hemosiderophages and achieved an mAP of 66% for the five classes approached, exceeding human expert concordance.

The R-CNN algorithm was only applied in one of the studies for cell detection (J Hung et al., 2020), with a dataset of 600 training images and 100 testing images, and achieved an mAP of 82%.

2.4.3.8. Eye

The field of ophthalmologic imaging has experienced significant growth in recent years, but it is only in recent times that deep learning techniques have been harnessed for eye image analysis, with a majority of the studies centering around color fundus imaging(Litjens et al., 2017).

The qualitative analysis of the eighty studies revealed that only two of them have been applied to eye imaging, as expressed in Table 2.15.

Table 2.15 – Publications in which object detection algorithms are applied on eye imaging

Reference	C	Modality	Algorithm	Task
(Nazir et al., 2020)	29	FP	Faster R-CNN	Diabetes-Based Eye Disease Detection
(Ju et al., 2021)	12	FP	YOLOv3	Lesion detection

Note. Overview of publications that apply object detection techniques on eye imaging, where FP stands for fundus photography. In the header of the table, C stands for number of citations.

The first study used FRCNN, a Keras implementation of Faster R-CNN, to identify diabetes-based eye diseases, including diabetic retinopathy, diabetic macular edema and glaucoma (Nazir et al., 2020). The authors achieved an mAP of 94% with the proposed model, outperforming two other approaches, SSPnet and R-CNN, which achieved mAPs of 85% and 89%, respectively. In the second study (Ju et al., 2021), the authors used YOLOv3 to identify lesions in eye images. The results were reported in terms of average precision (0.08 and 0.52), average recall (0.86 and 0.91) and F1 Score (0.16 and 0.66), for both the number of lesions and number of images analyzed.

2.4.3.9. Others

This final subsection lists papers that address multiple applications, where the organ is not unique, and diverse applications that were not included in the previous subsections, as seen in Table 2.16.

Out of the thirteen studies analyzed in this subsection, seven of them are focused on object detection without specifying a particular anatomical area.

Table 2.16 – Publications in which object detection algorithms are applied on various topics

Reference	C	Modality	Organ	Algorithm	Task
(Yap et al., 2021)	38	PH	Foot	Faster R-CNN	Diabetic foot ulcers detection
(Jojoa Acosta et al., 2021)	37	PH	Skin	Mask R-CNN	Melanoma detection
(Chang et al., 2020)	19	PH	Skin	Faster R-CNN	Scalp health diagnosis
(H. Li et al., 2018)	98	US	Thyroid	Faster R-CNN	Thyroid papillary cancer detection
(H. Duan et al., 2019)	27	A	Vessels	Custom	Intracranial aneurysm detection
(Ariji et al., 2021)	10	CT	Lymph nodes	DetectNet	Cervical lymph nodes detection
(K. Yan et al., 2018)	228	CT	Several	Faster R-CNN	Lesion detection
(X. Xu et al., 2019)	54	CT	Several	Faster R-CNN	Organ localization
(Zeng et al., 2020)	20	US	Several	RFCN	Region detection in Ultrasounds
(Lan et al., 2019)	25	E	Several	Fast R-CNN	Wireless capsule endoscopy abnormal pattern detection
(K. Yan et al., 2021)	12	CT	Several	Mask R-CNN	Lesion detection
(Ding et al., 2019)	22	E	Several	YOLO	Upper gastrointestinal disease
(K. Kim et al., 2018)	26	MRI	Several	SSD	Tuberculous and pyogenic spondylitis diagnosis

Note. Overview of publications that apply object detection techniques with various purposes, where PH stands for photography, US for ultrasound, A for angiography, CT for computed tomography, E for endoscopy and MRI for magnetic resonance imaging. In the header of the table, C stands for number of citations.

For the lesion detection task, two studies have been carried out. The first study (K. Yan et al., 2018) uses an algorithm based on Faster R-CNN and a dataset that was compiled with 23,735 lesions from 32,120 CT scans. The results shows a sensitivity rate of 81.1%, with five false positives per image. The second study (K. Yan et al., 2021) proposes an universal lesion detection in CT imaging. The data set was created by merging images from data sets related to the liver, lymph nodes and the entire body, resulting in a total of over 30,000 lesions. The average sensitivity of the proposed model was reported to be 47.6%.

Lesions in various gastrointestinal regions, including the esophagus, stomach, pylorus, duodenum, and cardia, can be detected in gastroscopic images using YOLOV3 (Ding et al., 2019). The authors reported an mAP of 58.10%. Additionally, this performance was compared against the performances of SSD and RetinaNet.

Organ localization is the focus of a study (X. Xu et al., 2019) where Faster R-CNN is used in CT imaging. In this work, the authors combined two datasets, one containing images of eleven body organs and the other containing images of twelve head organs, achieving an mAP of 73.01% for the former, and 84.78% for the latter.

A study on region detection in ultrasounds (Zeng et al., 2020) employs the use of Faster R-CNN, and its performance is compared against that of SSD and RFCN. This study diagnoses a variety of diseases, including gallbladder stones and polyps, hydronephrosis, kidney stones, renal cysts, hemangiomas, and fatty liver. Out of the three models, RFCN emerged as the top performer with an mAP of 78.7%, followed closely by SSD with 76.7% and Faster R-CNN with 75.4%.

Another study uses Fast R-CNN based model on wireless capsule endoscopy abnormal pattern detection on more than 7,000 annotated images. The final proposal achieved an mAP of 72.3% (Lan et al., 2019).

In the field of medical imaging, the diagnosis of tuberculous and pyogenic spondylitis is carried out using an SSD approach on MRI scans (K. Kim et al., 2018). The results of this study show that the model achieved an AUC of 0.80, with a sensitivity of 85% and a specificity of 67.9%.

Additionally, the skin is also the subject of research in automatic detection systems. A study was conducted on melanoma detection using Mask R-CNN (Jojoa Acosta et al., 2021), with the purpose of identifying potential areas of skin lesion and cropping those areas for further analysis by a classifier algorithm. Another study analyzed scalp health and analyzed four distinct scalp hair symptoms (Chang et al., 2020). The results showed that Faster R-CNN achieved an mAP of 91.75%, while SSD achieved an mAP of 87.16%.

The detection of diabetic foot ulcers is assessed using Faster R-CNN on a photographic imaging dataset, consisting of 2,000 training images, 200 validation images, and 2,000 testing images (Yap et al., 2021). In this study, the authors compare the performance of different algorithms in terms of mAP and other metrics, with Faster R-CNN achieving an mAP of 69.40%, YOLOv3 an mAP of 65.60%, Cascade DetNet an mAP of 63.94%, YOLOv5 an mAP of 62.94%, and EfficientDet an mAP of 56.94%.

In CT imaging for patients with oral cancers, a DetectNet is utilized for cervical lymph node detection (Ariji et al., 2021). The sensitivity for metastatic lymph nodes was identified as 73%, and for non-metastatic lymph nodes, a sensitivity of 52.5% was observed.

For intracranial aneurysm detection on digital subtraction angiography, a custom object detection network is employed on vessel imaging (H. Duan et al., 2019). The proposed architecture attained an accuracy of 93% and an AUC of 0.942. In this study, the proposed network is also compared to YOLOv3 and RetinaNet, both with lower performance results.

2.5. CONCLUSION

This study conducted a thorough bibliometric analysis of the relevant studies in the rapidly growing field of the application of object detection in medical image analysis. A PRISMA methodology was

applied to identify 311 relevant publications in this field. Our research findings led us to understand that this is a recent area of study, but one in which research is steadily increasing and has the potential to grow. Most of the publications were included in the research areas of Medicine, Computer Science, and Engineering, and were published in top journals belonging to the first quartile. China, the United States, and Japan were the most productive countries, with the United States being the country with the highest level of international cooperation among these three countries. The majority of the authors with the highest number of citations were associated with only one published paper, and only six authors had five or more publications associated. In the authors' keyword analysis, it was clear that concepts such as object detection, deep learning, artificial intelligence, and convolutional neural networks were prevalent, while in the index keyword analysis, medical concepts were introduced with higher relevance.

From the 311 papers that were analyzed, 80 of them underwent a rigorous qualitative analysis from three distinct perspectives: the algorithms utilized, the imaging modalities employed, and the anatomical areas of application. The results of the analysis revealed that a majority of the publications employed Faster R-CNN and YOLO (regardless of their specific variant) in their research, indicating that both one-stage and two-stage object detectors play a significant role, depending on the trade-off between performance and training speed. It is clear, however, that there is no consensus on the best algorithm to use, since its selection depends on several factors, such as the quantity of data available, the high need for precision versus the high need for inference speed, and even the particular characteristics of the objects of interest in the problem at hand.

With regards to medical imaging modalities, object detection algorithms have been applied to a wide variety of imaging techniques, and this diversity is also reflected in the broad scope of anatomical areas to which these automatic systems have been applied. It is also possible to conclude that the detection of colon polyps in endoscopies, breast lesions in mammography, and teeth detection using computer radiographies are among the most popular areas of application.

One of the primary challenges in this study was to evaluate the performance of various models across different tasks. There is a lack of consensus among researchers on which metrics should be used to make such comparisons, with some advocating for the use of commonly accepted metrics such as Average Precision (AP) and mean Average Precision (mAP) (Padilla et al., 2021), while others prefer to utilize alternative metrics such as sensitivity, specificity and accuracy.

It is also important to note that some studies highlight the high capability of these automatic systems in detecting the objects of interest, sometimes with equal or higher performance than human specialists.

Considering these findings, it is reasonable to conclude that object detection in medical imaging is a rapidly developing research area, and this work can have a significant impact in helping to understand the direction of further research in this field.

3. OBJECT DETECTION FOR AUTOMATIC CANCER CELL COUNTING IN ZEBRAFISH XENOGRAFTS

Cell counting is a frequent task in medical research studies. However, it is often performed manually; thus, it is time-consuming and prone to human error. Even so, cell counting automation can be challenging to achieve, especially when dealing with crowded scenes and overlapping cells, assuming different shapes and sizes. In this paper, we introduce a deep learning-based cell detection and quantification methodology to automate the cell counting process in the zebrafish xenograft cancer model, an innovative technique for studying tumor biology and for personalizing medicine. First, we implemented a fine-tuned architecture based on the Faster R-CNN using the Inception ResNet V2 feature extractor. Second, we performed several adjustments to optimize the process, paying attention to constraints such as the presence of overlapped cells, the high number of objects to detect, the heterogeneity of the cells' size and shape, and the small size of the data set. This method resulted in a median error of approximately 1% of the total number of cell units. These results demonstrate the potential of our novel approach for quantifying cells in poorly labeled images. Compared to traditional Faster R-CNN, our method improved the average precision from 71% to 85% on the studied data set.

3.1. INTRODUCTION

Cancer caused 10 million deaths in 2020 and is ranked as the sixth leading cause of death worldwide. Moreover, approximately one in five people worldwide develop cancer during their lifetime, and by 2040, the burden of cancer is projected to increase by 47% (Organization, 2020). Despite the recent advances in this field, cancer treatment usually follows a "one-size-fits-all" approach, which leads to a good response for some patients but not for all. Many patients undergo through trial and error treatment and are subjected to needless toxicity in pursuit of the best treatment. Targeted cancer treatment is a response to this inefficiency; however, methods for predicting how a particular cancer will behave to a specific treatment are still lacking (Fior et al., 2017). In evaluating cancer treatments, the zebrafish larvae xenograft is a promising assay for identifying which therapies can lead to better results for precise and targeted medicine(Costa et al., 2020; de Almeida et al., 2020; Fior et al., 2017; Varanda et al., 2020). With this method, patients do not need to undergo various rounds of treatment based on guidelines to find the best treatment option. Instead, zebrafish larvae xenografts are used as cancer behavior sensors, representing a potential approach to assessing tumor response to treatment in just four days (Fazio et al., 2020; Fazio & Zon, 2017; Fior et al., 2017). During this process, the number of cancer cells must be quantified to measure resistance or sensitivity to anti-cancer treatments. However, the cell-counting process is a bottleneck. Due to the cells' morphology, simple pre-processing techniques do not allow for an accurate counting of the cells; instead, the process requires manual counting, which is a laborious, time-consuming, prone to human error (Alahmari et al., 2019; S. He et al., 2021). Furthermore, manual cell counting is a subjective process due to its considerable inter-observer and intra-observer variability (De Boodt et al., 2013). Consequently, automated cell quantification is a long-awaited tool among researchers, and various techniques have been created in recent years to address this problem.

With the advent of deep learning (DL) and convolutional neural networks (CNNs), several authors proposed various techniques that surpassed most traditional approaches. Automatic cell counting can be divided into two major categories: detection-based counting, which demands prior detection or segmentation, and another method based on density estimation or regression without the need for prior labeling (S. He et al., 2021; W. Xie et al., 2018).

Most cell counting tools developed through the usage of DL are based on recurring object segmentation in frameworks such as Mask R-CNN (Hadi Hosseini et al., 2020; Loh et al., 2021), U-NET (Falk et al., 2019; Yue Guo et al., 2019; Kong et al., 2020), or feature pyramid networks (Korfhage et al., 2020). However, this type of approach demands pixel-wise labeling of the ground truth, which usually is difficult to obtain.

Another well-explored category is cell counting using regression networks (S. He et al., 2021; Q. Liu et al., 2019; W. Xie et al., 2018; Zheng et al., 2020), in which the global cell count is taken as an annotation to supervise training, instead of the counting following a classification or detection framework. In this case, there is no need for prior object detection or segmentation as the ground truth. However, the spatial locations of the objects of interest are not identified.

Object detection in a cell quantification context is still poorly explored. The YOLO method has been applied to counting blood cells (Z. Jiang et al., 2021); however, this system privileges speed over accuracy (J. Huang et al., 2017), and it is not sufficiently reliable for use in the medical field. The faster regions of CNN (Faster R-CNN) method has been used on malaria images (Jane Hung et al., 2018), low-contrast microscopic images (Uka et al., 2020), and even breast cancer histopathology images (Mahmood et al., 2020), demonstrating the system's potential as an accurate cell counting method. However, Faster R-CNN has mainly been applied to analyze images with negligible cell overlapping and numbers of small objects.

To our knowledge, accurate quantification of single cells in zebrafish xenografts and zebrafish avatars is still highly demanding. Several available cell-counting plugins produce satisfactory performance in two dimensions, but they fail to obtain acceptable results in three dimensions. This is the case of zebrafish xenografts, as well as organoids or spheroids for which these existing systems are ineffective. Furthermore, automatic segmentation tools cannot differentiate objects based on more than the shape, size, or other basic cell features. At the same time, techniques such as intensity thresholds and watershed methods fail when the cell overlap is significantly high, and the nuclei of different cells are located close to one another. This limitation creates a demand for more advanced techniques, and DL systems appear to be reasonable candidates. In 2015 (B. Dong et al., 2015), a supervised max-pooling CNN was trained to detect cell pixels in regions that were preselected by a support vector machine (SVM) for automatic cell detection in zebrafish images. Although this CNN performed significantly better than the traditional approaches, the results still showed significant room for improvement, which can be addressed using more complex networks.

In this work we used a detection approach by applying the Faster R-CNN algorithm with the Inception ResNet V2 feature extractor (see Appendix 1) for cell quantification, focusing on accuracy. We conducted experiments using a data set of zebrafish xenografts provided by the Champalimaud

Foundation of Lisbon, Portugal. Due to the cell morphology in the provided images, simple pre-processing techniques do not allow for accurate cell counts, and poor labeling prevents the development of a segmentation process. Our experiments focused on optimizing the cell-quantification process. The results indicate that the employed method can achieve promising detection performance, despite the dense cell overlap, the heterogeneity of the cell morphology, and the reduced sample size.

3.1.1. Main Contributions

We present a fully automatic approach for detecting and counting cancer cells in zebrafish xenograft tumor images(see Appendix 2), with the following main contributions:

1. We demonstrate the superior capability of Faster R-CNN compared to single shot detector (SSD), You Only Look Once (YOLO) and region-based fully convolutional networks (RFCNs) as a detection and classification system for complex imaging, when accuracy is a major concern.
2. We comprehensively analyze the parameters of Faster R-CNN that can influence the ability to detect small objects, a common problem in the medical imaging research field.
3. We evaluate the effect of changing the number of proposals when dealing with overcrowded situations and cell overlapping.
4. We demonstrate the potential of data augmentation to increase a network's performance in given small data sets, which are typical of many medical applications.
5. We analyze the effect of defining accurate anchor rates and scales based on a detailed exploration of the cells, to optimize the process of detecting objects with different shapes.
6. We demonstrate our system's ability to detect cancer cells in images featuring several problems, by refining the system and adjusting it to address the problems at hand. In this way, we prove the suitability of object-detection frameworks as automatic cell-counting tools for problems in which segmentation is impossible due to inadequate labeling.

Furthermore, we present an improved version of Faster R-CNN, with precise fine-tuning that can handle common issues such as overlapping, small object size, and small data sets in medical imaging. This contribution should encourage further research, beyond the specific data used here.

3.2. PREVIOUS AND RELATED WORK

3.2.1. Object-detection algorithms

In recent years, various strategies have been used to address object-detection problems. However, comparing the performance of systems in the literature is challenging because they feature different base features extractors, image resolutions, and software and hardware platforms. Nevertheless,

these systems can be distinguished by considering the trade-off between speed and accuracy (J. Huang et al., 2017).

In this work, we focus on four recent meta-architectures that provide differing trade-offs between speed and accuracy: SSD, YOLO, Faster R-CNN, and RFCNs.

SSD (W. Liu et al., 2016b) is an architecture that uses a single feed-forward convolutional network to predict classes and anchor offsets without the need for a second classification stage, as Faster R-CNN and RFCN require. This characteristic tends to increase the system's speed while decreasing the model's accuracy (which is preferable, for instance, for video object detection). The accuracy can always be increased in each of the meta-architectures exposed by using a more robust feature extractor.

YOLO (Redmon et al., 2016) was originally published in 2016 by Redmon et. al. It was announced as an object detection framework that combines in a single network the problem of object localization and classification. YOLO treats detection as a regression problem, in which the image is divided into a grid and for each grid there is a boxing box confidence and a class probability. Known for its efficiency in real-time detection, YOLO has had several proposed improvements (Bochkovskiy et al., 2020; Redmon & Farhadi, 2017, 2018) over the years. The most recent improvement, YOLO v5, developed by Ultralytics (Jocher et al., 2021) uses a cross stage partial network (CSPNet) (C. Y. Wang et al., 2020) as the model backbone and path aggregation network (PANet) (S. Liu et al., 2018) as the neck for feature aggregation.

Faster R-CNN (Ren et al., 2017) was developed based on the architecture of the Fast R-CNN method (R. Girshick, 2015b), which, in turn, was based on the regions of CNNs (R-CNN) method (R. Girshick et al., 2014). Faster R-CNN comprises two networks: the region-proposal network (RPN), whose primary purpose is to generate a set of proposed regions where objects could be present, and a network that uses the first network's output to detect objects in those regions.

Finally, RFCN is an approach that should be faster than Faster R-CNN because the former is fully convolutional, with almost all computation shared on the entire image. RFCN uses position-sensitive score maps to address the problem of translation-invariance in image classification and translation-variance in object detection (Dai et al., 2016b).

In parallel with the definition of the object-detection system used, feature extractors also can be chosen. Depending on the problem's complexity, various feature extractors should be tested and have their performance compared.

Residual networks (ResNets), were introduced with the ResNet50 architecture (K. He et al., 2016). This type of network uses so-called skipping connections, also called residual blocks, in which some activations from one layer are fed into a deeper layer. This allows the number of layers in the network to be increased efficiently (Sewak et al., 2018).

GoogLeNet/Inception (Szegedy et al., 2015) introduced the concept of inception modules to reduce the number of parameters, even with 22 layers of depth. The network comprises nine inception

modules, with a total of 100 layers. The inception modules created micro-architectures within the network's macro-architecture, where operations happen in parallel, and filters were applied to the output to reduce dimensionality (Shanmugamani, 2018).

The Inception ResNet (Szegedy et al., 2017) is based on the Inception architecture and introduces residual blocks to the architecture. Szegedy et al. found that Inception ResNet models can achieve higher accuracy values at lower epochs.

The NASNet system (Zoph et al., 2018) was developed to optimize convolutional architectures. Inspired by the neural architecture search (NAS) framework, Zoph et al. (Zoph et al., 2018) developed a system for "searching" the space of network architectures on a small proxy data set (CIFAR-10). Then, the learned architecture was transferred to a larger data set (IMAGENET), using concepts such as architecture scalability and flexibility. The system outperformed a set of human-designed models.

Most state-of-the-art detectors perform well on challenging data sets such as COCO or PASCAL VOC since these data sets typically contain objects taking medium or large parts of an image, and the number of samples is significant. However, most of them struggle to detect overlapping and/or small objects in data sets of small size. In crowded scenes, objects tend to overlap largely with each other, leading to occlusions, and detection boxes with high overlaps can match the same object. In such a situation, it is often appropriate to apply strategies such as, for instance, Non-Max suppression, where the most appropriate bounding box for the object is selected. Additionally, small objects can be difficult to detect due to their low resolution. Moreover, existing deep neural networks lose the features of objects after several convolutions and pooling operations(G. X. Hu et al., 2018). The approaches to address those problems are typically meta-architecture dependent, since the strategies and hyperparameters used can significantly change from a type of architecture to another. The following subsections refer to some of the used techniques to address this problem by employing Faster R-CNN, the algorithm used in this study.

3.2.2. Detection of small objects

In recent years, several authors have proposed various solutions to optimize small-object detection in images. Hu et al. (G. X. Hu et al., 2018) extracted image features from their third, fourth, and fifth convolutions and merged those into a one-dimensional vector. Some methods for detecting small objects were suggested using Faster R-CNN or SSD as the background (Guan & Zhu, 2017b).

Eggert et al. (Eggert et al., 2017) presented an improved scheme for generating anchor suggestions and they proposed a modified Faster R-CNN that used higher-resolution feature maps for small objects. In (C. Cao et al., 2019), an improved loss function was proposed based on the intersection over union (IoU) for bounding box regression, and bilinear interpolation was used to improve the pooling operation for regions of interest (RoIs), to solve positioning error. In the detection phase, the authors used multiscale convolution feature fusion so that the feature map would contain more information and to improve the non-maximum suppression (NMS) technique in order to avoid the loss of overlapping objects.

Fu et al. (Fu et al., 2017) added deconvolution layers to SSD+Residual-101 to introduce additional large-scale context into object recognition and improve accuracy, especially for small objects. Cao et al. (G. Cao et al., 2018) proposed a multi-level feature-fusion method for introducing context information into SSD to improve accuracy for small objects. In (Cui et al., 2020), several high-level feature maps at various scales were extrapolated simultaneously to increase the spatial resolution. Tong et al. (Tong et al., 2020) delved deeper into handling small objects in computer vision. They suggested five perspectives of possible future research directions: emerging small-object-detection data sets and benchmarks, multi-task joint learning and optimization, information transmission, weakly supervised small -object -detection methods, and frameworks for small-object detection. Moreover, the atrous rate is a mechanism that increases the model's performance when detecting small objects (Guan & Zhu, 2017a). This rate is applied to the tensor associated with the features to crop in the first stage in order to obtain box predictions.

3.2.3. Detection in crowded and overlapping scenes

Faster R-CNN outperforms Fast R-CNN by using an RPN with a CNN model, whose primary purpose is to propose a set of regions where objects could be present. By default, Faster R-CNN assumes 2,000 proposals, which are then reduced to a small number of proposals, based on the number of relevant objects detected in the first stage and reshaped to a fixed size in a process called RoI pooling (Ren et al., 2017). Because the original version of Faster R-CNN was designed for images with a relatively small number of relevant objects, the network must be adjusted to the new context, in which the number of cells in the image can be relatively high, similar to what is applied for detecting humans in crowded scenarios (K. Zhang et al., 2019). This change demands higher precision in adjusting the NMS threshold, which is responsible for selecting the most appropriate bounding box for the detected object, because a higher value than needed can lead to more false positives, and a lower threshold can miss the detection of possibly relevant objects.

3.2.4. Small data sets

Data augmentation is the process whereby new instances are created from existing ones, usually on the fly, to avoid waste of storage. Several techniques can be applied, such as rotation, zoom, change in channel colors, and change of saturation.

These approaches introduce noise into the training process and force the model to be more tolerant of possible changes to the images, including position, orientation, and size, making the model more robust and mitigating overfitting. This process can be useful when dealing with small data sets, a common problem when analyzing medical images. In fact, it artificially boosts the size of the training set (Shorten & Khoshgoftaar, 2019). Several techniques were applied in this work:

a) **Random horizontal flip**—The image and detections were flipped horizontally. This occurred randomly with a 50% probability.
b) **Random vertical flip**—The image and the ground truth annotations were flipped vertically. This occurred randomly with a 50% probability.

c) **Pixel value scale**—The values of all pixels in the image were scaled randomly by a constant value between fixed minimum and maximum values.
d) **Random Image scale**—Images were enlarged or shrunk randomly, while keeping the same aspect ratio.
e) **RGB to gray**—Entire images were converted to grayscale randomly, with a 10% probability.
f) **Adjust brightness**—The image brightness was changed randomly, up to a maximum prefixed threshold. The image outputs were saturated at values between 0 and 1.
g) **Adjust contrast**—The contrast was scaled randomly by a value between fixed minimum and maximum values.
h) **Adjust saturation**—The saturation was altered randomly by a value between fixed minimum and maximum values.
i) **Distort color**—A random color distortion was performed.
j) **Jitter boxes**—The corners of the boxes in the images were jittered randomly, as determined by a ratio. For instance, if a box was [100, 200] and the ratio was 0.02, the corners could move by [1, 4].
k) **Crop image**—The images and bounding boxes were cropped randomly.
l) **Crop to aspect ratio**—The images were cropped randomly to a given aspect ratio.
m) **Black patches**—Black square patches were randomly added to an image.
n) **Rotation 90**—The image and detections were rotated randomly by 90° counter-clockwise 50% of the time.

3.2.5. Dealing with objects of different shapes

The original version of Faster R-CNN, by default, generated anchor boxes with three aspect ratios (1:1, 1:2, 2:1) and three scales (128 × 128, 256 × 256, 512 × 512), resulting in nine anchor boxes. The anchors can be adjusted to the problem at hand, considering different objects have different shapes, which will affect the algorithm's performance in the detection stage, as presented in (Y. Wang et al., 2019), where the anchors were adjusted to detect cars. In the present work, this parameter can be changed within Faster R-CNN to deal with the morphology of the studied cells.

3.3. MATERIALS AND METHODS

3.3.1. Data

The data we used in this study included 97 RGB images of zebrafish xenografted tumors, acquired with the initial purpose of evaluating targeted cancer treatment and assessing tumor response to various therapies. For this specific study, the images were labeled manually using the Fiji software (Schindelin et al., 2012). The labeling was performed manually by a domain expert at the Champalimaud Foundation, who applied a single dot over each cell.

Among the images, 89 had dimensions of 512 × 512 pixels, and the remaining eight had dimensions of 1280 × 1280 pixels, all with three channels, as a result of immunofluorescence technique, used to label nuclei, human cells and apoptotic or immune cells. These eight images were scaled to 512 × 512 pixels, using the inter-area technique, which is considered the best for decreasing the dimensions of the

images (Gonzalez & Woods, 2018). Because the number of cells varied significantly across the images (ranging between 18 to 661), the images were divided into three sections based on the cells' complexity and density:

- **Low**—Images with fewer than 50 cells;
- **Medium**—Images with cells numbering in the range [250]; and
- **High**—Images with greater than 250 cells.

Figure 3.1. shows the distribution of the numbers of cells in each image.

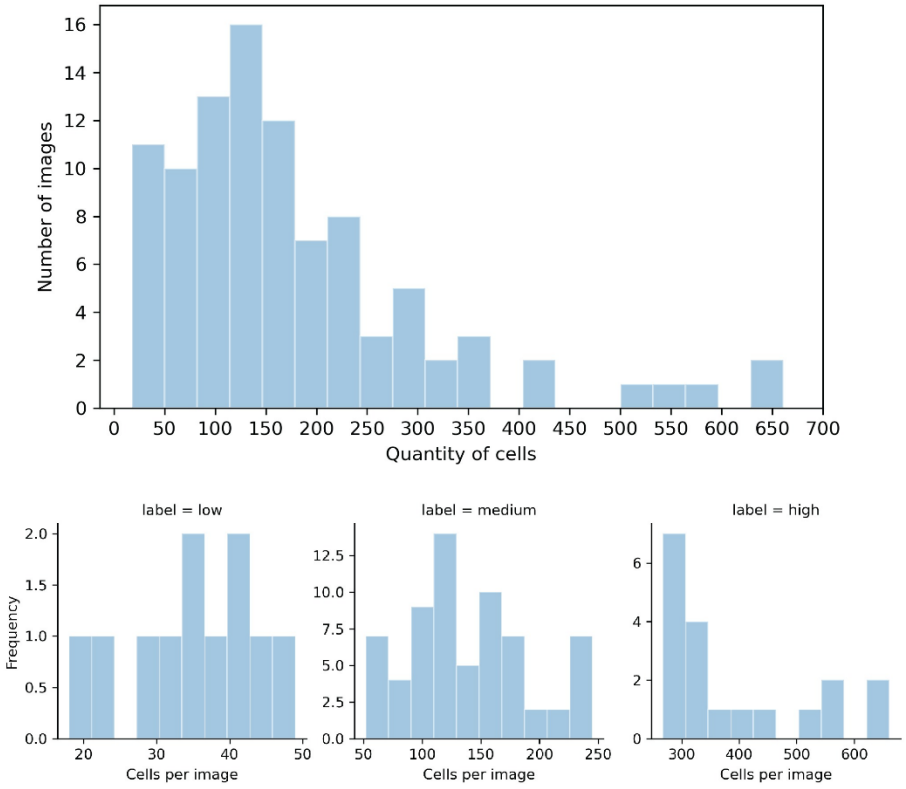

Figure 3.1 – Distribution of the number of cells in the considered images.

Note. The top histogram shows the distribution for the whole data set, and the bottom histograms show the distributions for the low, medium, and high levels of complexity. In a range of 18 to 661 cells, the majority of images had at most 250 cells. Only eleven images had fewer than 50 cells, and nineteen images had more than 250.

The majority of the images had between 50 and 300 cells. Additional descriptive statistics regarding the cell numbers in the raw images are given in Table 3.1.

Table 3.1 – Descriptive summary (mean, range, percentiles, skewness and kurtosis) of the cell quantities in the provided images, for the whole data set and for the various complexity levels.

Images (count)	Mean ± SD	Range [min, max]	25%	50%	75%	Skew	Kurtosis
All (96)	176.6 ± 132.4	[18, 661]	95	137	230	1.7	3.3
Low (11)	35.2 ± 9.1	[18, 49]	31	35	42	-0.5	-0.2
Medium (66)	138.9 ± 52.5	[52, 245]	104	129	173	0.4	-0.6
High (19)	391.2 ± 131	[267, 661]	295	343	471	1.0	-0.3

Note. On average, we had approximately 176 cells per image, but the standard deviation was 132, highlighting the need to define different groups of images according to the number of cells. Defining different groups causes the standard deviation of each group will decrease, as well as its skewness and kurtosis to decrease. Having high values for such elements can lead to underestimation for some predictions.

High-skewness situations are assumed when the value of skewness is less than −1 or greater than 1 (Bulmer, 1979). Similar to skewness, high kurtosis values are less than −2 or greater than 2. As shown in Table 3.2, the whole data set has a skewness of 1.7, and we also had a leptokurtic (kurtosis higher than 3) distribution, which can lead to situations in which the predictive model underestimates some predictions. With that in mind, the data set was partitioned into three sets (low, medium, and high) to reduce skewness and kurtosis for each sample.

Table 3.2 – Descriptive summary (mean, range, percentiles, skewness and kurtosis) of the cell dimensions (width, height and ratio).

Dimension	Mean ± SD	Range [min, max]	25%	50%	75%	Skew	Kurtosis
Width	15.9 ± 4.5	[4, 44]	13	15	19	0.8	1.1
Height	15.2 ± 4.3	[3, 45]	12	15	18	0.9	2.8
Ratio	1 ± 0.3	[0.3, 3.3]	0.8	0.9	1.1	18.9	0.1

Note. On average, the cells had a width of 15.9 pixels, a height of 15.2 pixels, and a ratio of 1. Nevertheless, their width ranged from 4 to 44 pixels, their height ranged from 3 to 45 pixels, and their ratio ranged from 0.3 to 3.3, demonstrating the cells' irregular morphology in terms of size and shape. The ratio is calculated by dividing width by height.

Taking into account the small number of samples available, the fine-tuning of the model considered the entire data set, but the regression measures calculated took data set's partitioning into account.

Besides the number of cells present in each image, it was important to establish aspect ratios and scales when generating anchors to evaluate the distributions of the width, height, and aspect ratios of the 17,128 cells present in the images. The statistical measures reported in Table 3.2, and displayed in Figure 3.2, refer to images with dimensions 512 x 512 pixels. If the image is resized when fine-tuning the model, the scaling of these values should be considered.

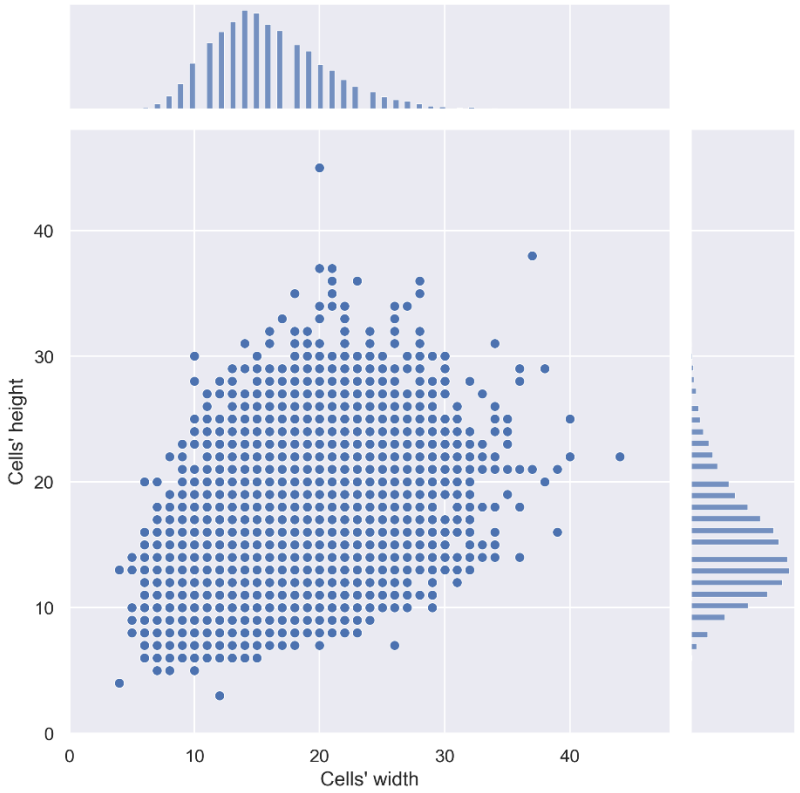

Figure 3.2 – Joint and marginal distribution plot representing the density and distribution of the cells according to their width and height.

Note. Although it is normally distributed, we can verify the significant variance of cell morphology available in the data set.

As shown in Figure 3.3, the cells' aspect ratio ranged between 0.5 and 2.0. One possible approach to selecting aspect ratios for the anchor generator was to start with three ratios (e.g., 0.5, 1.0, and 2.0) that cover the majority of the distribution, as in Figure 3.4.

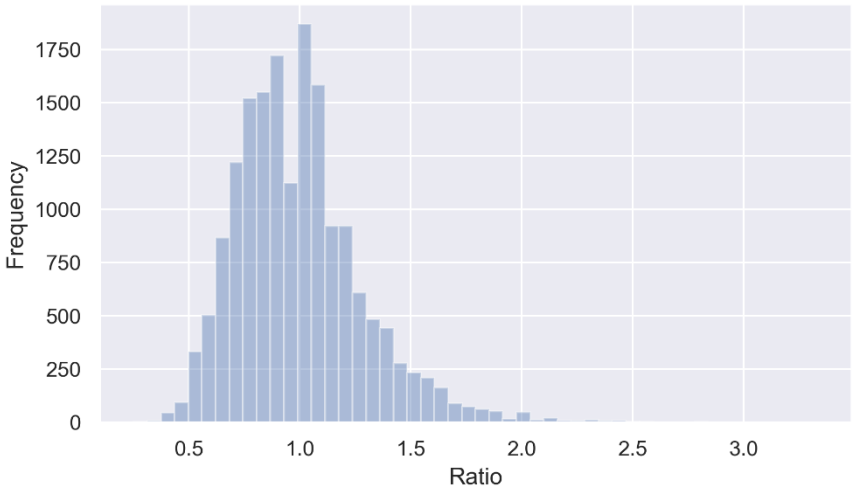

Figure 3.3 – Histogram plot for the frequency of cells according to their ratio (width/height).

Note. The cells' ratio, which can influence the aspect ratio definition in Faster R-CNN, varied between 0.3 and 3.3.

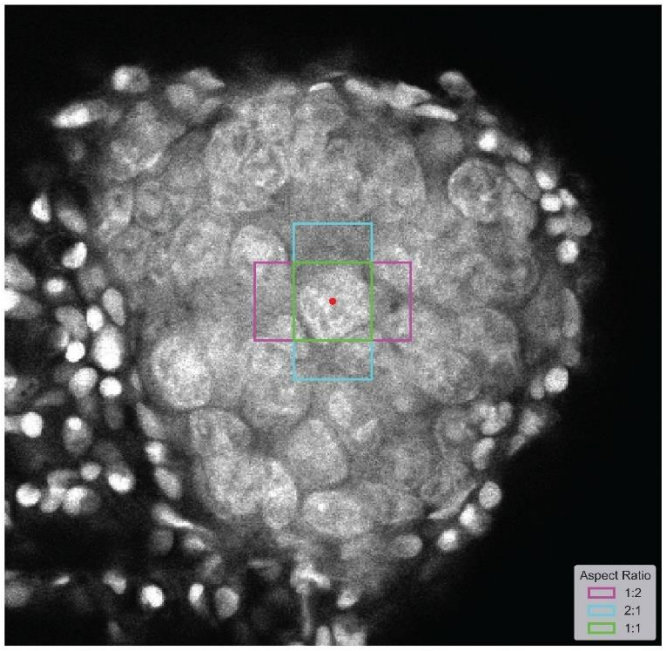

Figure 3.4 – Example of possible aspect ratios, taking into account the ratio distribution.
Note. The magenta shape corresponds to a 1:2 aspect ratio, the blue corresponds to a 2:1 aspect ratio, and the green corresponds to a square. The red dot is the anchor.

3.3.1.1. Data set splitting and labeling

To divide the initial data set, we considered not only the number of images but also the number of available cells or objects shown in the various images and the average number of cells in each image for the data sets, as this value could range between 18 and 661 cells.

The initial data set was split into three data sets, taking into account the following proportions: around 80% for training, 15% for validation, and 5% for testing, as shown in Table 3.3. Considering the data set's small size, as well as the idea that this work aimed at tuning a network to be able to deal with the several limitations available in the data, we decided to add a more considerable proportion of data to the validation set and a smaller proportion to the test set, the latter of which was used only after we selected the best model.

The labeling of the images in the data set was inadequate for segmentation. Each cell was annotated by a dot, which did not correspond exactly to the center of the cell. To overcome this problem and apply object detection, we performed additional labeling considering those initial annotations using the labelImg software. This software allows bounding boxes to be created around objects to extract the coordinates of those boxes in the image.

Table 3.3 – Data set partitioning into training, validation and test data sets, considering the quantity of cells and the quantity of images.

Data set	# cells	% cells	# images	% images	Avg. # cells per image
Train	13790	80.5	79	81.3	176.8
Validation	2292	13.4	13	13.5	176.3
Test	1046	6.1	5	5.2	209.2
Total	17128	100	97	100	176.6

Note. The data was partitioned into training, validation and test data sets. This partition took into consideration the number of cells, the number of images, and the average quantity of cells per image. We used around 80% of the data to train the model, 15% for validation purposes, and 5% for testing.

3.3.1.2. Loss functions

The loss function in Faster R-CNN is evaluated during the algorithm's two stages. At each stage, the error is measured using a multi-loss function, which sums up to four losses in total. The first stage of the algorithm, the RPN, measures the model's performance, taking into account objectness loss and localization loss, which result in the multi-loss function defined in Equation 3.1:

$$L(\{p_i\},\{t_i\}) = \frac{1}{N_{cls}}\sum_i L_{cls}(p_i, p_i^*) + \lambda\frac{1}{N_{reg}}\sum_i p_i^* L_{reg}(t_i, t_i^*) \quad (3.1)$$

Thus, the RPN takes an image (of any size) as its input and outputs a set of rectangular object proposals, each with an objectness score (Ren et al., 2017). In Equation 3.1, p stands for the predicted objectness class, p^* for the ground truth objectness class, t for the ground- truth bounding box, and t^* for the predicted bounding box.

Objectness loss classifies a patch as having, or lacking an object. In this phase, the goal is not to identify the class of the identified object but to determine whether a patch contains an object or the background. This objectness score is used to filter the bad predictions in the second stage. With this purpose in mind, Faster R-CNN uses a classifier with two possible classes: one for the presence of an object and one for the background.

Localization loss, in which a regression is applied to the bounding box, considers the parametrization of the four following coordinates, as shown in Equation 3.2:

$$t_x = \frac{(x - x_a)}{w_a} \qquad t_y = \frac{(y - y_a)}{h_a}$$

$$t_w = \frac{w}{w_a} \qquad t_h = log\frac{h}{h_a}$$

$$t_x^* = \frac{(x^* - x_a)}{w_a} \qquad t_y^* = \frac{(y^* - y_a)}{h_a} \tag{3.2}$$

$$t_w^* = log\frac{w^*}{w_a} \qquad t_h^* = log\frac{h^*}{h_a}$$

where *x*, *y*, *w*, and *h* denote the box's center coordinates and its width and height. The variables *x*, *xa*, and *x** denote the predicted box, anchor box, and ground-truth box respectively (likewise for *y*, *w*, and *h*).

For the second stage of the algorithm, two new losses are computed: the box-predictor localization loss and the box-predictor classification loss. Each RoI is labeled with a ground truth class *u* and a ground-truth bounding-box regression target *v*. The multi-loss function is defined as seen in Equation 3.3:

$$L(p, u, t^u, v) = L_{cls}(p, u) + \lambda[u \geq 1]L_{loc}(t^u, v) \tag{3.3}$$

where *p* stands for the discrete probability distribution over the categories and t^u is the bounding-box regression achieved from the first stage using the RPN localization loss. The indicator function [$u \geq 1$] evaluates to 1 when u ≥ 1 and 0 otherwise (the background is labeled as 0). The hyperparameter λ controls the balance between the two losses (R. Girshick, 2015b).

Box-predictor localization loss. Ren et al. (Ren et al., 2015) applied a smooth-L1 loss on the position (x,y) of the top-left of the bounding box and the logarithm of the heights and widths, similar to the RPN localization loss applied on the first stage, as shown in Equation 3.4:

$$L_{loc}(t^u, v) = \sum_{i \in x,y,w,h} smooth_{L_1}(t_i^u - v_i) \tag{3.4}$$

where

$$smooth_{L_1}(x) = \begin{cases} 0.5x^2 \; if \; |x| < 1 \\ |x| - 0.5 \; otherwise \end{cases}$$

where *x* is the difference between the predictions and target.

Choosing a smooth L1 loss comes from the fact that this type of loss is less sensitive to outliers and thus is more robust. When the regression targets are unbounded, training with L2 loss can demand careful tuning of the learning rates to avoid exploding gradients (R. Girshick, 2015b).

This loss is aimed at refining the localization of the bounding boxes before classification.

Box-predictor classification loss. By default, weighted sigmoid was used to quantify classification loss. However, focal sigmoid classification loss and bootstrapped sigmoid classification loss were also tested.

As a binary classification, all former classification losses were based on the cross-entropy loss for binary classification (T. Y. Lin et al., 2020), as presented in Equation 3.5:

$$CE(p, y) = \begin{cases} -\log(p) \; if \; y = 1 \\ -\log(1 - p) \; otherwise \end{cases} \tag{3.5}$$

In Equation 3.5, $y \in \{\pm 1\}$ specifies the ground truth class and $p \in [0, 1]$ is the model's estimated probability for the class with label $y = 1$. For notational convenience, *pt* is defined as follows in Equation 3.6:

$$p_t = \begin{cases} p \; if \; y = 1 \\ 1 - p \; otherwise \end{cases} \tag{3.6}$$

and

$$CE(p, y) = CE(p_t) = -\log(p_t)$$

3.3.1.3. Evaluation metrics

Mean average precision (mAP) is the standard metric used to evaluate an object-detection algorithm. This metric is the product of the precision and recall of the detected bounding boxes and ranges from 0 to 1, with higher values indicating better performance. The mAP, which is based on the concept of IoU, is a good measure of the network's sensitivity (Shanmugamani, 2018). The IoU is the ratio of the overlapping area between the ground truth and predicted area to the total area or union area.

In addition to the mAP metric, used to fine-tune the model, and taking into account the problem in hand, other metrics were considered to evaluate the model's performance. Although mAP is the standard metric used in object-detection systems, it has limitations, especially when dealing with a large number of objects. This happens because if a specific object is not identified on a region proposal at the first stage of the algorithm, it will not be taken into account for the model's accuracy. In this way, each model's performance in the validation data set was tested, and several metrics were applied to measure the regression performance between the predicted value, (i.e., the number of objects predicted) and the ground-truth value. The root mean squared error (RMSE), mean absolute error (MAE), and median absolute error (MedAE) were taken into account. In the end, due to the nature of the problem, our best models should account for not only the mAP, but also the regression metrics. With that purpose, six champion models were selected and trained for additional epochs. The model that provided the best performance was considered the winner.

3.3.1.4. Implementation details

The experiments were implemented using the TensorFlow Object Detection API, an open-source framework built on top of TensorFlow. All tests described in the following sections were carried out using Azure cloud computing resources, namely four Tesla K80 GPUs (2 physical cards) with 32 GiB GPU memory, on a virtual machine with 24 vCPUs, 224 GiB of memory, and a 1.44 TB SSD as temporary storage.

3.4. RESULTS AND DISCUSSION

The research conducted in (Albuquerque, C., 2019) yielded the following results pertaining to hyperparameter fine-tuning. An additional algorithm was tested as benchmark, and these findings were organized in this work based on the primary constraints inherent in the explored dataset. These constraints include the presence of small objects, overcrowded scenes, a limited dataset, and the existence of objects with diverse shapes and sizes.

3.4.1. Performance comparison of different object-detection systems

In this experiment, we tested the performance of different feature extractors and meta-architectures using a model pretrained on the COCO data set (T.-Y. Lin et al., 2014). Each model was run for 4,000 steps (200 epochs) with images resized to 600 x 600 pixels, the default value in TensorFlow API.

3.4.1.1. Choosing the meta-architecture

During the initial stage, our concern was defining the best meta-architecture for the problem at hand.

As shown in Figure 3.5, we created four models using RFCN, SSD, YOLO and Faster R-CNN. The R-CNN model showed a clear advantage over the others. When comparing SSD and Faster R-CNN, and using Inception V2 as a feature extractor, Faster R-CNN achieved an mAP value of 0.34 after 4,000 steps, whereas SSD was unable to exceed 0.04 mAP. When comparing RFCN and Faster R-CNN, this time with ResNet101 as a feature extractor, RFCN achieved an mAP of 0.31 after 4,000 steps, whereas Faster R-CNN achieved a value of 0.53 mAP. When implementing YOLOv5, provided by Ultralytics (Jocher et al., 2021; Solawetz & Nelson, 2020), we achieved a value of 0.43 mAP at 4,000 steps. This experiment shows the suitability of Faster R-CNN for the problem at hand, compared to the other meta-architectures tested.

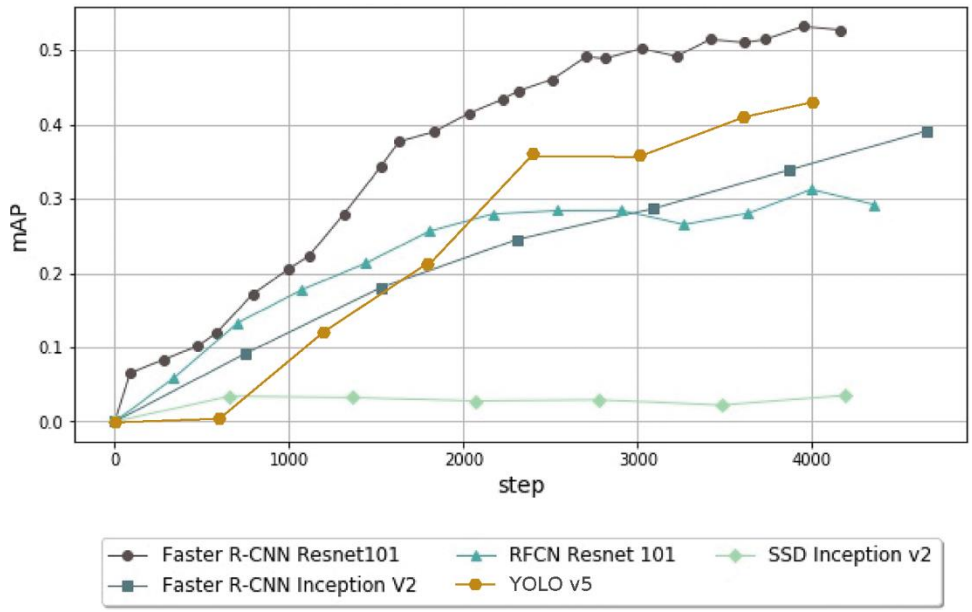

Figure 3.5 – Performance comparison using the mAP of the object-detection models trained using various meta-architectures (Faster R-CNN, SSD, YOLO and RFCN) for 4,000 steps.

Note. Faster algorithms such as SSD cannot deal with the complexity of the problem. Faster R-CNN emphasizes accuracy over speed and can achieve over six times better performance than SSD with the same feature extractor. YOLO v5, the last version of YOLO, outperforms SSD and RFCN, but Faster R-CNN still has an advantage of 0.1 mAP at 4,000 steps.

3.4.1.2. Defining the best feature extractor

To better understand the influence of the feature extractor, we applied five feature extractors to Faster R-CNN: Inception V2, NAS, ResNet50, ResNet101, and Inception ResNet V2.

Table 3.4 compares the mAP values at the validation data set and the speed of the feature extractors when training the model. At 4,000 steps, when Faster R-CNN achieves convergence, Inception ResNet

V2 performed the best, achieving a value of 0.71 mAP. As expected, increasing the feature extractor complexity increases training speed, and when using NAS, this value reaches 96 seconds per epoch, even if the model's performance is not the highest, with an mAP of 0.44.

Table 3.4 – Validation performance and training speed for Inception V2, NAS, ResNet50, ResNet101 and Inception ResNet V2 on Faster R-CNN.

Meta-Architecture	Feature Extractor	mAP at 4,000 steps	Speed/epoch (sec)
Faster R-CNN	Inception V2	0.34	6.17
Faster R-CNN	NAS	0.44	93.36
Faster R-CNN	ResNet50	0.50	10.34
Faster R-CNN	ResNet101	0.53	47.08
Faster R-CNN	Inception ResNet V2	0.71	45.55

Note. This table presents mAP values at 4,000 steps and time to train an epoch in seconds using Faster R-CNN with different feature extractors (Inception V2, NAS, ResNet50, ResNet101, and Inception ResNet V2). Changing the feature extractor and, consequently, the network complexity, can lead to significant changes in mAP, ranging from 0.34 on Faster R-CNN with Inception V2 to 0.71 on Faster R-CNN with Inception Resnet V2. Increasing the complexity also affects training speed. Clear evidence of this fact can be seen when changing the network from ResNet50, which is 50 layers deep, to ResNet101, which has 101 layers. The mAP increases slightly, but training takes more than four time longer.

These results demonstrate that Faster R-CNN can achieve a performance of around 18 times better when upgrading the feature extractor from Inception V2 to Inception ResNet V2, thus demonstrating the advantage of using the latter when accuracy is the primary goal.

3.4.2. Performance evaluation for detecting small objects

The purpose of the experiments that followed the testing of the best feature extractor was to increase the system's capability to detect small cells. Thus, we trained our model (Faster R-CNN with Inception ResNet V2) in various experiments by changing the size of the figure, changing the stride used in the anchor generator, using an atrous rate, and adjusting the IoU threshold. Starting from the default values at TensorFlow API (Appendix 3-5), grid search was used on those hyperparameters to maximize the model's performance without overfitting. This tuning process is clarified on the following sections.

3.4.2.1. Changing the image size

Because we are dealing with small objects and considering the known limitations of object-detection algorithms, one possible approach to dealing with this situation is to increase the size of the input images. To test the effect of image resizing on the algorithm's performance, we conducted experiments using various sizes and resizing methods.

We trained five models for 3,000 steps and changed the input images' size, where the lowest resolution was 256 × 256 and the highest was 1100 × 1100.

As a default, TensorFlow API assumes that an image should have a minimum of 600 pixels or a maximum of 1024 pixels by mantaining the aspect ratio. By dealing with squared images and knowing that an increase in the number of pixels demands a higher training time and more computational resources, we made several adjustments to understand how the image resizing can affect the algorithm's performance. At most, we resized the image to a dimension of 1100 × 1100 pixels, the maximum possible size for training the network with a batch of 1 without a RAM leak.

As Figure 3.6 indicates, the images' dimensions significantly influenced the model's mAP, probably because of the presence of small objects in the input data. This was corroborated by the results presented in Table 3.5, which indicate a decrease in the RPN localization loss and RPN objectness loss for the higher-resolution images.

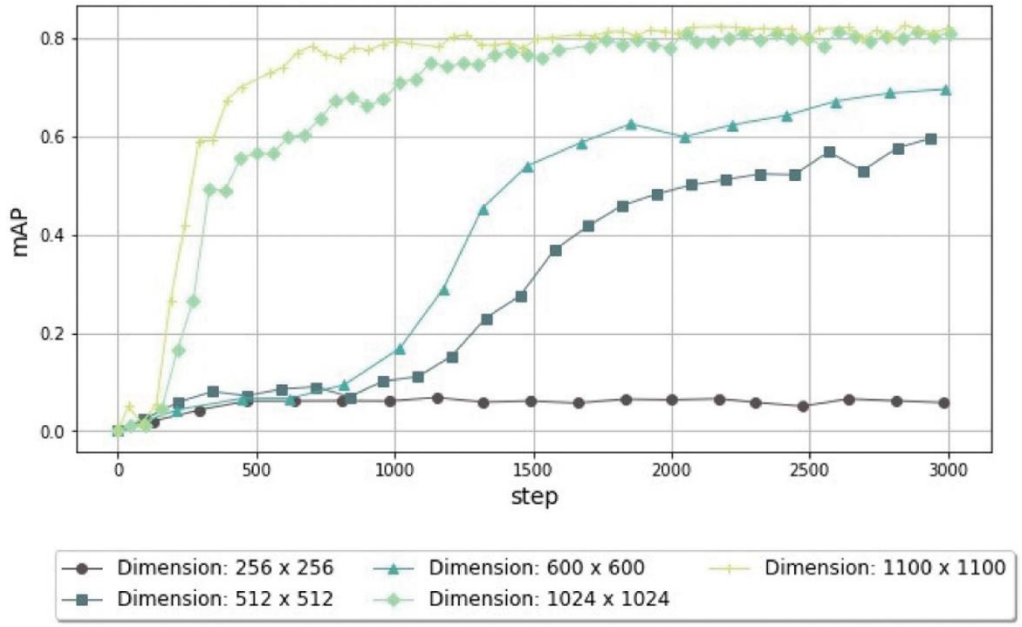

Figure 3.6 – Performance comparison using the mAP of the object-detection models trained using different image sizes (from 256 × 256 pixels to 1100 × 1100 pixels) for 3,000 steps.

Note: When dealing with very small cells with a small number of pixels, it is important to increase the resolution of the image to improve the richness of features. In this way, the detector can correctly identify and classify the cells.

Furthermore, the dimensions also affected the model's convergence: when using the original dimensions (512 × 512), the model will began to stabilize only at 3,000 steps. However, when using the maximum dimensions, a convergent state is achieved at around 1,000 steps, as shown in Figure 3.6. Moreover, it is possible to conclude that low-resolution images do not allow the model's performance to improve during training, as seen in the images with dimensions of 256 × 256. Finally, as expected, increasing the dimensions of the images also increases the time required to conclude each epoch also increases. This ranged from 28 seconds for the lowest resolution to 94 seconds for the highest one.

Table 3.5 – Validation performance and training speed for Inception V2, NAS, ResNet50, ResNet101 and Inception ResNet V2 on Faster R-CNN.

Size (pixels)	Total Loss Validation	RPN validation loss		mAP	Speed/epoch (sec)
		Loc.	Obj.		
256 x 256	3.47	2.92	0.63	0.06	28.22
512 x 512	3.19	1.82	0.29	0.57	38.80
600 x 600	2.61	1.65	0.16	0.70	45.55
1024 x 1024	1.86	0.89	0.07	0.81	83.77
1100 x 1100	2.28	0.82	0.06	0.82	94.13

Note. Total loss for validation, RPN validation loss for localization and objectness, mAP, and speed of training one epoch in seconds for 3,000 steps using different image-input dimensions (from 256 x 256 pixels to 1100 x 1100 pixels). There is a trade-off between accuracy and speed. A higher resolution will allow achieving a better mAP and a slower training time. With an increase of the image resolution from 256 pixels to 1100 pixels, the mAP rose from 0.06 to 0.82.

Before running further experiments, we adjusted the optimizer and the learning rate (LR). Whereas the Momentum optimizer was applied in the original Faster R-CNN, we tested three optimization algorithms to check their impact on the model (Figure 3.7).

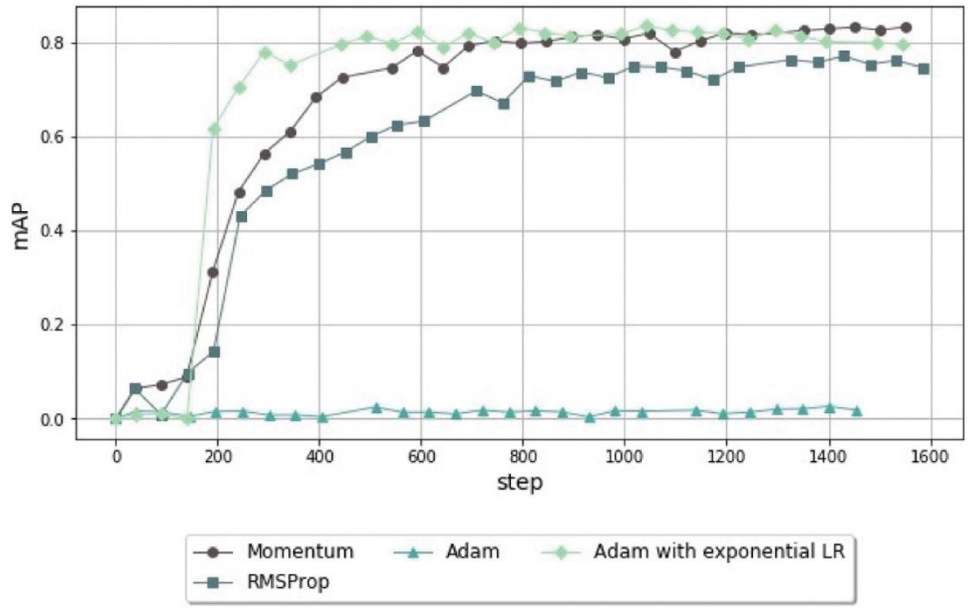

Figure 3.7 – Comparison of mAP performance of the object-detection models trained using different optimizers (Momentum, Adam, Adam with exponential learning rate, and RMSProp) for 1,400 steps.

Note. Adam, which is known for achieving a faster convergence than the remaining optimizers, cannot achieve a satisfying performance if the learning rate decay is not well adjusted.

As shown in Figure 3.7, changing the optimizer from Momentum to Adam with exponential LR allowed the model to stabilize at around 400 steps, with an approximate mAP value of 0.8, whereas Momentum required 600 steps to achieve that performance. Due to this faster convergence, Adam was selected as the optimizer in the subsequent experiments. Furthermore, with Adam as the optimizer, the initial LR was defined as 0.0002, which seemed to offer better accuracy and to boost the model's performance in earlier stages, when compared to other LR values, as shown in Figure 3.8.

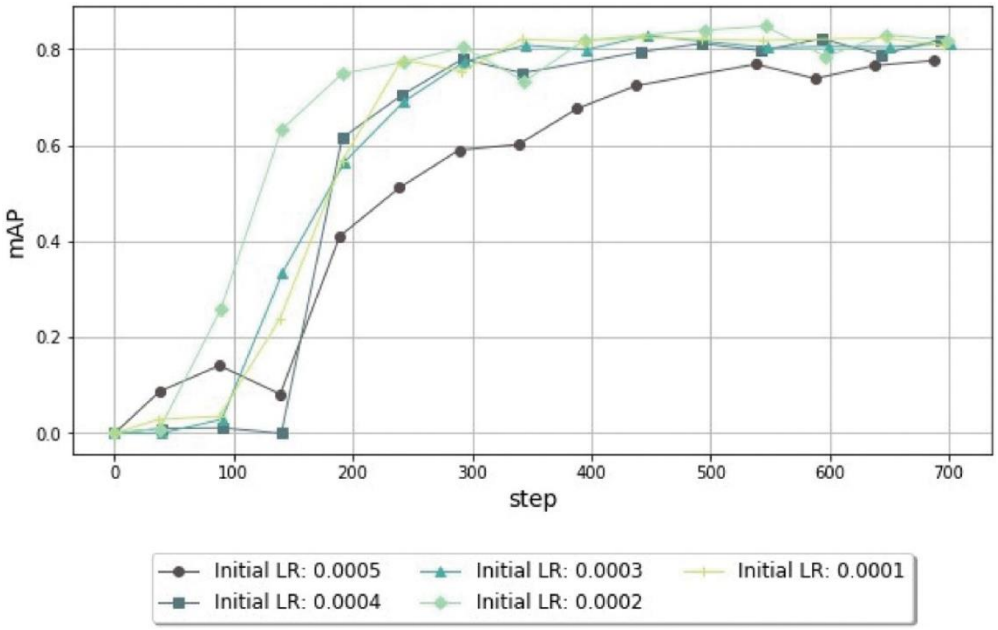

Figure 3.8 – Comparison of mAP values from the object-detection models trained using different initial learning rates (LR) for 700 steps.

Note. After a set of experiments where we test the learning rate with different orders of magnitude, the range of values between 0.0001 and 0.0005 leads to an improvement in the mAP. In this range, the value of 0.0002 results in faster convergence.

3.4.2.2. Changing the anchor generator's stride

When generating bounding boxes, different parameters should be adjusted to optimize the system's performance. One of these parameters is the stride (i.e., the space between each anchor). The TensorFlow API defines as default a value of 8 pixels, but this value can be doubled, when needed, to achieve a faster training time.

As seen in Table 3.6, because were dealing with small cells, a value of 8 pixels was adequate for the problem at hand. When defining this value as 16 pixels, an mAP of 0.8 was achieved, whereas with 8 pixels, the mAP reached a value of 0.85.

Table 3.6 – Comparison of detection results with strides of 8 and 16 pixels on the anchor generator.

Stride	mAP at step				Speed/epoch (sec)
	100	200	300	400	
8	0.67	0.82	0.82	0.85	96.77
16	0.76	0.80	0.80	0.82	46.01

Note. Performance comparison using mAP at different stride values applied on the anchor generator (8 and 16 pixels), at 100, 200, 300, and 400 steps, and the training time for one epoch in each scenario. A smaller stride might improve mAP further.

Changing the stride also affects the training time. In particular, decreasing the stride's value from 16 pixels to 8 pixels seemed to decrease the model's speed from 46 seconds to 96 seconds.

3.4.2.3. Using an atrous rate

The atrous rate allows object-detection models to detect smaller objects. In this experiment, we tested the model's performance with and without an atrous rate. As seen in Table 3.7, using an atrous rate during training provided slightly better results than not using an atrous rate did, and doing so did not affect the training time.

Table 3.7 – Comparison of detection results when using an atrous rate.

Atrous Rate	mAP at step				Speed/epoch (sec)
	100	200	300	400	
True	0.67	0.82	0.82	0.85	96.77
False	0.63	0.75	0.82	0.84	96.77

Note. Comparison of the mAP values of the models trained in two scenarios: when using an atrous rate (TRUE) and not using one (FALSE). This performance was evaluated at 100, 200, 300, and 400 steps, and the training time per epoch is shown in seconds for each context.

3.4.2.4. Changing the IoU threshold

The IoU threshold defines what bounding boxes should be considered as overlapping during the NMS technique. The chosen threshold value, or cutoff value, defines the models' level of sensitivity and specificity. The higher the value, the more specific and less sensitive the model, leading to fewer false positives and more true negatives. In contrast, a lower value will translate in higher sensitivity and lower specificity by obtaining more false positives and fewer true negatives. The threshold value, a problem-dependent parameter, was adjusted and optimized using grid search and the one that produced a better evaluation metric on the validation set was chosen.

We evaluated the model's performance under various IoU thresholds, ranging from 0.3 to 0.8, as seen in Table 3.8.

Table 3.8 – Comparison of detection results using various IoU thresholds, ranging from 0.3 to 0.8.

IoU (threshold)	Total Loss Validation	Box Validation Loss		mAP	Speed/epoch (sec)
		Loc.	Class.		
0.3	1.15	0.16	0.17	0.84 (317)	102.40
0.4	1.36	0.19	0.40	0.84 (400)	98.09
0.5	1.61	0.33	0.49	0.84 (329)	97.30
0.6	2.17	0.58	0.72	0.83 (332)	96.26
0.7	2.75	0.74	1.02	0.85 (400)	96.77
0.8	2.13	0.57	0.73	0.83 (337)	96.40

Note. Total loss for validation, box-validation loss for localization (Loc.) and classification (Class.), best mAP before 400 steps, and training time for one epoch in seconds using various IoU thresholds (ranging from 0.3 to 0.8). The change in the IoU threshold, although it did not lead to significant changes in mAP, led to the best performance with a value of 0.7.

Decreasing the IoU threshold in NMS increases the number of accepted proposals, which increases training time. However, decreasing the IoU threshold seems to affect the validation losses regarding classification and localization at the second stage.

3.4.3. Performance evaluation for overcrowded scenes

The maximum number of proposals parameter should always be, at a minimum, the maximum number of ground-truth boxes present in the input images. We tested how the number of proposals influenced the model's performance by manually adjusting this value from 1000 to 3500 proposals in steps of 500, as seen in Table 3.9. Increasing the number of proposals increased the training time.

Table 3.9 – Validation losses, mAP, and training speed comparison using different numbers of proposals in a range of 1000 to 3500.

Number of proposals	Total loss Val.	Box val. loss		mAP	Speed/epoch (sec)
		Loc.	Class.		
1000	1.67	0.39	0.40	0.83 (340)	95.49
1500	1.34	0.25	0.32	0.83 (381)	97.56
2000	1.36	0.19	0.40	0.84 (400)	98.09
2500	1.29	0.16	0.33	0.83 (400)	101.05
3000	1.15	0.11	0.25	0.85 (310)	104.34
3500	1.12	0.12	0.20	0.84 (213)	106.80

Note. Total loss for validation, box-validation loss for localization (Loc.) and classification (Class.), best mAP until 400 steps, and training time for one epoch in seconds using between 1,000 and 3,500 proposals. Increasing the number of proposals decreased the box validation loss and, consequently, improved the mAP.

Although the final mAP indicates no apparent differences between the number of proposals, the model performed better with 3,000 proposals. Thus, we can conclude that the number of proposals generated can affect the training speed without any significant effects on the model's accuracy. However, the losses associated with the second stage of the algorithm showed significant reductions when increasing the number of proposals, even though that shrinkage was not visible in the final mAP.

3.4.4. Dealing with small datasets

In this experiment, we evaluated the effectiveness of various data-augmentation techniques on the model's performance. We tested the approaches individually for 400 steps (20 epochs). Table 3.10 shows that the model's performance increased significantly when applying data-augmentation techniques. In particular, the mAP values increased by 0.3 with respect to the baseline model, which did not use data augmentation.

Table 3.10 – mAP value at steps 100, 200, 300, and 400 when using various combinations of data-augmentation techniques.

Data Augmentation	mAP at step			
	100	**200**	**300**	**400**
(00)None	0.12	0.32	0.51	0.59
(01)Horizontal Flip	0.51	0.75	0.81	0.82
(02)Vertical Flip	0.12	0.57	0.78	0.81
(03)Pixel value scale	0.40	0.76	0.78	0.81
(04)Image scale	0.72	0.79	0.80	0.80
(05) RGB to gray	0.47	0.79	0.81	0.81
(06) Adjust brightness	0.68	0.82	0.82	0.80
(07)Adjust contrast	0.01	0.02	0.29	0.76
(08)Adjust saturation	0.02	0.11	0.66	0.78
(09)Distort color	0.25	0.75	0.80	0.81
(10)Jitter boxes	0.76	0.77	0.82	0.79
(11)Crop image	0.67	0.78	0.76	0.80
(12)Crop to aspect ratio	0.36	0.80	0.82	0.81
(13)Black patches	0.02	0.40	0.75	0.80
(14)90º rotation	0.04	0.45	0.76	0.80
(15)Top4: (01, 06, 10, 12)	0.68	0.78	0.78	0.81
(16)All	0.51	0.74	0.81	0.82

Note. Performance comparison for the mAP of the models trained using distinct data-augmentation scenarios. This performance was evaluated at 100, 200, 300 and 400 steps, and the training time per epoch is shown in seconds for each context. The set of augmentations implemented should not be random. We tested various methods on the data to understand the techniques to be applied.

The results indicate that the most effective augmentation techniques for short-term training were image scaling, jitter boxes, brightness adjustment, and image cropping. The saturation adjustment proved its efficiency only after more than 300 steps of training. After checking the data-augmentation techniques individually that produced the best performance, we tested two situations: combining the techniques that gave the best results (0.82 mAP) individually, named the Top 4 (15), and testing all augmentation techniques simultaneously (16). Although the mAP of the Top 4 did not differ when compared to the best individual techniques, using a set of data-augmentation techniques can provide results that are better than or comparable to those techniques used in isolation.

3.4.5. Dealing with objects of different shapes

The grid-anchor generator can be defined by the stride, scales, and aspect ratios. The experiments that followed the evaluation of the augmentation techniques analyzed the behavior of the training according to some changes to those parameters. Notably, at any given location (i.e., at each anchor), the number of the scales times the number of the aspect ratio anchors was generated with all possible combinations of scales and aspect ratios. In this experiment, we optimized the anchor generation process in a quantifiable way to match the receptive field shapes and reduced the number of invalid anchors. Taking as the baseline the cell exploration performed in the Data section, in which the dimensions and the aspect ratio of the different cells were analyzed, we made some changes to the default values used in Faster R-CNN and evaluated how those changes affected performance during the detection stage.

3.4.5.1. Changing the anchor generator's aspect ratio

By default, Faster R-CNN considers the aspect ratio using three values (0.5, 1.0, and 2.0). In this experiment, we checked whether increasing the number of aspect ratios could also increase the model's performance. Therefore, we added the value of 1.5, considering that the cells could assume different shapes and could demand more flexible aspect ratios. As the results in Table 3.11 show, adding one more aspect ratio to the anchor generator did not improve the model's performance in the long term. Although the mAP rose faster when using this additional value, we verified that using only three aspect ratios was sufficient to detect the objects in our images. This could be explained by the fact that the bounding boxes are eventually readjusted to the ground-truth boxes using the localization losses in the first and second stages of the object detection.

Table 3.11 – Comparison of detection results for three and four aspect ratio values.

Aspect Ratios	mAP at step				Speed/epoch (sec)
	100	200	300	400	
[0.5, 1.0, 2.0]	0.51	0.74	0.81	0.82	94.99
[0.5, 1.0, 1.5, 2.0]	0.74	0.80	0.80	0.82	96.00

Note. Performance comparison by mAP of the models trained using two possible aspect ratios. This performance was evaluated at 100, 200, 300, and 400 steps, and the training time per epoch for each configuration is shown in seconds. In object detection systems, the bounding boxes in an image can

assume different shapes, reflected by the aspect ratios of those bounding boxes. An additional aspect ratio was used to create initial bounding boxes that could potentially improve the fit to the cells' morphology.

3.4.5.2. Changing the anchor generator's aspect scale

Image resizing to 1100 × 1100 pixels affects the scale value used in the anchor generator. Therefore, the values presented in Table 3.2 for the cells' width, height, distribution should undergo adjustments due to the new dimensions. Table 3.12 shows those values scaled for the new reality, as well as a descriptive statistical summary.

Table 3.12 – Descriptive summary (mean, range and percentiles) of the cells' dimensions in the scaled images.

Dimension	Mean	Std	Min	25%	50%	75%	Max
Width	34.2	9.8	9	28	32	41	95
Height	32.6	9.4	6	26	32	39	97

Note. Descriptive summary (mean; standard deviation; minimum; first, second and third quartiles; and maximum) related to the width and height of the cells present in the input images scaled to 1100 x 1100 pixels. There was a need to adjust the values from Table 3.2 due to image resizing.

To calculate the best scales to use for the anchors, and because the size basis for the anchors was 256 pixels, the values of the pixels presented on Table 3.13 were scaled to this size.

Table 3.13 – Comparison of detection results using different scales for the anchor generator.

Scale	mAP at step				Speed/epoch (sec)
	100	200	300	400	
[0.25, 0.5, 1.0, 2.0]	0.74	0.80	0.80	0.82	96.00
[0.05, 0.15, 0.3, 0.5]	0.58	0.76	0.79	0.83	97.11
[0.15, 0.3, 0.5]	0.76	0.82	0.81	0.82	95.74
[0.15, 0.3, 0.5, 1.0]	0.67	0.82	0.82	0.85	96.77
[0.3]	0.01	0.02	0.01	0.01	92.82

Note. mAP performance comparison of the models trained using five scales at the anchor generator. This performance was evaluated at 100, 200, 300, and 400 steps, and the training time per epoch is shown in seconds for each configuration. The scales were adjusted so that the cells' dimensions were taken into account.

The cells' mean width and height was around 0.13 of the basis anchor size. The minimum was around 0.03, and the maximum was 0.38. Considering these values, we executed experiments with various scales to check the behavior of the mAP at 400 steps.

The results shown in Table 3.13 indicate that limiting the scales to values similar to the cell size increases overfitting, thus leading to poorer performance. However, reducing the lowest value of the scale improves the model because the value is better adjusted to the problem's reality.

By testing a single scale (in this case, the value 0.3), we were able to corroborate that the object detection was affected profoundly. After concluding that the scales list [0.15, 0.3, 0.5, 1.0] achieved the best results in the previous experiments, we re-evaluated the mAP using the default list of aspect ratios to determine the best configuration to use in continuing the experiments. By combining the information on scales and aspect ratio, we confirmed, as shown in Table 3.14, that combination of the aspect ratios [0.5, 1.0, 1.5, 2.0] and scales [0.15, 0.3, 0.5, 1.0] resulted in the best mAP. Following this configuration, each anchor defines 16 bounding boxes.

Table 3.14 – Comparison of detection results with various aspect ratios and the scale [0.15, 0.3, 0.5, 1.0].

Aspect Ratios	**mAP at step**				**Max mAP (step)**
	100	**200**	**300**	**400**	
[0.5, 1.0, 2.0]	0.73	0.76	0.80	0.81	0.82 (238)
[0.5, 1.0, 1.5, 2.0]	0.67	0.82	0.82	0.85	0.85 (400)

Note. Performance comparison by mAP of the models trained using various aspect ratios with the scale [0.15, 0.3, 0.5, 1.0]. This performance was measured at 100,200,300,and 400 steps, and we identified the step at which the modelsachieved the best mAP.

3.4.6. Evaluating the models' regression metrics

To select the models in the previous steps, we mainly considered the mAP obtained after training. To evaluate the models further, we evaluated several regression metrics, including the RMSE, MAE, and MEDAE, on the data set with medium complexity. The results are shown in the Tables in the Appendixes 3 to 5.

After evaluating the results, we selected the six best models and executed each for 2,000 steps to check their performance. The six models were as follows:

- **Fine-tuned (A)**—Takes into account the model that was fine-tuned to balance mAP and speed over several adjustments. It is represented in blue, and it achieved an mAP of 0.8457.
- **Best regression (B)**—Takes into account the model that returned the best average regression metrics. It is represented in yellow, and it achieved an mAP of 0.7705.
- **Mixed regression and fine-tuned (C)**—As a mix between the two previous options, the best regression metric results were chosen only if the mAP did not change by more than 0.01 from the best mAP obtained with the present configuration. It is represented in green, and it achieved an mAP of 0.8300.

- **Best mAP (D)**—The model that obtained the highest mAP during the experiments. It is represented in magenta and achieved an mAP of 0.8558.
- **Best RMSE&MAE (E)**—The model that obtained the highest RMSE and MAE during the experiments. It is represented in orange and had an mAP of 0.8293.
- **Best MEDAE (F)**—The model that obtained the highest MEDAE during the experiments. It is represented in gray, and it had an mAP of 0.8426.

As Table 3.15 shows, the model with the highest mAP is not necessarily the one with the best regression results.

Table 3.15 – Performance comparison of the six best models for the medium level of complexity and for the entire data set

Model	Medium Level			All Levels		
	RMSE	**MAE**	**MEDAE**	**RMSE**	**MAE**	**MEDAE**
A	9.8	9.0	9.5	64.5	30.8	10.0
B	122.0	112.1	99.5	223.4	160.4	100.0
C	18.7	15.4	13.5	81.0	41.2	15.0
D	11.2	9.4	10.0	45.1	24.8	11.0
E	8.8	6.4	3.5	52.7	23.7	7.0
F	10.0	7.5	6.0	24.8	17.4	13.0

Note. Performance comparison of the six final models using regression metrics, specifically the root mean squared error (RMSE), mean absolute error (MAE), and median absolute error (MEDAE). This evaluation was conducted using the entire data set (all levels) and those images considered to have medium complexity (medium level).

Upon noticing the different effects of image complexity on the final architectures, we defined three models for each level and adjusted the score threshold.

For the low-complexity images, we chose the best MEDAE model and defined the score threshold as 0.9. For the medium level, we selected the best RMSE & MAE model, and a threshold of 0.4. Finally, for the high-complexity images, we selected the best RMSE & MAE model, and a threshold of 0.2.

Finally, descriptive statistics were inferred from the final results, as shown in Table 3.16, and conclusions were made.

Table 3.16 – The range, mean, and percentiles of error associated with the various complexity levels.

Level	[Min, Max]	Mean ± STD	25%	50%	75%
Low	[0,10]	5.86 ± 3.58	3	6	10
Medium	[1,45]	14.37 ± 13.12	3.25	6	22.5
High	[1,79]	35.00 ± 32.13	7	22	68
All	[0,79]	16.53 ± 20.96	3.75	6.5	22.25

Note. Descriptive summary statistics (minimum; maximum; mean; standard deviation; and first, second, and third quartiles) describing the error encountered in the subsets of images defined by the complexity level.

The results were as follows:

- For the low level, the error found in samples was in the range of [0, 10] units, with a standard deviation of 3.58. Half of the samples had fewer than 6 units of error.
- For the medium level, the error range was [0, 45], with a standard deviation of 13.12. Half of the samples had fewer than 6 units of error, and 75% had fewer than 22.5 units of error.
- For the high level, the error range was [1, 79], with a standard deviation of 32.13. Half of the samples had fewer than 22 units of error.
- Overall, the samples' error range was [0, 79], with a standard deviation of 20.96. Half of the predicted values had an absolute difference from the real values of lower than 6.5 units of error.
- Figures in Appendixes 6 to 8 present the inference of the three levels as examples, alongside the originally labeled ones.

3.5. CONCLUSION

This paper proposes a refined version of Faster R-CNN (Appendix 9) for automatic counting of cancer cells with a promising mAP.

As an extension of the work carried out in (Albuquerque, C., 2019), we presented a new comprehensive literature review that highlights the enduring significance of the concerns addressed in the former work. Moreover, we incorporated YOLOv5, a more recent algorithm, as an additional benchmark to compare against the frameworks proposed in (Albuquerque, C., 2019).

Finally, the obtained results were meticulously structured, taking into account the various constraints commonly encountered in medical field datasets. These findings offer valuable insights applicable to similar scenarios in distinct datasets.

We overcame image constraints such as small objects, overcrowded scenes, and small data sets by carefully adjusting distinct hyperparameters included in this process. We showed that object detection

is a potentially effective and efficient counting technique that could lead to good results, even given weak labeling of the ground truth. In future work, we will attempt to expand this process to other cell types that have similar problems. Applying the insights obtained during this project to a new, larger data set will eventually lead to models with higher accuracy. Moreover, at least three labeled data sets with significant numbers of images should be defined, taking into account the images' complexity. As perceived during the experiments, the model's performance in inferring counts is positively associated with the clump of cells present in the images. This should be considered in future works.

4. A STACKING-BASED ARTIFICIAL INTELLIGENCE FRAMEWORK FOR AN EFFECTIVE DETECTION AND LOCALIZATION OF COLON POLYPS

Polyp detection through colonoscopy is a widely used method to prevent colorectal cancer. The automation of this process aided by artificial intelligence allows faster and improved detection of polyps that can be missed during a standard colonoscopy. In this work, we propose to implement various object detection algorithms for polyp detection. To improve the mean average precision (mAP) of the detection, we combine the baseline models through a stacking approach. The experiments demonstrate the potential of this new methodology, which can reduce the workload for oncologists and increase the precision of the localization of polyps. Our proposal achieves an mAP of 0.86, translated into an improvement of 34.9% compared to the best baseline model and 28.8% with respect to the weighted boxes fusion ensemble technique.

4.1. INTRODUCTION

In the United States, colorectal cancer (CRC) stands as the third leading cause of cancer-related deaths and it is expected to cause more than 50.000 fatalities by 2022 (ACS, 2020). Additionally, recent studies show that CRC incidence in adults younger than 50 years old has nearly doubled since the early 1990s (Stoffel & Murphy, 2020) . Colonoscopy is considered the most effective procedure to detect colon polyps and cancer (Issa & NouredDine, 2017) and is of paramount importance for effective prevention and reduced risk of death from CRC. Evidence suggests that having a colonoscopy was associated with a decrease of 67% in the risk of death from CRC (Doubeni et al., 2018) and a 70% reduction in the incidence of late-stage CRCs (Doubeni et al., 2013) . However, research has shown that in patients undergoing colonoscopy, 25% of polyps are missed (Leufkens et al., 2012). Reasons behind the oversight include overloaded healthcare systems, the presence of fat and small-sized polyps, or workers' lack of experience (N. H. Kim et al., 2017; Maeng et al., 2007; Wallace et al., 2022).

With the rise of artificial intelligence, significant technological advances have occurred in the medical and healthcare field (Bohr & Memarzadeh, 2020). Deep learning (DL) is widely used as a computer vision tool to classify and detect lesions and many diseases by efficiently addressing the unique challenges of medical data (Esteva et al., 2021).

In polyp detection, evidence shows that using convolutional neural networks (CNNs) to detect polyps automatically under colonoscopy can improve the detection rate. Qadir et al. (Qadir et al., 2021) proposed a single-shot feed-forward fully convolutional neural network to develop a real-time polyp detection model using two-dimensional Gaussian masks. Li et al. (W. Li et al., 2021) used an adaptive training sample to select high-quality training samples to improve generalizability on the accurate segmentation of polyps. Taş et al. (Taş & Yılmaz, 2021) proposed implementing Faster R-CNN with a preprocessing approach based on a super-resolution method to improve the model's performance in detecting colon polyps. Tang et al. (Tang et al., 2021) also used Faster R-CNN with transfer learning to improve polyp detection. The YOLO algorithm has also been proposed to improve the efficiency of polyp detection. Guo et al. (Z. Guo et al., 2020) proposed an automatic polyp detection framework based on Yolov3 and active learning to reduce the rate of false positive polyp detection. Pacal et al.

(Ishak Pacal & Karaboga, 2021) considered Yolov4 for real-time polyp detection, and Wan et al. (Wan et al., 2021) used YOLOv5 for the same purpose. Jha et al. (Jha et al., 2021) applied EfficientDet, RetinaNet, Faster R-CNN, and YOLOv4 to compare their performance on polyp segmentation. Wu et al. (Lingyun Wu et al., 2021) compared UNet, Faster R-CNN, R-FCN, RetinaNet, Yolov3, FCOS, and PraNet and presented a spatial–temporal feature transformation to detect and localize polyps in endoscopy videos automatically.

Ensemble techniques were also considered to improve the polyp detection task. Sharma et al. (Sharma et al., 2022) applied a voting ensemble technique combining the results of ResNet101, GoogLeNet, and Xception for polyp classification. Younas et al. (Younas et al., 2022) proposed a similar approach by implementing a weighted ensemble of GoogleNet and ResNet50, among others, to improve the accuracy of the polyp class identification. In segmentation, DivergentNets (Thambawita et al., 2021) combines five models, and masks are averaged to make the final segmentation mask. In object detection, Hong et al. (Honga et al., 2021) and Polat et al. (Polat et al., 2020) used weighted boxes fusion methods as an ensemble technique to combine predictions from different models.

The purpose of our study was to analyse the efficacy of implementing a stacking approach to combine the predictions of distinct object detection techniques with the goal of improving the precision in polyp detection.

4.2. METHODS

4.2.1. Baseline models

In this study, we approach the polyp detection problem using five well-known object detection algorithms proposed in the literature.

Faster R-CNN, defined by Ren et al. (Ren et al., 2017), is a two-stage object detection model, where in the first module, regions of interest are proposed, and in the second stage, Fast R-CNN (R. Girshick, 2015b) is applied to detect the final boxes and classify them.

Fully Convolutional One-Stage Object Detection (FCOS) is an anchor-box-free single-stage object detection model proposed by Tian et al (Tian et al., 2019). By eliminating the predefined set of anchor boxes and all related hyperparameters, FCOS avoids computation related to this aspect, with the advantage of being a more straightforward and solid alternative to other object detection algorithms.

RetinaNet (T. Y. Lin et al., 2020) is a one-stage framework that uses focal loss to prevent the high number of negative detections from overwhelming the detector during training.

EfficientDet (Tan et al., 2020) is a single-shot detector that uses EfficientNets (Tan & Le, 2019) as the backbone network along with weighted bidirectional feature networks for feature fusion.

Ultralytics (Jocher et al., 2021) proposed YOLOv5 as a recent update to the YOLO family of models. YOLO algorithms are characterized by being the first object detection model that combined bounding box prediction and object classification into a single end-to-end differentiable network.

Although one-stage detectors have high inference speed, two stage-detectors are known for their high localization capability and recognition accuracy.

4.2.2. Ensemble techniques

To compare our method against other ensemble algorithms, we evaluate the performance of four distinct algorithms, considering six variants in total.

In Non-Maximum Suppression (NMS) (Neubeck, 2006), all detection boxes are sorted according to their confidence scores, and the detection box D with the maximum score is selected, while the remaining boxes that overlap D more than a predefined threshold are suppressed. These steps are recursively applied to the remaining boxes.

In Soft-NMS (Bodla et al., 2017), the authors propose a simple change to NMS to surpass the NMS limitation where detection proposals with high Intersection over Union (IoU) and high confidence can be removed. The algorithm decays the detection scores of all the detection boxes as a continuous function of their overlap with D. Two versions of Soft-NMS are tested in this study. In the first version a Gaussian distribution is implemented to modify the detection scores, whereas in the second, a linear function is used.

In Non-Maximum Weighted (NMW) (H. Zhou et al., 2017), all detection boxes are considered, and a weighted box is created using IoU values. In this algorithm, the confidence scores are not changed, and the IoU value is used to weight the boxes. Furthermore, NMW does not consider the number of models used in the ensemble.

In Weighted Boxes Fusion (WBF) (Solovyev et al., 2021), similar to NMW, all detection boxes are considered to create a weighted box. However, in WBF, the confidence value is changed using an average value of all the boxes used in each fusion. The coordinate of the fused box is a weighted sum of coordinates of each box where weights are the confidence for boxes. In this case, the boxes with more significant scores will have more influence in defining the coordinates of the fused box than boxes with lower scores would have. A second version of this approach is applied, WBF maximum, where confidence in weighted boxes is calculated using the maximum value instead of using an average value.

4.2.3. Multistage Algorithms

Cascade R-CNN (Cai & Vasconcelos, 2018) is a multistage object detection algorithm, considered an extension of R-CNN, where stages are trained sequentially, using the output of one stage to train the next one. By adjusting the bounding boxes at each stage, this approach tries to optimize the IoU values, which sequentially allows the algorithm to be more selective against close false positives for training the next stage.

4.2.4. Our proposal: StackBox

In this work, we propose a novel ensemble technique to combine the predictions of different models into a final improved prediction. In the stacking approach, we combine multiple algorithms via meta-

learning. This procedure involves two or more base models, often referred to as level-0 models or base learners, and a meta-model (which is also called a level-1 model) that combines the predictions of the level-0 models. In stacking, base learners fit on the training data, and those predictions are combined at the end; the resulting combination is then added as input features in the meta-model.

StackBox is a stacking technique that uses a machine-learning model to learn how best to combine the predictions from contributing base learners. Based on the training data set's predictions, base learners (level 0) are combined and are trained using a meta-model (level 1). This stacking technique will combine the capabilities of different base learners, which in this case are traditional object detection algorithms, and the meta-model, a traditional machine learning regressor, trained using the predictions of the base learners on training data, which can be subsequently used to predict new coordinates on the test data, using as input the predictions in the test set, as seen in Figure 4.1. When applying StackBox, a different treatment is used in training and test data.

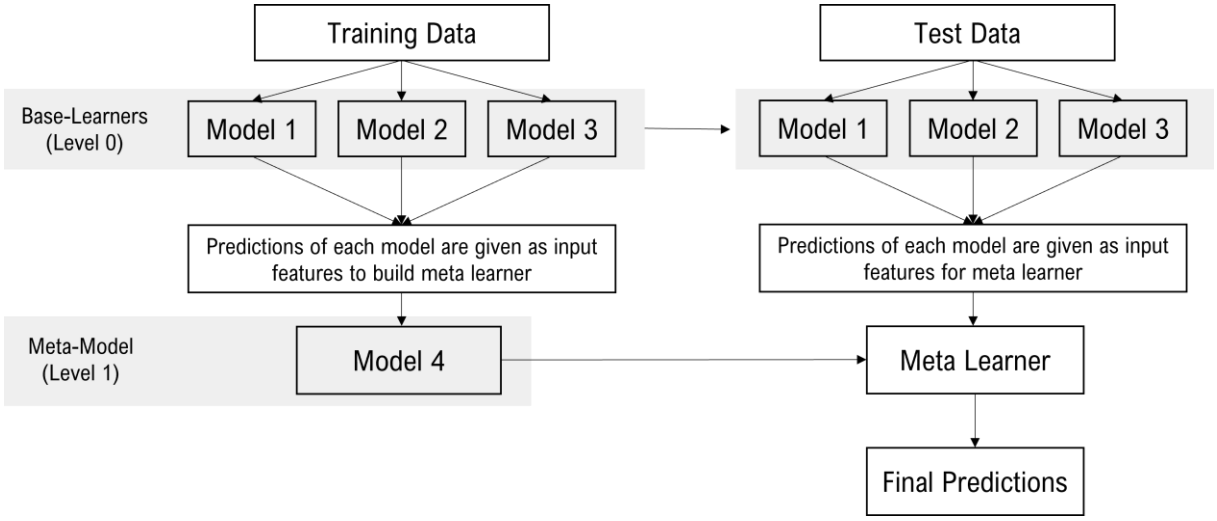

Figure 4.1 – Illustration of StackBox framework.

Note. The framework builds distinct base learners (using the training data), and from these models predicts the bounding boxes around the detected objects. Using these predictions as input and using the ground-truth as output, a meta-model combines the base learners' output, building a new model with improved performance. The base learners built previously are subsequently applied to the test data set to detect polyps on unseen data. Finally, these detections are used as input features for the meta-learner built on the training data to obtain the final predictions on the test data set.

In training data, we assume that the target of the meta-model is the ground truth bounding box, and the input is the base models' predictions that have the highest IoU associated with the ground truth. In a ground truth where no prediction is available (i.e., where no predicted box is found in any of the models), applying the meta-learner will not be considered. In case the number of predictions available for a specific ground truth is lower than the number of base models used, the missing predictions will be replaced by the values of the predicted box with the highest IoU, independently of the model. In this way, each ground truth will be associated with different predicted boxes, in the same number as the base learners.

In object detection, each object of interest is outlined by a bounding box, determined by the x and y coordinates. In this way, each predicted box would be represented by four coordinates, namely x_{min}, y_{min}, x_{max} and y_{max}, where min and max stands for minimum and maximum value. Thus, as can be seen in Step 2 of Figure 4.2, each ground-truth is associated with a set of coordinates (and the cardinality of this set corresponds to the number of the base learners). Subsequently, each coordinate (x_{min}, y_{min}, x_{max} and y_{max}) will be split, and a meta-model will be applied to each of them. More specifically, each coordinate individually will be considered to apply a metalearner. As an example, for x_{min}, a new data set is built where the number of rows is the same as the number of objects of interest, and the input features are the predictions of the coordinate x_{min} obtained by each base learner, while the output is the x_{min} of the ground truth. Figure 4.2 shows all the steps of the proposed StackBox technique when processing the training data.

In the test set, we need to define the boxes that will be the input for the meta-learner acquired on training data. At this point, we consider each model's prediction in the test data as the ground truth. For each prediction in a first model, we find the boxes from the remaining models with the highest IoU and repeat the process for them. This process will lead to several duplicated inputs. All duplicated inputs are removed, and finally we apply the meta-learner obtained in training data to predict the new boxes. Afterward, we apply a NMS strategy to all predictions to remove boxes with an IoU overlap higher than 0.5, keeping the one with the highest confidence. Figure 4.3 shows all the steps of the proposed StackBox technique for the analysis of the test set. The source code is publicly available at https://github.com/calbuquerque-novaims/StackBox.

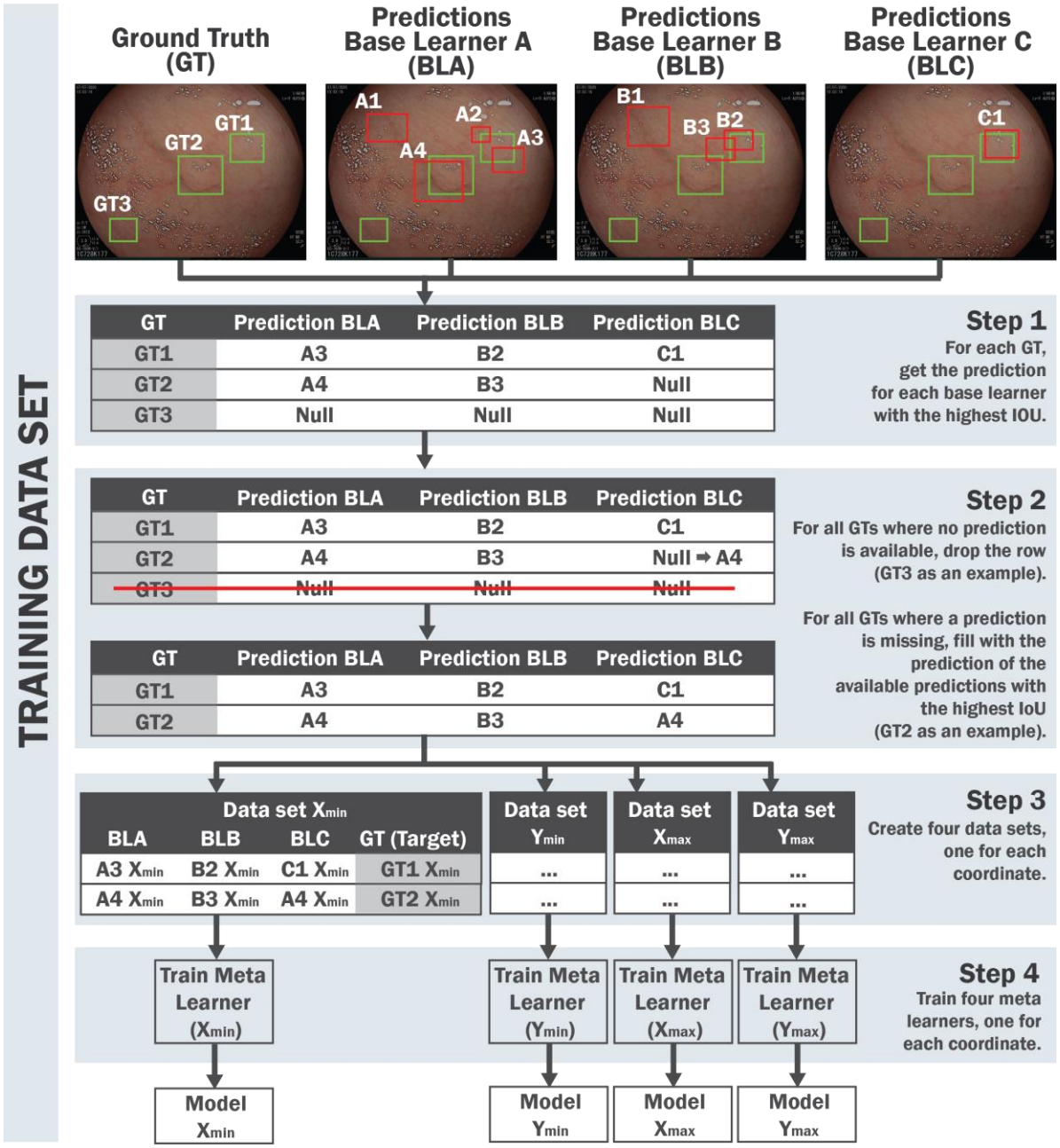

Figure 4.2 – StackBox over the training data.

Note. In Step 1, for each ground-truth, we find the prediction of each base learner with the highest IoU. In Step 2, the ground truths without associated predictions are removed; when the number of predictions is lower than the number of base learner models used, the null values are filled with the prediction where the IoU with the ground-truth is higher. In Step 3, four data sets are created according to the coordinates available for each bounding box. The predictors include the corresponding coordinate of each base learner, and the target is the analogous coordinate of the ground truth. In Step 4, a meta-learner is applied to each data set, and the final models are saved for subsequently application to test data. The ground truth is represented by the green bounding boxes and the predictions by the red bounding boxes.

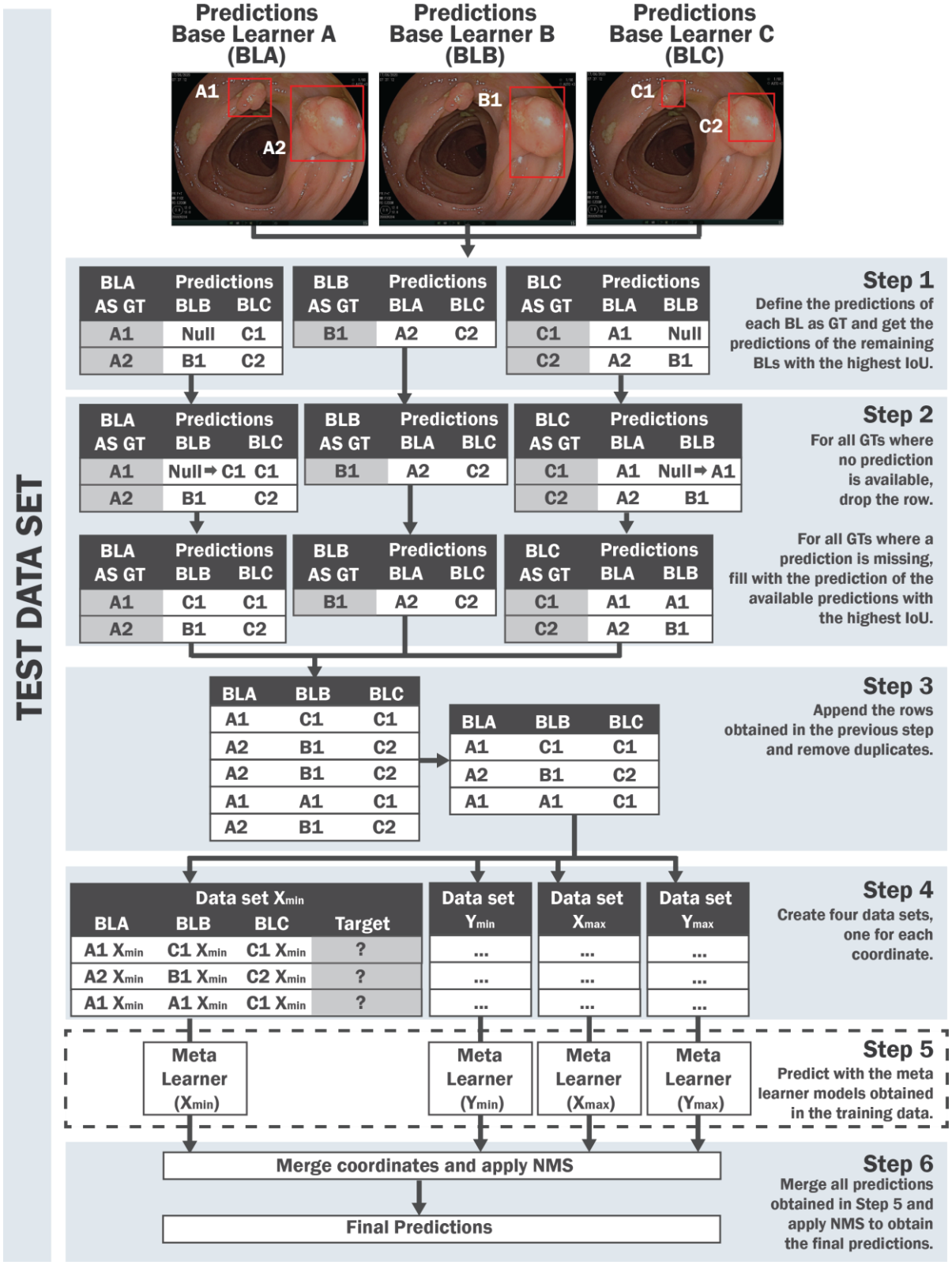

Figure 4.3 – StackBox over the test data.

Note. In Step 1, the predictions of each base learner are considered as the ground truth, one at a time, and the predictions returned by the remaining models that have the highest IoU with the ground truth are chosen. In Step 2, similar to the method used in the training data, the ground truths without associated predictions are removed. When the number of predictions is lower than the number of base learner models used minus one, the null values are filled with the prediction where the IoU with the ground truth is higher. In Step 3, all matches obtained in the previous step are concatenated, and the duplicates are removed. In Step 4, four data sets are created according to the coordinates available for each bounding box. The predictors include the corresponding coordinate of each base learner, and the target is predicted using the meta learner models obtained in training data (Step 5). In Step 6, the predictions are combined, and NMS (with a threshold of 0.5) is applied to remove redundant boxes. The red boxes represent the predictions of the base learner models in the test data.

Figure 4.4 shows an overview of the StackBox workflow, where the considered meta-learner is the Linear Regression.

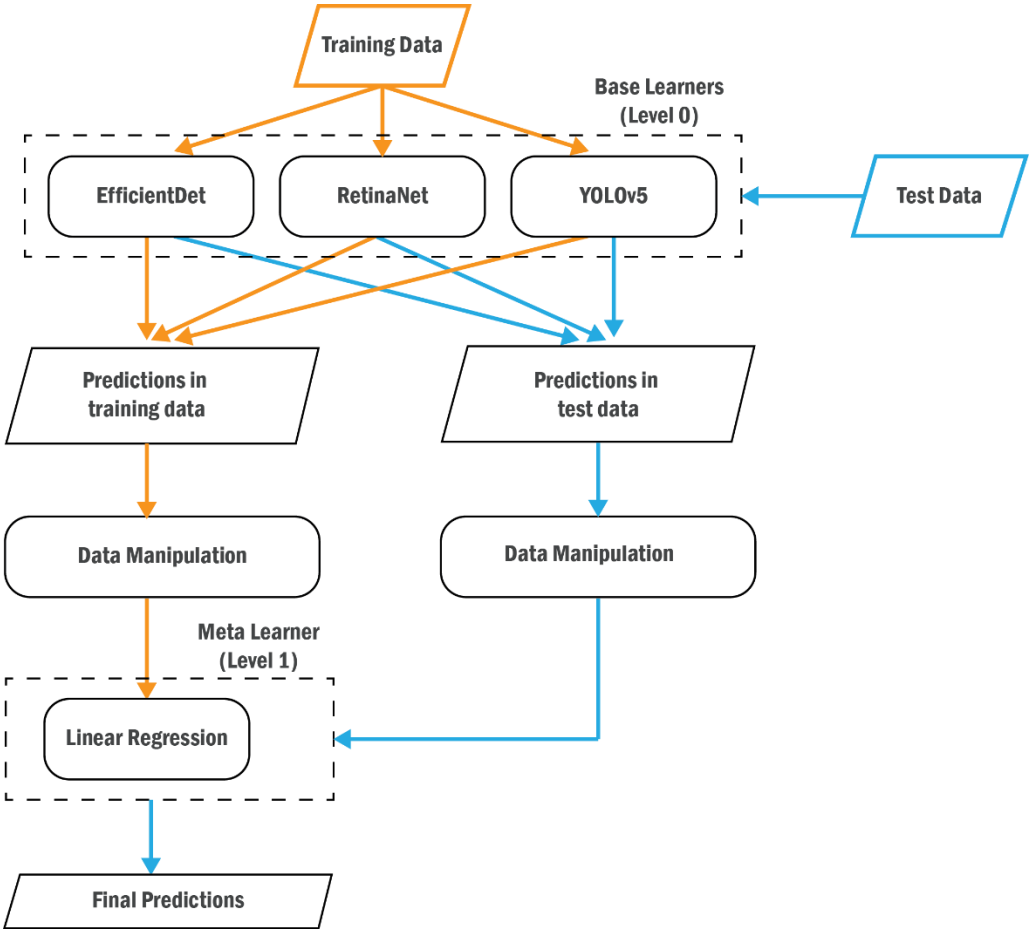

Figure 4.4 – StackBox general overview.

Note. The orange flow represents the training stage, and the blue flow represents the testing stage. The data manipulation process in the training stage includes Steps 1 to 3, detailed in Figure 4.2. The data manipulation process in the testing phase comprises the Steps 1 to 4, described in Figure 4.3.

We tested different machine learning models as meta-learner. Results show the performance of our stacking technique by applying Linear Regression (LR), Adaboost, Random Forest (RF), GradientBoosting (GB), and XGBoost.

To validate the effectiveness of our proposal, we perform three experiments:

- A comparison with baseline models, where we compare our stacking technique with five widely used object detection models: Faster R-CNN, FCOS, RetinaNet, EfficientDet, and YOLOv5.
- A comparison of our stacking approach with some available ensemble techniques: NMS, Soft-NMS NMW, and WBF.
- A comparison with a multistage approach, Cascade R-CNN.

In all experiments, standard metrics for object detection (Padilla et al., 2021) are employed for performance measurement, namely AP@[.5:.05:.95], AP@.50, AP@.75, APM, APL, AR1, AR10, ARM, ARL, and mAP (IOU=.50).

4.2.5. Polyp data set

BKAI-IGH Neopolyp-Small (An et al., 2022; Ngoc Lan et al., 2021), a data set of 1000 annotated endoscopic images provided publicly by BK.AI, Hanoi University of Science and Technology incorporation with the Institute of Gastroenterology and Hepatology (IGH), is curated to train and benchmark the proposed approach. The images were collected in IGH, and annotations were added and verified by two experienced endoscopists in IGH.

Originally developed as a segmentation problem, annotations in the data set were converted to a detection problem, where a bounding box identifies each polyp. The data set is randomly split into a training set of 800 images and a test set of 200 images. A fivefold cross-validation approach was used to measure the performance of each of the base models, and the ensembles applied, with no overlapping, and average scores were calculated. The original size of the images is not constant and ranges from 959×1280 pixels to 1024×1280 pixels. On training, all images were converted to 640×640 pixels.

4.2.6. Experimental setup

All experiments were conducted using models provided by IceVision, a framework for object detection and deep learning that offers an end-to-end workflow with different models from TorchVision, Open MMLab's MMDetection, and Ultralytic's YOLOv5, among others. Each base learner model was trained during 50 epochs, and we applied transfer learning using a previously trained model on the Microsoft COCO (T.-Y. Lin et al., 2014) data set. As a backbone, we used ResNet101 in RetinaNet, Faster R-CNN and FCOS, D1 in EfficientDet, and the large version of YOLOv5. Each model's learning rate was automatically defined by the Fastai (Howard & Gugger, 2020) learning rate finder. The Cascade R-CNN was implemented using Detectron2 (Y. Wu et al., 2019). The meta learner models were applied using SkLearn and XGBoost library. The ensemble and stacking techniques use the results of the three best

baseline models. All metrics were measured using Rafael Padilla's tool (Padilla et al., 2021). In ensemble techniques, the Weighted Boxes Fusion tool was applied (Solovyev et al., 2021).

The experiments were executed on a Linux system with an Intel Core i7-10750H CPU @ 2.60 GHz, a NVIDIA GeForce RTX 3080 Laptop GPU, and 16 GB of RAM.

4.3. RESULTS

The fivefold cross-validation is employed to evaluate each model's performance, where the training set and the test set do not share the same images. As seen in Figure 4.5, the StackBox algorithm, independently of the model used as meta-learner, achieves significantly higher results concerning mAP when compared to base learner models and ensemble techniques.

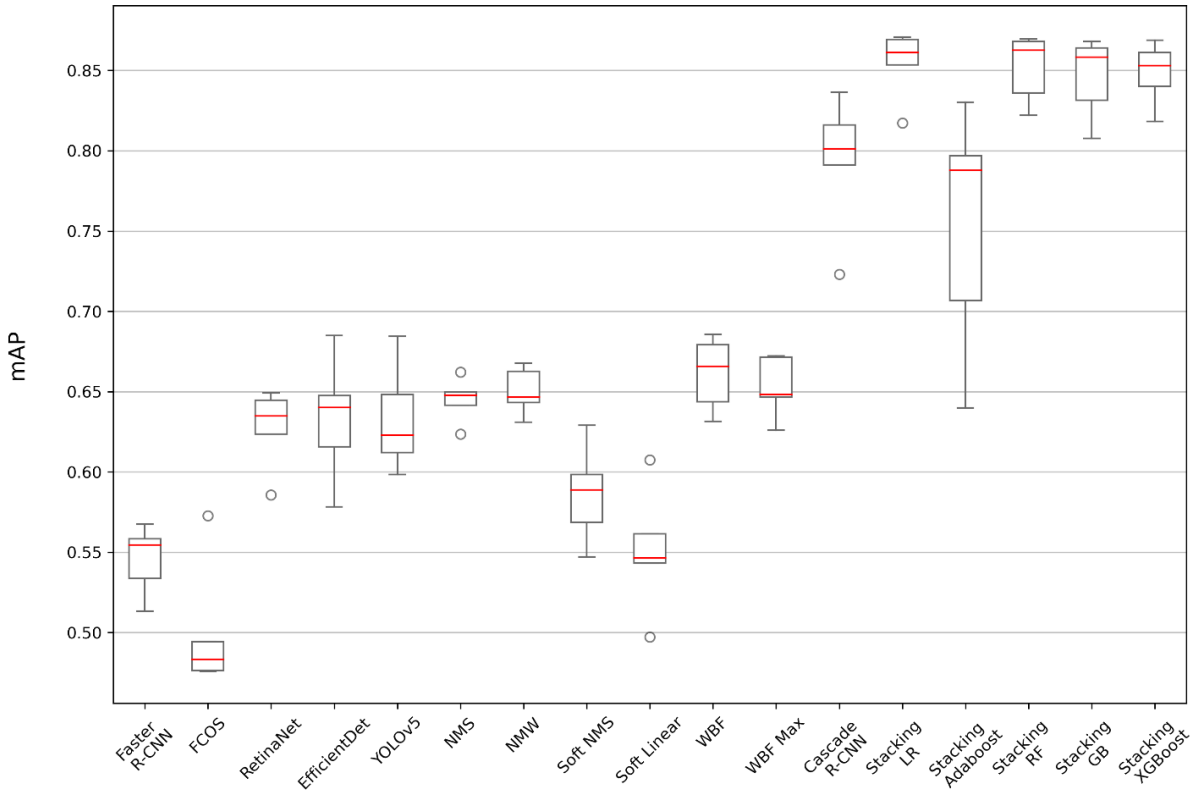

Figure 4.5 – mAP comparison through boxplots.

Note. The tested models (base-learner models, ensemble models, Cascade R-CNN and StackBox models) are compared in terms of mAP. The figure shows the distribution of the results on the fivefold cross-validation.

Concerning mAP, EfficientDet, RetinaNet, and YOLOv5 achieve similar results, of around 0.63 on average. The WBF ensemble technique is able to improve this value to 0.66. Cascade R-CNN achieves an average mAP value of 0.79. Our proposal, StackBox, raises the mAP to 0.85 in all the meta-learners used, except for Adaboost, where the mAP is 0.75, as shown in Figure 4.5.

Table 4.1 presents the results that average the five folds together concerning precision. In object detection, precision is a model's capability to identify only relevant objects, corresponding to the percentage of correct positive predictions (Padilla et al., 2021).

Table 4.1 - Model comparison in terms of precision.

Algorithm	**AP@[.5:.05:.95]**	**AP@.50**	**AP@.75**	**AP_M**	**AP_L**
Faster R-CNN	0.20 ± 0.01	0.54 ± 0.02	0.07 ± 0.01	0.02 ± 0.02	0.24 ± 0.02
FCOS	0.19 ± 0.01	0.50 ± 0.04	0.04 ± 0.01	0.08 ± 0.00	0.21 ± 0.01
RetinaNet	0.25 ± 0.01	0.63 ± 0.02	0.10 ± 0.01	0.03 ± 0.03	0.28 ± 0.02
EfficientDet	0.24 ± 0.02	0.63 ± 0.04	0.06 ± 0.02	0.03 ± 0.02	0.27 ± 0.03
YOLOv5	0.24 ± 0.02	0.63 ± 0.03	0.05 ± 0.03	0.02 ± 0.01	0.27 ± 0.02
NMS	0.24 ± 0.02	0.64 ± 0.02	0.06 ± 0.02	0.03 ± 0.03	0.27 ± 0.02
NMW	0.25 ± 0.02	0.65 ± 0.02	0.08 ± 0.02	0.03 ± 0.02	0.28 ± 0.02
Soft NMS	0.23 ± 0.02	0.58 ± 0.03	0.08 ± 0.01	0.03 ± 0.02	0.26 ± 0.03
Soft Linear	0.22 ± 0.02	0.55 ± 0.04	0.08 ± 0.02	0.03 ± 0.02	0.25 ± 0.03
WBF	0.26 ± 0.02	0.65 ± 0.02	0.09 ± 0.03	0.03 ± 0.01	0.29 ± 0.02
WBF Max	0.25 ± 0.02	0.65 ± 0.02	0.09 ± 0.03	0.03 ± 0.02	0.28 ± 0.03
Cascade R-CNN	0.64 ± 0.02	0.79 ± 0.04	0.72 ± 0.03	**0.31** ± 0.11	**0.69** ± 0.01
StackBox with LR	**0.65** ± 0.03	**0.85** ± 0.02	**0.75** ± 0.04	**0.31** ± 0.03	**0.69** ± 0.04
StackBox with Adaboost	0.35 ± 0.09	0.75 ± 0.08	0.29 ± 0.16	0.06 ± 0.03	0.39 ± 0.10
StackBox with RF	0.64 ± 0.03	0.85 ± 0.03	0.74 ± 0.04	**0.31** ± 0.02	**0.69** ± 0.04
StackBox with GB	0.62 ± 0.04	0.84 ± 0.03	0.72 ± 0.05	0.28 ± 0.03	0.67 ± 0.04
StackBox with XGBoost	0.63 ± 0.03	**0.85** ± 0.02	0.73 ± 0.04	0.29 ± 0.02	0.68 ± 0.04

Note. The results present the average values obtained by combining the 5 folds ± SD of those results. AP@[.5:.05:.95] computes the average precision with 10 different IoU thresholds and takes the average among all computed results. In AP@.50 and AP@.75, the interpolation is performed in N = 101 recall points, and the first uses an IoU threshold equal to 0.5, whereas the second uses a threshold of 0.75. AP_M only evaluates medium-sized ground-truth objects, whereas AP_L only evaluates large ground-truth objects (Padilla et al., 2021). Bold denotes the highest values for each metric. The StackBox with Logistic Regression stands as the best model for all the metrics under consideration.

RetinaNet achieves the best results in terms of precision when comparing base learner models, but with a slight difference from EfficientDet and YOLOv5, as seen in Table 4.1. Faster R-CNN and FCOS achieve the worst performance. Considering ensemble techniques, we can see a subtle improvement for some of the techniques, with relevance to WBF, with an improvement of 0.02 in AP@[.5:.05:.95]

and AP@.50 compared with the best base learner models. In our approach, independently of the meta learner algorithm used, except for Adaboost, we verify a significant improvement regarding the base learner models and the ensemble techniques. Cascade R-CNN achieves similar results to our StackBox technique in APM and APL but slightly worse results in the remaining measures. StackBox with LR increases precision to around 0.4 in AP@[.5:.05:.95] and in APL, 0.7 in AP@.75, and 0.2 in AP@.50 and APM when compared to base learner models and the remaining ensemble techniques.

To compare the performance of all tested models concerning recall, we measure the performance of all models in various metrics usually applied in object detection research. Recall is the capability of a model to find all the ground-truth bounding boxes, corresponding to the percentage of correct positive predictions among all given ground truths (Padilla et al., 2021).

In Table 4.2, we can verify that results show similar results as precision. One clear difference is that Faster R-CNN achieves results similar to RetinaNet concerning the recall, whereas FCOS, is the worst model (e.g., in precision). Cascade R-CNN achieves similar results when compared with StackBox, but with lower performance in AR10 and ARL. StackBox with LR achieves the highest average values, with 0.65 in AR1, 0.71 in AR10, 0.34 in ARM, and 0.76 in ARL.

Table 4.2 – Model comparison in terms of recall.

Algorithm	**AR_1**	**AR_{10}**	**AR_M**	**AR_L**
Faster R-CNN	0.30 ± 0.01	0.33 ± 0.01	0.07 ± 0.04	0.36 ± 0.02
FCOS	0.24 ± 0.02	0.24 ± 0.02	0.01 ± 0.01	0.27 ± 0.01
RetinaNet	0.31 ± 0.01	0.32 ± 0.01	0.06 ± 0.04	0.36 ± 0.02
EfficientDet	0.29 ± 0.01	0.30 ± 0.01	0.06 ± 0.02	0.34 ± 0.02
YOLOv5	0.29 ± 0.01	0.30 ± 0.01	0.08 ± 0.03	0.34 ± 0.01
NMS	0.31 ± 0.01	0.32 ± 0.01	0.08 ± 0.04	0.36 ± 0.02
NMW	0.32 ± 0.01	0.33 ± 0.01	0.08 ± 0.04	0.37 ± 0.01
Soft NMS	0.31 ± 0.01	0.35 ± 0.01	0.08 ± 0.04	0.39 ± 0.02
Soft Linear	0.31 ± 0.01	0.36 ± 0.01	0.08 ± 0.04	0.40 ± 0.02
WBF	0.31 ± 0.01	0.33 ± 0.01	0.08 ± 0.04	0.37 ± 0.01
WBF Max	0.32 ± 0.01	0.33 ± 0.01	0.08 ± 0.04	0.37 ± 0.01
Cascade R-CNN	**0.65** ± 0.02	0.69 ± 0.03	**0.34** ± 0.12	0.75 ± 0.02
StackBox with LR	**0.65** ± 0.03	**0.71** ± 0.03	**0.34** ± 0.03	**0.76** ± 0.04
StackBox with Adaboost	0.42 ± 0.08	0.44 ± 0.09	0.08 ± 0.04	0.49 ± 0.10
StackBox with RF	**0.65** ± 0.03	0.70 ± 0.03	**0.34** ± 0.02	0.75 ± 0.04
StackBox with GB	0.64 ± 0.03	0.69 ± 0.03	0.31 ± 0.03	0.74 ± 0.04
StackBox with XGBoost	**0.65** ± 0.03	0.70 ± 0.03	0.33 ± 0.03	0.75 ± 0.04

Note. The results present the average values obtained by combining the 5 folds ± SD of those results. AR_1 measures the average recall considering up to one detection per image, averaged over all IoUs,

whereas AR_{10} considers 10 detections at most. Similar to precision, AR_M measures the average recall on medium-sized ground-truth objects, whereas AR_L only evaluates large ground-truth objects(Padilla et al., 2021). Bold denotes the highest values for each metric. The StackBox with Logistic Regression stands as the best model for all the metrics under consideration.

4.4. DISCUSSION

Many studies have demonstrated the suitability of object detection approaches for efficiently detecting polyps. Different algorithms have been tested, and to achieve better results on the task, ensemble techniques combining the predictions of these algorithms have been proposed. Knowing that different algorithms have their specificities, advantages, and disadvantages, the results can significantly differ when considering the precision, recall, and mAP of the resulting models. Following this reasoning, in this study, we demonstrate that the stack of predictions from separate object detection algorithms improved the precision of polyp detections. Independently of the meta learner used, the mAP increased significantly compared to base learner algorithms such as EfficientDet and RetinaNet, prior ensemble techniques such as NMS and WBF, and multistage architecture Cascade R-CNN.

To the best of our knowledge, this is the first stacking approach to combine the predictions of the coordinates of different object detection algorithms. In the context of this study, the technique was applied to polyp detection. However, it can be easily used in other medical applications and, in general, in all the problems in which the precision of the localization of objects of interest is the main concern.

Due to the different natures of the algorithms used, the predictions of each base model are computed differently, leading to different bounding boxes. We can use this dissimilarity and the advantages of each algorithm to combine them in a more precise prediction.

Regarding the mAP, the base learner with the highest value is RetinaNet, with an average mAP of 0.63, whereas the WBF ensemble technique can increase this value to an average of 0.66 and Cascade R-CNN can improve this value to 0.79. Our proposal, StackBox with LR, achieves an average mAP of 0.85, representing an increase of 0.22 compared to RetinaNet, 0.19 compared to WBF, and 0.06 compared to Cascade R-CNN.

Concerning precision, EfficientDet, RetinaNet, and YOLOv5 are the three best base learner models for most of the considered metrics. Using ensemble techniques, we can improve those results by around 0.02, and, with Cascade R-CNN, we achieve slightly worse results when compared to StackBox. Our approach can increase the precision of the models significantly. Considering stacking with LR, we double the performance (for most metrics) compared to base learner models.

Concerning recall, FCOS presents the worst results compared to the other baseline models. Faster R-CNN, RetinaNet, EfficientDet, and YOLOv5 achieve similar results, with approximately 0.3 in AR1 and AR10, 0.07 in ARM, and 0.35 in ARL. Prior ensemble techniques can slightly improve those values, but StackBox increases AR1 to 0.65, AR10 to 0.71, ARM to 0.34, and ARL to 0.76. Cascade R-CNN presents slightly worse results than StackBox does.

Figure 4.6 shows the results achieved, on a sample image, by the models considered in this study. Clearly, StackBox, independently of the meta learner used, stands as the best performer, with significant improvement in the precision of the predicted boxes compared to the other methods under consideration.

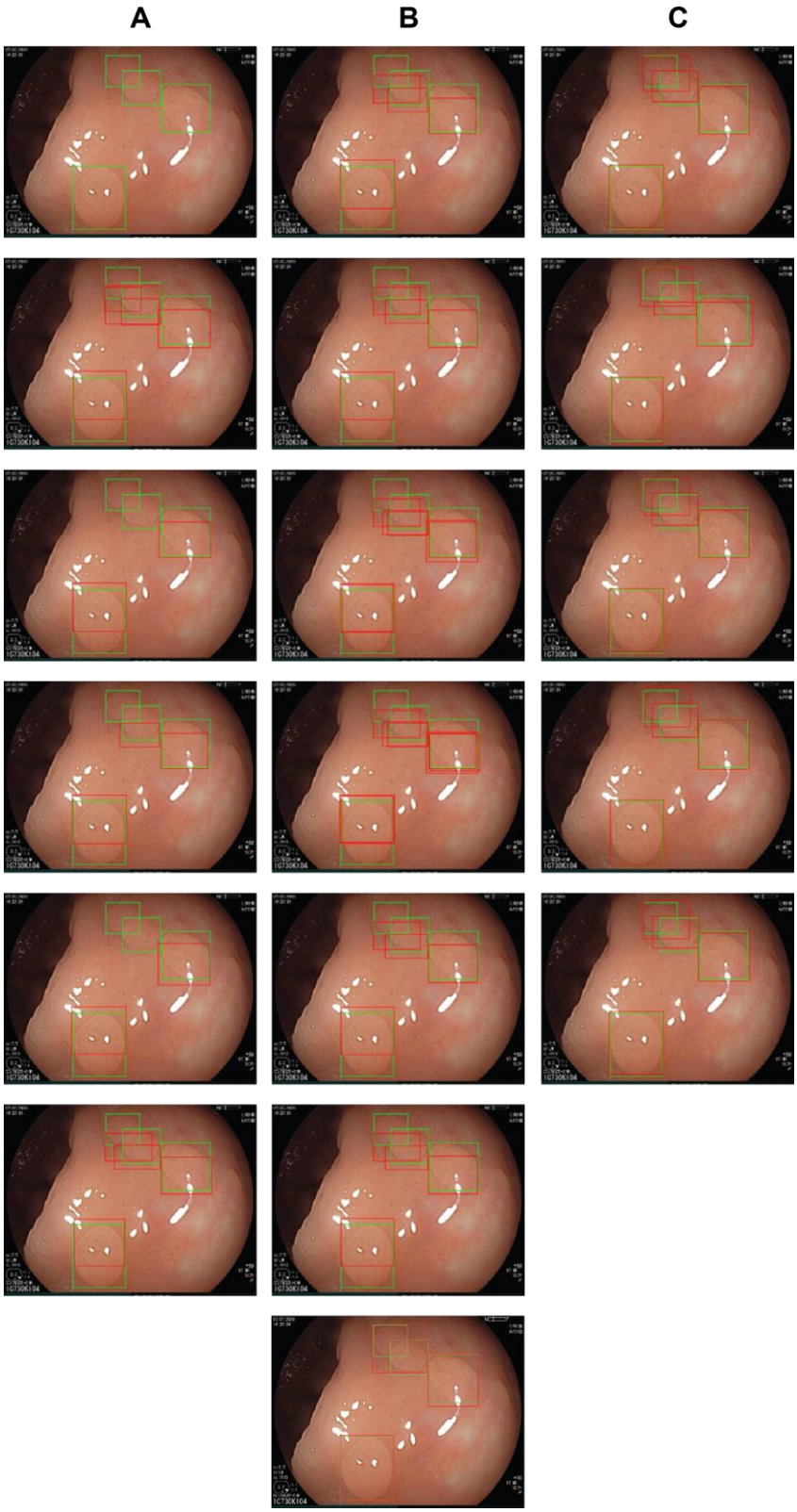

Figure 4.6 – Predictions comparison sample.

Note. Column A displays, from the top to the bottom, the results of the following models: GT, Faster R-CNN, FCOS, RetinaNet, EfficientDet, and YOLOv5. Column B evaluates the predictions using ensemble techniques (NMS, NMW, SOFT-NMS, Soft Linear, WBF, and WBF Max) and Cascade R-CNN. The third column reports the results of our stacking technique using different meta-learner models: StackBox with Logistic Regression, StackBox with Adaboost, StackBox with Random Forest, StackBox with Gradient Boosting, and StackBox with XGBoost. The ground truth is represented by the green bounding boxes and the predictions by the red bounding boxes.

Regarding the real-time applicability of this approach and to validate the practical usefulness of StackBox in real-world colonoscopy, we evaluate the processing time for each image. When we apply StackBox, the inference on new images includes the inference of each base learner model in the new data, the manipulation of those predictions in a format viable to apply stacking techniques, the stacking technique itself, and finally, the implementation of a NMS strategy to remove redundant boxes. For the example given, where we use EfficientDet, RetinaNet, and YOLOv5, the inference time is approximately 0.054, 0.057, and 0.010 s per image, respectively. The prediction manipulation to obtain the needed format for stacking application requires around 0.010 s per image. The inference during the stacking approach when implementing a LR demands 0.00048 s per image, and the NMS application requires around 0.020 s per image. Summing up all the procedures needed to obtain the final predictions, we obtain an inference time of 0.144 s per image, translating into around seven frames per second. This value is considered lower than inference times associated with widely used algorithms, such as the Faster R-CNN Inception ResNet V2 640×640 (0.206 s/image) (Dao, 2020).

This study poses the basis for further solutions to this challenging problem. In future works, this methodology can be applied to data sets with a larger number of samples (to improve the performance of the base learners), and more advanced strategies to combine the predictions of the base learners can be defined and analysed.

4.5. CONCLUSION

To achieve better results on the polyp detection task, in this paper, we proposed the use of StackBox. StackBox combines the predictions on training data sets from YOLOv5, RetinaNet, and EfficientDet by stacking the results with a meta-learner, aiming to build a model that can increase the detection capability over new data. Experimental results demonstrated the suitability of the proposed method for the polyp detection task. More specifically, StackBox can significantly improve the mAP of the detections, not only when compared to the tested baseline models, namely Faster R-CNN, FCOS, YOLOv5, RetinaNet, and EfficientDet, but also with respect to existing ensemble techniques, namely NMS, Soft-NMS, NMW, and WBF, and the multistage architecture Cascade R-CNN. These results, obtained by considering distinct metrics commonly used in object detection problems, demonstrate that StackBox is superior to all the tested approaches.

We believe that the proposed algorithm may contribute to successful colonoscopy procedures by reducing the polyp miss rate due to the increase in detection precision; furthermore, by combining several object detection frameworks with different skills on the task, we obtain different predictions, which will provide a more robust model with a higher polyp detection capability. Thus, StackBox can

be considered a procedure of significant relevance to CRC prevention using deep learning techniques, and the feasibility of the approach in real-world clinical practice is supported by its short inference time on new data.

The results achieved in this study open a wide range of future research directions, including the construction of generalizable models to deal with various object detection tasks.

5. CONCLUSION

5.1. SUMMARY OF FINDINGS

This dissertation offers, in Chapter 2, a comprehensive quantitative and qualitative analysis of object detection in medical imaging, yielding several noteworthy insights. Despite the emergence of deep learning object detection methods in 2014, medical imaging only adopted this approach in 2018, when the algorithms had evolved and demonstrated improved performance capabilities, which is a high demand in the medical field. Despite its relatively late arrival in the field, the introduction of object detectors in medical imaging has been modest and only thirteen studies were published in 2018. Nonetheless, our research findings indicate that this is a rapidly developing area of study. Four years later, the number of publications has increased by more than ten times, demonstrating the significant growth of the field. Additionally, it can be understood that while the establishment of object detectors in the field was initially cautious, it quickly expanded into various medical imaging techniques and a diversity of anatomical application domains. By 2022, 311 studies were available, most of which were featured in top journals, in the disciplines of Computer Science, Medicine, and Engineering.

Of the papers identified through the PRISMA methodology, 80 of them underwent qualitative analysis based on three key criteria: the algorithms applied, the imaging modalities employed, and the anatomical regions studied. The results showed that the majority of the publications applied either Faster R-CNN or YOLO algorithms, suggesting that both single-stage and two-stage object detectors can be effective, depending on the nature of the task. For instance, in applications where real-time predictions are a demand, such as endoscopic exams, the authors tend to prefer one-stage algorithms due to their ability to make faster inferences on new images. Additionally, the YOLO algorithm has gained relevance in recent years, with the various evolutions and developments across its multiple versions. However, there is still no consensus on the best algorithm to use, reinforcing the concept of the No Free Lunch Theorem, which states that all optimization algorithms perform identically on average when evaluated over the space of all possible problems. The strategy for one model surpassing another relies on the personalization and optimization of the algorithm for the particular task.

Regarding the utilization of imaging modalities, object detection algorithms are applied across a wide range of modalities, including CR scans, pathology images, and endoscopic imaging which are among the most commonly used. Nevertheless, object detection is also applied to CT scans, photography, MRI, ultrasound, fundus photo, and angiography imaging.

When it comes to the anatomical applications of object detection algorithms, digital pathology and microscopy imaging of tissues and blood smears are the most commonly explored. However, other areas are also frequently studied in this field, such as abdominal organs with a focus on colon polyp detection, digestive system organs with an emphasis on tooth-related tasks, bone imaging for fracture identification, and breast CR scans for breast lesion detection.

Chapter 3 features a fine-tuned and optimized version of Faster R-CNN used to tackle a complex case study of detecting and counting cancer cells in zebrafish xenografts. The dataset was challenging, with limitations such as small cells, crowded scenes, overlapping cells of diverse shapes, and only 97 images.

After evaluating various algorithms, including Faster R-CNN, RFCN, SSD, and YOLOv5, Faster R-CNN was identified as the most promising. To overcome these difficulties, several strategies were applied.

For addressing small objects, the image size was resized, leading to improved performance as the images became larger. This also expedited the convergence of the model, leading to the conclusion that low-resolution images hinder the model's training performance. Another strategy was to compare the performance of the model using an anchor generator with a stride of 8 pixels or 16 pixels. The presence of small objects suggests that a smaller stride would benefit the process. The atrous rate, which allows object-detection models to detect smaller objects, was used during training and resulted in slightly better performance than not using it, with no impact on the training time. Additionally, increasing the IoU threshold also showed a positive effect on detecting small objects.

To handle overcrowded scenes, modifications were made to the initial number of proposals. The images in the study had a wide range of cell counts, from 18 to 661 cells. Increasing the number of proposals from the default value to 3000 led to the best results without major changes, though it did significantly increase the training time.

The issue of a limited dataset was addressed through the implementation of various data augmentation techniques. Over fifteen techniques were tested, including combinations of the most effective methods, resulting in an mAP range of 59% to 82%, without and with data augmentation. This showed that data augmentation can be a useful tool for small datasets, which is common in the medical field.

In addressing the problem of overlapping cells with varying shapes, adjustments were made to the stride, scales, and aspect ratios of the anchor generator. To do this, an initial exploration was carried out on the dimensions of the objects of interest, in terms of width, height, and ratio. It was found that limiting the scales to values similar to the cell sizes in the dataset resulted in overfitting, but reducing the lowest value of the scale improved performance, better accommodating the presence of small objects in the data set.

Chapter 4 exposes a new stacking approach for detecting colon polyps in endoscopic imaging efficiently. To identify the most promising models, five different object detection models were analyzed and compared. The three models that demonstrated the most potential were YOLOv5, RetinaNet, and EfficientDet. By stacking the outcomes of these models in a unique process, significant improvements were obtained in comparison to the best baseline model and established ensemble techniques such as WBF, or multi-stage algorithms such as Cascade R-CNN. Our findings from this approach lead us to the conclusion that merging the strengths of multiple object detection frameworks can result in a more robust model, with an increased capability for polyp detection. Furthermore, the novel approach not only exhibits competitive inference times but can even surpass the performance of some standalone networks, such as Faster R-CNN with Inception Resnet V2.

5.2. CONTRIBUTIONS

Chapter 2 guides the reader through a comprehensive and systematic analysis, in quantitative and qualitative terms, of scientific literature and research, on the development and use of object detectors

in medical image analysis. Recurring to the PRISMA methodology, 311 records are identified and explored in a quantitative approach, considering aspects such as the source, authors, subject areas, citations, and countries. Additionally, a qualitative analysis is conducted, wherein records with an annual citation rate equal to or greater than five are examined and summarized. In this qualitative analysis, we investigate the object detection algorithms employed, the modality of medical imaging used, and the specific body parts under investigation.

Chapter 3 dives into the application of various object detection algorithms on a particular data set of zebrafish xenografts, intending to identify and quantify human cancer cells with the lowest error possible. This is an in-depth exploration in which we first identify the unique characteristics, particularities, and constraints of our data. We then proceed to the application of various algorithms, including one-stage object detectors (SSD and YOLOv5) as also two-stage approaches (Faster R-CNN and RFCN), to identify the most promising one. Subsequently, we conducted an extensive hyperparameter optimization to further improve and enhance the performance of our final model. Throughout this process, we take into consideration and address the limitations presented by this particular data set, such as its limited size, the irregular shape of the cells' morphology, the presence of overcrowded objects, the detection of significantly small cells, and the significant variation in the number of cells to be detected in each image. The results of our study lead to a promising framework adapted to the problem at hand. Additionally, they provide valuable insights on how to overcome the aforementioned limitations when applying, in this case, Faster R-CNN.

In Chapter 4, we introduce a novel, staking-based AI framework for the detection of polyps in colonoscopy images. To begin, we apply a comprehensive selection of object detection algorithms to our proposed data set, including Faster R-CNN, FCOS, RetinaNet, EfficientDet, and YOLOv5, to identify the approaches that demonstrate the highest level of performance for the problem at hand. Using these predictions, we develop an innovative procedure for ensemble and stacking these results, with the ultimate goal of maximizing the precision of the final prediction. Our results are then compared to other available and commonly used ensemble techniques, such as NMS, NMW, and WBF, as well as the multi-stage algorithm Cascade R-CNN. Throughout this work, the importance of precision in the identification of objects in medical image analysis is reinforced. Our proposal achieves an improvement of 34.9% on mAP compared to the best baseline model, and 28.8% when compared to the best ensemble technique.

5.3. LIMITATIONS AND FUTURE RESEARCH

This dissertation, and its related studies, present some limitations that are worth mentioning. The quantitative and qualitative analysis performed in this study, based on two widely used academic databases, has employed the PRISMA methodology to screen for relevant articles. However, due to the bi-disciplinary nature of the field of study, combining both medicine and computer science, it is apparent that each discipline utilizes distinct concepts and terminologies to address similar issues. The search strategy used in this study was focused on the title, abstract, and keywords of the publications. However, it became evident that many medical-oriented studies did not include AI concepts as a primary topic, while computer science-oriented works did not focus on medical concepts. To capture a more comprehensive set of relevant publications, future research should consider adjusting the

search queries to account for the different terminology used in both disciplines. Furthermore, a more extensive comparison between the developed work should be addressed, including important details such as the data sets used, the metrics used to evaluate the models and their corresponding performance, and the remarks of the most relevant publications. Additionally, the search was limited to articles and focused on English-language publications, which may have resulted in a biased selection. This filtering may have missed relevant studies conducted in different formats and languages. Considering these limitations, future research should aim to broaden the search scope, exploring a wider range of relevant studies in various formats and languages.

Concerning the proposed framework for cell quantification on zebrafish xenografts, some limitations should be taken into consideration. Due to the several parameters tested and the demanding computational and time resources needed, cross-validation, an essential technique for assessing the performance of a model and the generalization capability of the same to an independent dataset, was not applied during the experiments. This could lead to biased results, and overestimate the behavior of the model. Moreover, the dataset size created several constraints during training. Training with a bigger dataset leads us to poor statistical conclusions while evaluating the performance by training with a smaller number of samples implied a significant decrease in the performance of the model trained. For further research on this topic, we are aware that a wider range of algorithms should be tested, and the proposed architecture should be challenged also on different data sets with similar characteristics, to evaluate its generalization capability. Furthermore, a cross-validation approach is essential to avoid misleading conclusions.

The limitations of the stacking-based study developed in this dissertation are worth acknowledging. It is possible that a wider range of baseline models could have been tested, and it would be beneficial to evaluate the performance of the proposed method on various datasets. Moreover, the focus of the study is only on binary classification problems, and the algorithm is not capable of handling multiclass problems. In future work, a more comprehensive investigation into the proposed algorithm should be conducted, exploring a broader range of algorithms for comparison and a wider range of datasets, including both binary and multiclass problems.

6. REFERENCES

Abdurahman, F., Fante, K. A., & Aliy, M. (2021). Malaria parasite detection in thick blood smear microscopic images using modified YOLOV3 and YOLOV4 models. *BMC Bioinformatics*, *22*(1). https://doi.org/10.1186/s12859-021-04036-4

ACS. (2020). Colorectal Cancer Facts and Figures 2020-2022. *American Cancer Society*, *66*(11), 1–41. https://www.cancer.org/content/dam/cancer-org/research/cancer-facts-and-statistics/colorectal-cancer-facts-and-figures/colorectal-cancer-facts-and-figures-2020-2022.pdf

Adachi, M., Fujioka, T., Mori, M., Kubota, K., Kikuchi, Y., Xiaotong, W., Oyama, J., Kimura, K., Oda, G., Nakagawa, T., Uetake, H., & Tateishi, U. (2020). Detection and diagnosis of breast cancer using artificial intelligence based assessment of maximum intensity projection dynamic contrast-enhanced magnetic resonance images. *Diagnostics*, *10*(5). https://doi.org/10.3390/diagnostics10050330

Ahmed, I., Ahmad, M., Ahmad, A., & Jeon, G. (2021). IoT-based crowd monitoring system: Using SSD with transfer learning. *Computers & Electrical Engineering*, *93*, 107226.

Ahsan, M. M., Luna, S. A., & Siddique, Z. (2022). Machine-Learning-Based Disease Diagnosis: A Comprehensive Review. *Healthcare*, *10*(3), 541.

Al-antari, M. A., Han, S.-M., & Kim, T.-S. (2020). Evaluation of deep learning detection and classification towards computer-aided diagnosis of breast lesions in digital X-ray mammograms. *Computer Methods and Programs in Biomedicine*, *196*. https://doi.org/10.1016/j.cmpb.2020.105584

Alahmari, S. S., Goldgof, D., Hall, L. O., & Mouton, P. R. (2019). Automatic cell counting using active deep learning and unbiased stereology. *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*, 1708–1713.

Alam, M. M., & Islam, M. T. (2019). Machine learning approach of automatic identification and counting of blood cells. *Healthcare Technology Letters*, *6*(4), 103–108. https://doi.org/10.1049/htl.2018.5098

Albuquerque, C. (2019). Convolutional neural networks for cell detection and counting : a case study of human cell quantification in zebrafish xenografts using deep learning object detection techniques. [Master's thesis, Universidade Nova de Lisboa]. http://hdl.handle.net/10362/62425

Albuquerque, C, Vanneschi, L., Henriques, R., Castelli, M., Póvoa, V., Fior, R., & Papanikolaou, N. (2021). Object detection for automatic cancer cell counting in zebrafish xenografts. *PLoS ONE*, *16*(11 November). https://doi.org/10.1371/journal.pone.0260609

Albuquerque, Carina, Henriques, R., & Castelli, M. (2022). A stacking-based artificial intelligence framework for an effective detection and localization of colon polyps. *Scientific Reports*, *12*(1), 17678.

An, N. S., Lan, P. N., Hang, D. V, Long, D. V, Trung, T. Q., Thuy, N. T., & Sang, D. V. (2022). BlazeNeo: Blazing Fast Polyp Segmentation and Neoplasm Detection. *IEEE Access*, *10*, 43669–43684. https://doi.org/10.1109/ACCESS.2022.3168693

Ariji, Y., Fukuda, M., Nozawa, M., Kuwada, C., Goto, M., Ishibashi, K., Nakayama, A., Sugita, Y., Nagao, T., & Ariji, E. (2021). Automatic detection of cervical lymph nodes in patients with oral squamous cell carcinoma using a deep learning technique: a preliminary study. *Oral Radiology*, *37*(2), 290–296. https://doi.org/10.1007/s11282-020-00449-8

Ariji, Y., Yanashita, Y., Kutsuna, S., Muramatsu, C., Fukuda, M., Kise, Y., Nozawa, M., Kuwada, C., Fujita, H., Katsumata, A., & Ariji, E. (2019). Automatic detection and classification of radiolucent lesions in the mandible on panoramic radiographs using a deep learning object detection technique. *Oral Surgery, Oral Medicine, Oral Pathology and Oral Radiology*, *128*(4), 424–430. https://doi.org/10.1016/j.oooo.2019.05.014

Arnold, M., Morgan, E., Rumgay, H., Mafra, A., Singh, D., Laversanne, M., Vignat, J., Gralow, J. R., Cardoso, F., & Siesling, S. (2022). Current and future burden of breast cancer: Global statistics for 2020 and 2040. *The Breast*, *66*, 15–23.

Ashraf, A. H., Imran, M., Qahtani, A. M., Alsufyani, A., Almutiry, O., Mahmood, A., & Habib, M. (2022). Weapons detection for security and video surveillance using cnn and YOLO-v5s. *CMC-Comput. Mater. Contin*, *70*, 2761–2775.

Aziz, L., Salam, M. S. B. H., Sheikh, U. U., & Ayub, S. (2020). Exploring deep learning-based architecture, strategies, applications and current trends in generic object detection: A comprehensive review. *IEEE Access*, *8*, 170461–170495.

Baccouche, A., Garcia-Zapirain, B., Olea, C. C., & Elmaghraby, A. S. (2021). Breast lesions detection and classification via YOLO-based fusion models. *Computers, Materials and Continua*, *69*(1), 1407–1425. https://doi.org/10.32604/cmc.2021.018461

Bai, B., Du, Y., Liu, P., Sun, P., Li, P., & Lv, Y. (2020). Detection of cervical lesion region from colposcopic images based on feature reselection. *Biomedical Signal Processing and Control*, *57*. https://doi.org/10.1016/j.bspc.2019.101785

Balomenos, A. D., Tsakanikas, P., & Manolakos, E. S. (2015). Tracking single-cells in overcrowded bacterial colonies. *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 6473–6476.

Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y. M. (2020). *YOLOv4: Optimal Speed and Accuracy of Object Detection*. http://arxiv.org/abs/2004.10934

Bodla, N., Singh, B., Chellappa, R., & Davis, L. S. (2017). Soft-NMS--improving object detection with one line of code. *Proceedings of the IEEE International Conference on Computer Vision*, 5561–5569.

Bohr, A., & Memarzadeh, K. (2020). The rise of artificial intelligence in healthcare applications. In *Artificial Intelligence in Healthcare* (Issue January). https://doi.org/10.1016/B978-0-12-818438-7.00002-2

Bria, A., Marrocco, C., & Tortorella, F. (2020). Addressing class imbalance in deep learning for small lesion detection on medical images. *Computers in Biology and Medicine*, *120*. https://doi.org/10.1016/j.compbiomed.2020.103735

Bulmer, M. G. (1979). *Principles of statistics*. Courier Corporation.

Cai, Z., & Vasconcelos, N. (2018). Cascade r-cnn: Delving into high quality object detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 6154–6162.

Cao, C., Wang, B., Zhang, W., Zeng, X., Yan, X., Feng, Z., Liu, Y., & Wu, Z. (2019). An Improved Faster R-CNN for Small Object Detection. *IEEE Access*. https://doi.org/10.1109/ACCESS.2019.2932731

Cao, G., Xie, X., Yang, W., Liao, Q., Shi, G., & Wu, J. (2018). Feature-fused SSD: Fast detection for small objects. *Ninth International Conference on Graphic and Image Processing (ICGIP 2017)*, *10615*, 106151E.

Cao, Z., Duan, L., Yang, G., Yue, T., & Chen, Q. (2019). An experimental study on breast lesion detection and classification from ultrasound images using deep learning architectures. *BMC Medical Imaging*, *19*(1). https://doi.org/10.1186/s12880-019-0349-x

Chang, W.-J., Chen, L.-B., Chen, M.-C., Chiu, Y.-C., & Lin, J.-Y. (2020). ScalpEye: A Deep Learning-Based Scalp Hair Inspection and Diagnosis System for Scalp Health. *IEEE Access*, *8*, 134826–134837. https://doi.org/10.1109/ACCESS.2020.3010847

Chen, A., Li, C., Zou, S., Rahaman, M. M., Yao, Y., Chen, H., Yang, H., Zhao, P., Hu, W., Liu, W., & Grzegorzek, M. (2022). SVIA dataset: A new dataset of microscopic videos and images for computer-aided sperm analysis. *Biocybernetics and Biomedical Engineering*, *42*(1), 204–214. https://doi.org/10.1016/j.bbe.2021.12.010

Chen, H., Li, H., Zhao, Y., Zhao, J., & Wang, Y. (2021). Dental disease detection on periapical radiographs based on deep convolutional neural networks. *International Journal of Computer Assisted Radiology and Surgery*, *16*(4), 649–661. https://doi.org/10.1007/s11548-021-02319-y

Chen, H., Zhang, K., Lyu, P., Li, H., Zhang, L., Wu, J., & Lee, C.-H. (2019). A deep learning approach to automatic teeth detection and numbering based on object detection in dental periapical films. *Scientific Reports*, *9*(1). https://doi.org/10.1038/s41598-019-40414-y

Chen, W., Huang, H., Peng, S., Zhou, C., & Zhang, C. (2021). YOLO-face: a real-time face detector. *The Visual Computer*, *37*, 805–813.

Cho, J., Lee, K., Shin, E., Choy, G., & Do, S. (2015). How much data is needed to train a medical image deep learning system to achieve necessary high accuracy? *ArXiv Preprint ArXiv:1511.06348*.

Chowdhary, C. L., & Acharjya, D. P. (2020). Segmentation and feature extraction in medical imaging: a systematic review. *Procedia Computer Science*, *167*, 26–36.

Costa, B., Estrada, M. F., Mendes, R. V, & Fior, R. (2020). Zebrafish avatars towards personalized medicine—A comparative review between avatar models. *Cells*, *9*(2), 293.

Cui, L., Ma, R., Lv, P., Jiang, X., Gao, Z., Zhou, B., & Xu, M. (2020). MDSSD: multi-scale deconvolutional single shot detector for small objects. In *Science China Information Sciences*. https://doi.org/10.1007/s11432-019-2723-1

Dai, J., Li, Y., He, K., & Sun, J. (2016a). R-FCN: Object detection via region-based fully convolutional networks. *Advances in Neural Information Processing Systems*, 379–387.

Dai, J., Li, Y., He, K., & Sun, J. (2016b). R-FCN: Object detection via region-based fully convolutional networks. *Advances in Neural Information Processing Systems*.

Dalal, N., & Triggs, B. (2005). Histograms of oriented gradients for human detection. *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, *1*, 886–893.

Dao, D. (2020). *Tensorflow 2 Detection Model Zoo*. https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/tf2_detection_zoo.md

de Almeida, C. R., Mendes, R. V., Pezzarossa, A., Gago, J., Carvalho, C., Alves, A., Nunes, V., Brito, M. J., Cardoso, M. J., & Ribeiro, J. (2020). Zebrafish xenografts as a fast screening platform for bevacizumab cancer therapy. *Communications Biology*, *3*(1), 1–13.

De Bellis, N. (2009). *Bibliometrics and citation analysis: from the science citation index to cybermetrics*. scarecrow press.

De Boodt, S., Poursaberi, A., Schrooten, J., Berckmans, D., & Aerts, J. M. (2013). A semiautomatic cell counting tool for quantitative imaging of tissue engineering scaffolds. *Tissue Engineering - Part C: Methods*. https://doi.org/10.1089/ten.tec.2012.0486

Ding, S., Li, L., Li, Z., Wang, H., & Zhang, Y. (2019). Smart electronic gastroscope system using a cloud–edge collaborative framework. *Future Generation Computer Systems*, *100*, 395–407. https://doi.org/10.1016/j.future.2019.04.031

Dong, B., Shao, L., Da Costa, M., Bandmann, O., & Frangi, A. F. (2015). Deep learning for automatic cell detection in wide-field microscopy zebrafish images. *Proceedings - International Symposium on Biomedical Imaging*. https://doi.org/10.1109/ISBI.2015.7163986

Dong, J., Liu, S., Liao, Y., Wen, H., Lei, B., Li, S., & Wang, T. (2020). A Generic Quality Control Framework for Fetal Ultrasound Cardiac Four-Chamber Planes. *IEEE Journal of Biomedical and Health Informatics*, *24*(4), 931–942. https://doi.org/10.1109/JBHI.2019.2948316

Donthu, N., Kumar, S., Mukherjee, D., Pandey, N., & Lim, W. M. (2021). How to conduct a bibliometric analysis: An overview and guidelines. *Journal of Business Research*, *133*, 285–296. https://doi.org/https://doi.org/10.1016/j.jbusres.2021.04.070

Doubeni, C. A., Corley, D. A., Quinn, V. P., Jensen, C. D., Zauber, A. G., Goodman, M., Johnson, J. R., Mehta, S. J., Becerra, T. A., Zhao, W. K., Schottinger, J., Doria-Rose, V. P., Levin, T. R., Weiss, N. S., & Fletcher, R. H. (2018). Effectiveness of screening colonoscopy in reducing the risk of death from right and left colon cancer: A large community-based study. *Gut*, *67*(2), 291–298. https://doi.org/10.1136/gutjnl-2016-312712

Doubeni, C. A., Weinmann, S., Adams, K., Kamineni, A., Buist, D. S. M., Ash, A. S., Rutter, C. M., Doria-Rose, V. P., Corley, D. A., Greenlee, R. T., Chubak, J., Williams, A., Kroll-Desrosiers, A. R., Johnson, E., Webster, J., Richert-Boe, K., Levin, T. R., Fletcher, R. H., & Weiss, N. S. (2013). Screening colonoscopy and risk for incident late-stage colorectal cancer diagnosis in average-risk adults: a nested case-control study. *Annals of Internal Medicine*, *158*(5 Pt 1), 312–320. https://doi.org/10.7326/0003-4819-158-5-201303050-00003

Duan, H., Huang, Y., Liu, L., Dai, H., Chen, L., & Zhou, L. (2019). Automatic detection on intracranial aneurysm from digital subtraction angiography with cascade convolutional neural networks. *BioMedical Engineering Online*, *18*(1). https://doi.org/10.1186/s12938-019-0726-2

Duan, K., Bai, S., Xie, L., Qi, H., Huang, Q., & Tian, Q. (2019). Centernet: Keypoint triplets for object detection. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 6569–6578.

Egger, J., Gsaxner, C., Pepe, A., Pomykala, K. L., Jonske, F., Kurz, M., Li, J., & Kleesiek, J. (2022). Medical Deep Learning–A systematic Meta-Review. *Computer Methods and Programs in Biomedicine*, 106874.

Eggert, C., Brehm, S., Winschel, A., Zecha, D., & Lienhart, R. (2017). A closer look: Small object detection in faster R-CNN. *Proceedings - IEEE International Conference on Multimedia and Expo*. https://doi.org/10.1109/ICME.2017.8019550

Elakkiya, R., Teja, K. S. S., Jegatha Deborah, L., Bisogni, C., & Medaglia, C. (2022). Imaging based cervical cancer diagnostics using small object detection - generative adversarial networks. *Multimedia Tools and Applications*, *81*(1), 191–207. https://doi.org/10.1007/s11042-021-10627-3

Estai, M., Tennant, M., Gebauer, D., Brostek, A., Vignarajan, J., Mehdizadeh, M., & Saha, S. (2022). Deep learning for automated detection and numbering of permanent teeth on panoramic images. *Dentomaxillofacial Radiology*, *51*(2). https://doi.org/10.1259/dmfr.20210296

Esteva, A., Chou, K., Yeung, S., Naik, N., Madani, A., Mottaghi, A., Liu, Y., Topol, E., Dean, J., & Socher, R. (2021). Deep learning-enabled medical computer vision. *Npj Digital Medicine*, *4*(1), 1–9. https://doi.org/10.1038/s41746-020-00376-2

Falk, T., Mai, D., Bensch, R., Çiçek, Ö., Abdulkadir, A., Marrakchi, Y., Böhm, A., Deubner, J., Jäckel, Z., Seiwald, K., Dovzhenko, A., Tietz, O., Dal Bosco, C., Walsh, S., Saltukoglu, D., Tay, T. L., Prinz, M., Palme, K., Simons, M., … Ronneberger, O. (2019). U-Net: deep learning for cell counting, detection, and morphometry. *Nature Methods*. https://doi.org/10.1038/s41592-018-0261-2

Fazio, M., Ablain, J., Chuan, Y., Langenau, D. M., & Zon, L. I. (2020). Zebrafish patient avatars in cancer biology and precision cancer therapy. *Nature Reviews Cancer*, *20*(5), 263–273.

Fazio, M., & Zon, L. I. (2017). Fishing for answers in precision cancer medicine. *Proceedings of the National Academy of Sciences*, *114*(39), 10306–10308.

Felzenszwalb, P., McAllester, D., & Ramanan, D. (2008). A discriminatively trained, multiscale, deformable part model. *2008 IEEE Conference on Computer Vision and Pattern Recognition*, 1–8.

Fior, R., Póvoa, V., Mendes, R. V., Carvalho, T., Gomes, A., Figueiredo, N., & Ferreira, F. R. (2017). Single-cell functional and chemosensitive profiling of combinatorial colorectal therapy in zebrafish xenografts. *Proceedings of the National Academy of Sciences of the United States of America*. https://doi.org/10.1073/pnas.1618389114

Fu, C. Y., Liu, W., Ranga, A., Tyagi, A., & Berg, A. C. (2017). DSSD: Deconvolutional single shot detector. In *arXiv*.

Fujiyoshi, H., Hirakawa, T., & Yamashita, T. (2019). Deep learning-based image recognition for autonomous driving. *IATSS Research*, *43*(4), 244–252.

Fukuda, M., Inamoto, K., Shibata, N., Ariji, Y., Yanashita, Y., Kutsuna, S., Nakata, K., Katsumata, A.,

Fujita, H., & Ariji, E. (2020). Evaluation of an artificial intelligence system for detecting vertical root fracture on panoramic radiography. *Oral Radiology*, *36*(4), 337–343. https://doi.org/10.1007/s11282-019-00409-x

Gan, K., Xu, D., Lin, Y., Shen, Y., Zhang, T., Hu, K., Zhou, K., Bi, M., Pan, L., Wu, W., & Liu, Y. (2019). Artificial intelligence detection of distal radius fractures: a comparison between the convolutional neural network and professional assessments. *Acta Orthopaedica*, *90*(4), 394–400. https://doi.org/10.1080/17453674.2019.1600125

Ghatwary, N., Zolgharni, M., & Ye, X. (2019). Early esophageal adenocarcinoma detection using deep learning methods. *International Journal of Computer Assisted Radiology and Surgery*, *14*(4), 611–621. https://doi.org/10.1007/s11548-019-01914-4

Girshick, R. (2015a). Fast R-CNN. *Proceedings of the IEEE International Conference on Computer Vision*. https://doi.org/10.1109/ICCV.2015.169

Girshick, R. (2015b). Fast R-CNN. *Proceedings of the IEEE International Conference on Computer Vision*, *2015 Inter*, 1440–1448. https://doi.org/10.1109/ICCV.2015.169

Girshick, R. B. (2012). *From rigid templates to grammars: Object detection with structured models*. The University of Chicago.

Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. https://doi.org/doi: 10.1109/CVPR.2014.81

Gómez, D., & Rojas, A. (2016). An empirical overview of the no free lunch theorem and its effect on real-world machine learning classification. *Neural Computation*, *28*(1), 216–228.

Gonzalez, R. C., & Woods, R. E. (2018). *Digital image processing 4th Edition*. Pearson.

Guan, T., & Zhu, H. (2017a). Atrous Faster R-CNN for Small Scale Object Detection. *Proceedings - 2017 2nd International Conference on Multimedia and Image Processing, ICMIP 2017*. https://doi.org/10.1109/ICMIP.2017.37

Guan, T., & Zhu, H. (2017b). Atrous faster R-CNN for small scale object detection. *2017 2nd International Conference on Multimedia and Image Processing (ICMIP)*, 16–21.

Guo, Yue, Stein, J., Wu, G., & Krishnamurthy, A. (2019). Sau-net: A universal deep network for cell counting. *Proceedings of the 10th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics*, 299–306.

Guo, Yuqi, Hao, Z., Zhao, S., Gong, J., & Yang, F. (2020). Artificial intelligence in health care: bibliometric analysis. *Journal of Medical Internet Research*, *22*(7), e18228.

Guo, Z., Zhang, R., Li, Q., Liu, X., Nemoto, D., Togashi, K., Niroshana, S. M. I., Shi, Y., & Zhu, X. (2020). Reduce False-Positive Rate by Active Learning for Automatic Polyp Detection in Colonoscopy Videos. *2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI)*, 1655–1658. https://doi.org/10.1109/ISBI45749.2020.9098500

Habuza, T., Navaz, A. N., Hashim, F., Alnajjar, F., Zaki, N., Serhani, M. A., & Statsenko, Y. (2021). AI applications in robotics, diagnostic image analysis and precision medicine: Current limitations,

future trends, guidelines on CAD systems for medicine. *Informatics in Medicine Unlocked*, *24*, 100596.

Hadi Hosseini, S. M., Chen, H., & Jablonski, M. M. (2020). Automatic detection and counting of retina cell nuclei using deep learning. In *arXiv*. https://doi.org/10.1117/12.2567454

Hardalaç, F., Uysal, F., Peker, O., Çiçeklidağ, M., Tolunay, T., Tokgöz, N., Kutbay, U., Demirciler, B., & Mert, F. (2022). Fracture Detection in Wrist X-ray Images Using Deep Learning-Based Object Detection Models. *Sensors*, *22*(3). https://doi.org/10.3390/s22031285

Hashimoto, R., Requa, J., Dao, T., Ninh, A., Tran, E., Mai, D., Lugo, M., El-Hage Chehade, N., Chang, K. J., Karnes, W. E., & Samarasena, J. B. (2020). Artificial intelligence using convolutional neural networks for real-time detection of early esophageal neoplasia in Barrett's esophagus (with video). *Gastrointestinal Endoscopy*, *91*(6), 1264-1271.e1. https://doi.org/10.1016/j.gie.2019.12.049

He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask r-cnn. *Proceedings of the IEEE International Conference on Computer Vision*, 2961–2969.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 770–778.

He, S., Minn, K. T., Solnica-Krezel, L., Anastasio, M. A., & Li, H. (2021). Deeply-supervised density regression for automatic cell counting in microscopy images. *Medical Image Analysis*, *68*, 101892. https://doi.org/10.1016/j.media.2020.101892

Ho, Y.-C., & Pepyne, D. L. (2002). Simple explanation of the no-free-lunch theorem and its implications. *Journal of Optimization Theory and Applications*, *115*, 549–570.

Honga, A., Leeb, G., Leec, H., Seod, J., & Yeoe, D. (2021). Deep Learning Model Generalization with Ensemble in Endoscopic Images. *Proceedings of the 3rd International Workshop and Challenge on Computer Vision in Endoscopy (EndoCV 2021) Co-Located with with the 18th IEEEInternational Symposium on Biomedical Imaging (ISBI 2021), Nice, France*, *2886*, 80–89.

Hossain, M. B., Nishikawa, R. M., & Lee, J. (2022). Developing breast lesion detection algorithms for digital breast tomosynthesis: Leveraging false positive findings. *Medical Physics*, *49*(12), 7596–7608. https://doi.org/10.1002/mp.15883

Howard, J., & Gugger, S. (2020). Fastai: a layered API for deep learning. *Information*, *11*(2), 108.

Hu, G. X., Yang, Z., Hu, L., Huang, L., & Han, J. M. (2018). Small Object Detection with Multiscale Features. *International Journal of Digital Multimedia Broadcasting*. https://doi.org/10.1155/2018/4546896

Hu, Y., Yu, Z., Cheng, X., Luo, Y., & Wen, C. (2020). A bibliometric analysis and visualization of medical data mining research. *Medicine*, *99*(22).

Huang, J., Rathod, V., Sun, C., Zhu, M., Korattikara, A., Fathi, A., Fischer, I., Wojna, Z., Song, Y., & Guadarrama, S. (2017). Speed/accuracy trade-offs for modern convolutional object detectors. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 7310–7311.

Huang, Q., Luo, H., Yang, C., Li, J., Deng, Q., Liu, P., Fu, M., Li, L., & Li, X. (2022). Anatomical prior

based vertebra modelling for reappearance of human spines. *Neurocomputing*, *500*, 750–760. https://doi.org/10.1016/j.neucom.2022.05.033

Hung, J, Goodman, A., Ravel, D., Lopes, S. C. P., Rangel, G. W., Nery, O. A., Malleret, B., Nosten, F., Lacerda, M. V. G., Ferreira, M. U., Rénia, L., Duraisingh, M. T., Costa, F. T. M., Marti, M., & Carpenter, A. E. (2020). Keras R-CNN: Library for cell detection in biological images using deep neural networks. *BMC Bioinformatics*, *21*(1). https://doi.org/10.1186/s12859-020-03635-x

Hung, Jane, Ravel, D., Lopes, S. C. P., Rangel, G., Malleret, B., Nosten, F., Lacerda, M. V. G., Ferreira, M. U., Rénia, L., Duraisingh, M. T., Costa, F. T. M., Marti, M., & Carpenter, A. E. (2018). Applying faster R-CNN for object detection on malaria images. In *arXiv*.

Issa, I. A., & NouredDine, M. (2017). Colorectal cancer screening: An updated review of the available options. *World Journal of Gastroenterology*, *23*(28), 5086–5096. https://doi.org/10.3748/wjg.v23.i28.5086

Jha, D., Ali, S., Tomar, N. K., Johansen, H. D., Johansen, D., Rittscher, J., Riegler, M. A., & Halvorsen, P. (2021). Real-Time Polyp Detection, Localization and Segmentation in Colonoscopy Using Deep Learning. *IEEE Access*, *9*(March), 40496–40510. https://doi.org/10.1109/ACCESS.2021.3063716

Jiang, H., & Learned-Miller, E. (2017). Face detection with the faster R-CNN. *2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*, 650–657.

Jiang, Z., Liu, X., Yan, Z., Gu, W., & Jiang, J. (2021). Improved detection performance in blood cell count by an attention-guided deep learning method. *OSA Continuum*, *4*(2), 323–333.

Jocher, G., Stoken, A., Borovec, J., NanoCode012, Chaurasia, A., TaoXie, Changyu, L., V, A., Laughing, tkianai, yxNONG, Hogan, A., lorenzomammana, AlexWang1900, Hajek, J., Diaconu, L., Marc, Kwon, Y., oleg, … Ingham, F. (2021). *ultralytics/yolov5: v5.0 - YOLOv5-P6 1280 models, AWS, Supervise.ly and YouTube integrations*. https://doi.org/10.5281/ZENODO.4679653

Jojoa Acosta, M. F., Caballero Tovar, L. Y., Garcia-Zapirain, M. B., & Percybrooks, W. S. (2021). Melanoma diagnosis using deep learning techniques on dermatoscopic images. *BMC Medical Imaging*, *21*(1). https://doi.org/10.1186/s12880-020-00534-8

Ju, L., Wang, X., Zhao, X., Bonnington, P., Drummond, T., & Ge, Z. (2021). Leveraging Regular Fundus Images for Training UWF Fundus Diagnosis Models via Adversarial Learning and Pseudo-Labeling. *IEEE Transactions on Medical Imaging*, *40*(10), 2911–2925. https://doi.org/10.1109/TMI.2021.3056395

Jung, H., Kim, B., Lee, I., Yoo, M., Lee, J., Ham, S., Woo, O., & Kang, J. (2018). Detection of masses in mammograms using a one-stage object detector based on a deep convolutional neural network. *PLoS ONE*, *13*(9). https://doi.org/10.1371/journal.pone.0203355

Kalpathy-Cramer, J., de Herrera, A. G. S., Demner-Fushman, D., Antani, S., Bedrick, S., & Müller, H. (2015). Evaluating performance of biomedical image retrieval systems—An overview of the medical image retrieval task at ImageCLEF 2004–2013. *Computerized Medical Imaging and Graphics*, *39*, 55–61.

Kassis, T., Hernandez-Gordillo, V., Langer, R., & Griffith, L. G. (2019). OrgaQuant: Human Intestinal Organoid Localization and Quantification Using Deep Convolutional Neural Networks. *Scientific Reports*, *9*(1), 1–7. https://doi.org/10.1038/s41598-019-48874-y

Kaur, A., Singh, Y., Neeru, N., Kaur, L., & Singh, A. (2021). A survey on deep learning approaches to medical images and a systematic look up into real-time object detection. *Archives of Computational Methods in Engineering*, 1–41.

Kawazoe, Y., Shimamoto, K., Yamaguchi, R., Shintani-Domoto, Y., Uozaki, H., Fukayama, M., & Ohe, K. (2018). Faster R-CNN-based glomerular detection in multistained human whole slide images. *Journal of Imaging*, *4*(7). https://doi.org/10.3390/jimaging4070091

Khan, A. I., & Al-Habsi, S. (2020). Machine learning in computer vision. *Procedia Computer Science*, *167*, 1444–1451.

Khandekar, R., Shastry, P., Jaishankar, S., Faust, O., & Sampathila, N. (2021). Automated blast cell detection for Acute Lymphoblastic Leukemia diagnosis. *Biomedical Signal Processing and Control*, *68*. https://doi.org/10.1016/j.bspc.2021.102690

Kim, K., Kim, S., Lee, Y. H., Lee, S. H., Lee, H. S., & Kim, S. (2018). Performance of the deep convolutional neural network based magnetic resonance image scoring algorithm for differentiating between tuberculous and pyogenic spondylitis. *Scientific Reports*, *8*(1). https://doi.org/10.1038/s41598-018-31486-3

Kim, N. H., Jung, Y. S., Jeong, W. S., Yang, H.-J., Park, S.-K., Choi, K., & Park, D. Il. (2017). Miss rate of colorectal neoplastic polyps and risk factors for missed polyps in consecutive colonoscopies. *Intestinal Research*, *15*(3), 411–418. https://doi.org/10.5217/ir.2017.15.3.411

Koga, S., Ikeda, A., & Dickson, D. W. (2022). Deep learning-based model for diagnosing Alzheimer's disease and tauopathies. *Neuropathology and Applied Neurobiology*, *48*(1). https://doi.org/10.1111/nan.12759

Komatsu, M., Sakai, A., Komatsu, R., Matsuoka, R., Yasutomi, S., Shozu, K., Dozen, A., Machino, H., Hidaka, H., Arakaki, T., Asada, K., Kaneko, S., Sekizawa, A., & Hamamoto, R. (2021). Detection of cardiac structural abnormalities in fetal ultrasound videos using deep learning. *Applied Sciences (Switzerland)*, *11*(1), 1–12. https://doi.org/10.3390/app11010371

Kong, Y., Li, H., Ren, Y., Genchev, G. Z., Wang, X., Zhao, H., Xie, Z., & Lu, H. (2020). Automated yeast cells segmentation and counting using a parallel U-Net based two-stage framework. *OSA Continuum*, *3*(4), 982–992.

Korfhage, N., Mühling, M., Ringshandl, S., Becker, A., Schmeck, B., & Freisleben, B. (2020). Detection and segmentation of morphologically complex eukaryotic cells in fluorescence microscopy images via feature pyramid fusion. *PLOS Computational Biology*, *16*(9), e1008179.

Kumar, A., Kim, J., Lyndon, D., Fulham, M., & Feng, D. (2016). An ensemble of fine-tuned convolutional neural networks for medical image classification. *IEEE Journal of Biomedical and Health Informatics*, *21*(1), 31–40.

Kutlu, H., Avci, E., & Özyurt, F. (2020). White blood cells detection and classification based on regional convolutional neural networks. *Medical Hypotheses*, *135*. https://doi.org/10.1016/j.mehy.2019.109472

Kuwana, R., Ariji, Y., Fukuda, M., Kise, Y., Nozawa, M., Kuwada, C., Muramatsu, C., Katsumata, A., Fujita, H., & Ariji, E. (2020). Performance of deep learning object detection technology in the detection and diagnosis of maxillary sinus lesions on panoramic radiographs. *Dentomaxillofacial*

Radiology, *50*(1). https://doi.org/10.1259/dmfr.20200171

Lan, L., Ye, C., Wang, C., & Zhou, S. (2019). Deep Convolutional Neural Networks for WCE Abnormality Detection: CNN Architecture, Region Proposal and Transfer Learning. *IEEE Access*, *7*, 30017–30032. https://doi.org/10.1109/ACCESS.2019.2901568

Leufkens, A. M., van Oijen, M. G. H., Vleggaar, F. P., & Siersema, P. D. (2012). Factors influencing the miss rate of polyps in a back-to-back colonoscopy study. *Endoscopy*, *44*(5), 470–475. https://doi.org/10.1055/s-0031-1291666

Li, H., Weng, J., Shi, Y., Gu, W., Mao, Y., Wang, Y., Liu, W., & Zhang, J. (2018). An improved deep learning approach for detection of thyroid papillary cancer in ultrasound images. *Scientific Reports*, *8*(1). https://doi.org/10.1038/s41598-018-25005-7

Li, J., Zhu, G., Hua, C., Feng, M., Li, P., Lu, X., Song, J., Shen, P., Xu, X., & Mei, L. (2021). A systematic collection of medical image datasets for deep learning. *ArXiv Preprint ArXiv:2106.12864*.

Li, K., Fathan, M. I., Patel, K., Zhang, T., Zhong, C., Bansal, A., Rastogi, A., Wang, J. S., & Wang, G. (2021). Colonoscopy polyp detection and classification: Dataset creation and comparative evaluations. *PLoS ONE*, *16*(8 August). https://doi.org/10.1371/journal.pone.0255809

Li, W., Yang, C., Liu, J., Liu, X., Guo, X., & Yuan, Y. (2021). Joint polyp detection and segmentation with heterogeneous endoscopic data. *CEUR Workshop Proceedings*, *2886*, 69–79.

Li, X., Zhang, Y., Cui, Q., Yi, X., & Zhang, Y. (2019). Tooth-Marked Tongue Recognition Using Multiple Instance Learning and CNN Features. *IEEE Transactions on Cybernetics*, *49*(2), 380–387. https://doi.org/10.1109/TCYB.2017.2772289Y

Li, Y.-C., Chen, H.-H., Horng-Shing Lu, H., Hondar Wu, H.-T., Chang, M.-C., & Chou, P.-H. (2021). Can a Deep-learning Model for the Automated Detection of Vertebral Fractures Approach the Performance Level of Human Subspecialists? *Clinical Orthopaedics and Related Research*, *479*(7), 1598–1612. https://doi.org/10.1097/CORR.0000000000001685

Li, Z., Peng, C., Yu, G., Zhang, X., Deng, Y., & Sun, J. (2018). Detnet: A backbone network for object detection. *ArXiv Preprint ArXiv:1804.06215*.

Liang, B., Zhai, Y., Tong, C., Zhao, J., Li, J., He, X., & Ma, Q. (2019). A deep automated skeletal bone age assessment model via region-based convolutional neural network. *Future Generation Computer Systems*, *98*, 54–59. https://doi.org/10.1016/j.future.2019.01.057

Lin, T.-Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2017). Focal loss for dense object detection. *Proceedings of the IEEE International Conference on Computer Vision*, 2980–2988.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. L. (2014). Microsoft coco: Common objects in context. *European Conference on Computer Vision*, 740–755.

Lin, T. Y., Goyal, P., Girshick, R., He, K., & Dollar, P. (2020). Focal Loss for Dense Object Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *42*(2), 318–327. https://doi.org/10.1109/TPAMI.2018.2858826

Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., Van Der Laak, J. A., Van

Ginneken, B., & Sánchez, C. I. (2017). A survey on deep learning in medical image analysis. *Medical Image Analysis*, *42*, 60–88.

Liu, M., Wang, S., Chen, H., & Liu, Y. (2022). A pilot study of a deep learning approach to detect marginal bone loss around implants. *BMC Oral Health*, *22*(1). https://doi.org/10.1186/s12903-021-02035-8

Liu, Q., Junker, A., Murakami, K., & Hu, P. (2019). A novel convolutional regression network for cell counting. *2019 IEEE 7th International Conference on Bioinformatics and Computational Biology (ICBCB)*, 44–49.

Liu, S., Qi, L., Qin, H., Shi, J., & Jia, J. (2018). Path Aggregation Network for Instance Segmentation. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 8759–8768. https://doi.org/10.1109/CVPR.2018.00913

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., & Berg, A. C. (2016a). Ssd: Single shot multibox detector. *European Conference on Computer Vision*, 21–37.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2016b). SSD: Single shot multibox detector. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. https://doi.org/10.1007/978-3-319-46448-0_2

Liu, Y, Zhang, F., Chen, C., Wang, S., Wang, Y., & Yu, Y. (2022). Act Like a Radiologist: Towards Reliable Multi-View Correspondence Reasoning for Mammogram Mass Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *44*(10), 5947–5961. https://doi.org/10.1109/TPAMI.2021.3085783

Liu, Yang, Sun, P., Wergeles, N., & Shang, Y. (2021). A survey and performance evaluation of deep learning methods for small object detection. *Expert Systems with Applications*, *172*, 114602.

Loh, D. R., Yong, W. X., Yapeter, J., Subburaj, K., & Chandramohanadas, R. (2021). A deep learning approach to the screening of malaria infection: Automated and rapid cell counting, object detection and instance segmentation using Mask R-CNN. *Computerized Medical Imaging and Graphics*, *88*, 101845.

Maeng, L.-S., Kim, J., Ko, B. M., Eun, C., Park, C., Baek, I., Jeen, Y., Shin, J., Han, D., Park, D., Lee, S., & Park, D. (2007). Adequate Level of Training for Technical Competence in Colonoscopy: A Prospective Multicenter Evaluation of the Learning Curve. *Gastrointestinal Endoscopy - GASTROINTEST ENDOSCOP*, *65*. https://doi.org/10.1016/j.gie.2007.03.045

Mahajan, S., Raina, A., Gao, X.-Z., & Pandit, A. K. (2022). COVID-19 detection using hybrid deep learning model in chest x-rays images. *Concurrency and Computation: Practice and Experience*, *34*(5). https://doi.org/10.1002/cpe.6747

Mahmood, T., Arsalan, M., Owais, M., Lee, M. B., & Park, K. R. (2020). Artificial intelligence-based mitosis detection in breast cancer histopathology images using faster R-CNN and deep CNNs. *Journal of Clinical Medicine*, *9*(3), 749.

Manescu, P., Shaw, M. J., Elmi, M., Neary-Zajiczek, L., Claveau, R., Pawar, V., Kokkinos, I., Oyinloye, G., Bendkowski, C., Oladejo, O. A., Oladejo, B. F., Clark, T., Timm, D., Shawe-Taylor, J., Srinivasan, M. A., Lagunju, I., Sodeinde, O., Brown, B. J., & Fernandez-Reyes, D. (2020). Expert-

level automated malaria diagnosis on routine blood films with deep neural networks. *American Journal of Hematology*, *95*(8), 883–891. https://doi.org/10.1002/ajh.25827

Marzahl, C., Aubreville, M., Bertram, C. A., Stayt, J., Jasensky, A.-K., Bartenschlager, F., Fragoso-Garcia, M., Barton, A. K., Elsemann, S., Jabari, S., Krauth, J., Madhu, P., Voigt, J., Hill, J., Klopfleisch, R., & Maier, A. (2020). Deep Learning-Based Quantification of Pulmonary Hemosiderophages in Cytology Slides. *Scientific Reports*, *10*(1). https://doi.org/10.1038/s41598-020-65958-2

McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, *5*, 115–133.

Mochalova, E. N., Kotov, I. A., Rozenberg, J. M., & Nikitin, M. P. (2020). Precise Quantitative Analysis of Cell Targeting by Particle-Based Agents Using Imaging Flow Cytometry and Convolutional Neural Network. *Cytometry Part A*, *97*(3), 279–287. https://doi.org/10.1002/cyto.a.23939

Montalbo, F. J. P. (2020). A computer-aided diagnosis of brain tumors using a fine-tuned yolo-based model with transfer learning. *KSII Transactions on Internet and Information Systems*, *14*(12), 4816–4834. https://doi.org/10.3837/tiis.2020.12.011

Moreira, I. C., Amaral, I., Domingues, I., Cardoso, A., Cardoso, M. J., & Cardoso, J. S. (2012). Inbreast: toward a full-field digital mammographic database. *Academic Radiology*, *19*(2), 236–248.

Muramatsu, C., Morishita, T., Takahashi, R., Hayashi, T., Nishiyama, W., Ariji, Y., Zhou, X., Hara, T., Katsumata, A., Ariji, E., & Fujita, H. (2021). Tooth detection and classification on panoramic radiographs for automatic dental chart filing: improved classification by multi-sized input data. *Oral Radiology*, *37*(1), 13–19. https://doi.org/10.1007/s11282-019-00418-w

Nazir, T., Irtaza, A., Javed, A., Malik, H., Hussain, D., & Naqvi, R. A. (2020). Retinal image analysis for diabetes-based eye disease detection using deep learning. *Applied Sciences (Switzerland)*, *10*(18). https://doi.org/10.3390/APP10186185

Neubeck, A. (2006). *Efficient Non-Maximum Suppression*. 0–5.

Ngoc Lan, P., An, N. S., Hang, D. V., Long, D. Van, Trung, T. Q., Thuy, N. T., & Sang, D. V. (2021). NeoUNet : Towards Accurate Colon Polyp Segmentation and Neoplasm Detection. In G. Bebis, V. Athitsos, T. Yan, M. Lau, F. Li, C. Shi, X. Yuan, C. Mousas, & G. Bruder (Eds.), *Advances in Visual Computing* (pp. 15–28). Springer International Publishing.

Organization, W. H. (2020). Latest global cancer data: Cancer burden rises to 19.3 million new cases and 10.0 million cancer deaths in 2020. In *2020-12-15)[2021-12-21]. https://www. iarc. who. int/news-events/latest-global-cancer-data-cancer-burden-rises-to-19-3-million-new-cases-and-10-0-million-cancer-deaths-in-2020*.

Ozawa, T., Ishihara, S., Fujishiro, M., Kumagai, Y., Shichijo, S., & Tada, T. (2020). Automated endoscopic detection and classification of colorectal polyps using convolutional neural networks. *Therapeutic Advances in Gastroenterology*, *13*. https://doi.org/10.1177/1756284820910659

Ozturk, T., Talo, M., Yildirim, E. A., Baloglu, U. B., Yildirim, O., & Rajendra Acharya, U. (2020). Automated detection of COVID-19 cases using deep neural networks with X-ray images. *Computers in Biology and Medicine*, *121*. https://doi.org/10.1016/j.compbiomed.2020.103792

Pacal, I, Karaman, A., Karaboga, D., Akay, B., Basturk, A., Nalbantoglu, U., & Coskun, S. (2022). An efficient real-time colonic polyp detection with YOLO algorithms trained by using negative samples and large datasets. *Computers in Biology and Medicine*, *141*. https://doi.org/10.1016/j.compbiomed.2021.105031

Pacal, Ishak, & Karaboga, D. (2021). A robust real-time deep learning based automatic polyp detection system. *Computers in Biology and Medicine*, *134*, 104519. https://doi.org/https://doi.org/10.1016/j.compbiomed.2021.104519

Padilla, R., Passos, W. L., Dias, T. L. B., Netto, S. L., & Da Silva, E. A. B. (2021). A comparative analysis of object detection metrics with a companion open-source toolkit. *Electronics (Switzerland)*, *10*(3), 1–28. https://doi.org/10.3390/electronics10030279

Pandey, M., Fernandez, M., Gentile, F., Isayev, O., Tropsha, A., Stern, A. C., & Cherkasov, A. (2022). The transformational role of GPU computing and deep learning in drug discovery. *Nature Machine Intelligence*, *4*(3), 211–221.

Pang, S., Ding, T., Qiao, S., Meng, F., Wang, S., Li, P., & Wang, X. (2019). A novel YOLOv3-arch model for identifying cholelithiasis and classifying gallstones on CT images. *PLoS ONE*, *14*(6). https://doi.org/10.1371/journal.pone.0217647

Pham, D. L., Xu, C., & Prince, J. L. (2000). Current Methods in Medical Image Segmentation. *Annual Review of Biomedical Engineering*, *2*(1), 315–337. https://doi.org/10.1146/annurev.bioeng.2.1.315

Polat, G., Isik-polat, E., Kayabay, K., & Temizel, A. (2020). *Polyp Detection in Colonoscopy Images using Deep Learning and Bootstrap Aggregation*.

Prince, S. J. D. (2012). *Computer vision: models, learning, and inference*. Cambridge University Press.

PUB, M. H., Bowyer, K., Kopans, D., Moore, R., & Kegelmeyer, P. (1996). The digital database for screening mammography. *Proceedings of the Third International Workshop on Digital Mammography, Chicago, IL, USA*, 9–12.

Puttagunta, M., & Ravi, S. (2021). Medical image analysis based on deep learning approach. *Multimedia Tools and Applications*, *80*, 24365–24398.

Qadir, H. A., Shin, Y., Solhusvik, J., Bergsland, J., Aabakken, L., & Balasingham, I. (2021). Toward real-time polyp detection using fully CNNs for 2D Gaussian shapes prediction. *Medical Image Analysis*, *68*, 101897. https://doi.org/10.1016/j.media.2020.101897

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, *2016-Decem*, 779–788. https://doi.org/10.1109/CVPR.2016.91

Redmon, J., & Farhadi, A. (2017). YOLO9000: Better, faster, stronger. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, *2017-Janua*, 6517–6525. https://doi.org/10.1109/CVPR.2017.690

Redmon, J., & Farhadi, A. (2018). *YOLOv3: An Incremental Improvement*. http://arxiv.org/abs/1804.02767

Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in Neural Information Processing Systems*, *28*.

Ren, S., He, K., Girshick, R., & Sun, J. (2017). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. https://doi.org/10.1109/TPAMI.2016.2577031

Ribli, D., Horváth, A., Unger, Z., Pollner, P., & Csabai, I. (2018). Detecting and classifying lesions in mammograms with Deep Learning. *Scientific Reports*, *8*(1). https://doi.org/10.1038/s41598-018-22437-z

Ritter, F., Boskamp, T., Homeyer, A., Laue, H., Schwier, M., Link, F., & Peitgen, H.-O. (2011). Medical image analysis. *IEEE Pulse*, *2*(6), 60–70.

Rosati, R., Romeo, L., Silvestri, S., Marcheggiani, F., Tiano, L., & Frontoni, E. (2020). Faster R-CNN approach for detection and quantification of DNA damage in comet assay images. *Computers in Biology and Medicine*, *123*. https://doi.org/10.1016/j.compbiomed.2020.103912

Rouzrokh, P., Ramazanian, T., Wyles, C. C., Philbrick, K. A., Cai, J. C., Taunton, M. J., Maradit Kremers, H., Lewallen, D. G., & Erickson, B. J. (2021). Deep Learning Artificial Intelligence Model for Assessment of Hip Dislocation Risk Following Primary Total Hip Arthroplasty From Postoperative Radiographs. *Journal of Arthroplasty*, *36*(6), 2197-2203.e3. https://doi.org/10.1016/j.arth.2021.02.028

Ruiz-Rosero, J., Ramírez-González, G., & Viveros-Delgado, J. (2019). Software survey: ScientoPy, a scientometric tool for topics trend analysis in scientific publications. *Scientometrics*, *121*(2), 1165–1188.

Schindelin, J., Arganda-Carreras, I., Frise, E., Kaynig, V., Longair, M., Pietzsch, T., Preibisch, S., Rueden, C., Saalfeld, S., & Schmid, B. (2012). Fiji: an open-source platform for biological-image analysis. *Nature Methods*, *9*(7), 676–682.

Sewak, M., Karim, M. R., & Pujari, P. (2018). *Practical convolutional neural networks: implement advanced deep learning models using Python*. Packt Publishing Ltd.

Shanmugamani, R. (2018). *Deep Learning for Computer Vision: Expert techniques to train advanced neural networks using TensorFlow and Keras*. Packt Publishing Ltd.

Sharma, P., Balabantaray, B. K., Bora, K., & Mallik, S. (2022). *An Ensemble-Based Deep Convolutional Neural Network for Computer-Aided Polyps Identi fi cation From Colonoscopy*. *13*(April), 1–11. https://doi.org/10.3389/fgene.2022.844391

Shelatkar, T., Urvashi, D., Shorfuzzaman, M., Alsufyani, A., & Lakshmanna, K. (2022). Diagnosis of Brain Tumor Using Light Weight Deep Learning Model with Fine-Tuning Approach. *Computational and Mathematical Methods in Medicine*, *2022*. https://doi.org/10.1155/2022/2858845

Shen, D., Wu, G., & Suk, H.-I. (2017). Deep learning in medical image analysis. *Annual Review of Biomedical Engineering*, *19*, 221.

Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on Image Data Augmentation for Deep Learning. *Journal of Big Data*. https://doi.org/10.1186/s40537-019-0197-0

Solawetz, J., & Nelson, J. (2020). *How to Train YOLOv5 On a Custom Dataset*. https://blog.roboflow.com/how-to-train-yolov5-on-a-custom-dataset/

Solovyev, R., Wang, W., & Gabruseva, T. (2021). Weighted boxes fusion: Ensembling boxes from different object detection models. *Image and Vision Computing*, *107*, 104117.

Souaidi, M., & Ansari, M. E. (2022). A New Automated Polyp Detection Network MP-FSSD in WCE and Colonoscopy Images Based Fusion Single Shot Multibox Detector and Transfer Learning. *IEEE Access*, *10*, 47124–47140. https://doi.org/10.1109/ACCESS.2022.3171238

Souaidi, M., & El Ansari, M. (2022). Multi-Scale Hybrid Network for Polyp Detection in Wireless Capsule Endoscopy and Colonoscopy Images. *Diagnostics*, *12*(8). https://doi.org/10.3390/diagnostics12082030

Srivastava, S., Divekar, A. V., Anilkumar, C., Naik, I., Kulkarni, V., & Pattabiraman, V. (2021). Comparative analysis of deep learning image detection algorithms. *Journal of Big Data*, *8*(1), 1–27.

Stoffel, E. M., & Murphy, C. C. (2020). Epidemiology and Mechanisms of the Increasing Incidence of Colon and Rectal Cancers in Young Adults. *Gastroenterology*, *158*(2), 341–353. https://doi.org/10.1053/j.gastro.2019.07.055

Su, J., Han, J., & Song, J. (2021). A benchmark bone marrow aspirate smear dataset and a multi-scale cell detection model for the diagnosis of hematological disorders. *Computerized Medical Imaging and Graphics*, *90*. https://doi.org/10.1016/j.compmedimag.2021.101912

Suckling, J. (1994). The mammographic images analysis society digital mammogram database. *Exerpta Medica. International Congress Series, 1994*, *1069*, 375–378.

Szegedy, C., Ioffe, S., Vanhoucke, V., & Alemi, A. A. (2017). Inception-v4, inception-ResNet and the impact of residual connections on learning. *31st AAAI Conference on Artificial Intelligence, AAAI 2017*.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2015). Going deeper with convolutions. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. https://doi.org/10.1109/CVPR.2015.7298594

Tan, M., & Le, Q. (2019). Efficientnet: Rethinking model scaling for convolutional neural networks. *International Conference on Machine Learning*, 6105–6114.

Tan, M., Pang, R., & Le, Q. V. (2020). EfficientDet: Scalable and efficient object detection. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 10778–10787. https://doi.org/10.1109/CVPR42600.2020.01079

Tang, C. P., Chen, K. H., & Lin, T. L. (2021). Computer-aided colon polyp detection on high resolution colonoscopy using transfer learning techniques. *Sensors*, *21*(16). https://doi.org/10.3390/s21165315

Tao, A., Barker, J., & Sarathy, S. (2016). Detectnet: Deep neural network for object detection in digits. *Parallel Forall*, *4*.

Taş, M., & Yılmaz, B. (2021). Super resolution convolutional neural network based pre-processing for automatic polyp detection in colonoscopy images. *Computers & Electrical Engineering*, *90*, 106959. https://doi.org/https://doi.org/10.1016/j.compeleceng.2020.106959

Thambawita, V., Hicks, S., Halvorsen, P., & Riegler, M. (2021). *DivergentNets: Medical Image Segmentation by Network Ensemble*.

Thanathornwong, B., & Suebnukarn, S. (2020). Automatic detection of periodontal compromised teeth in digital panoramic radiographs using faster regional convolutional neural networks. *Imaging Science in Dentistry*, *50*(2), 169–174. https://doi.org/10.5624/isd.2020.50.2.169

Tian, Z., Shen, C., Chen, H., & He, T. (2019). FCOS: Fully convolutional one-stage object detection. *Proceedings of the IEEE International Conference on Computer Vision*, *2019-Octob*, 9626–9635. https://doi.org/10.1109/ICCV.2019.00972

Tokuyasu, T., Iwashita, Y., Matsunobu, Y., Kamiyama, T., Ishikake, M., Sakaguchi, S., Ebe, K., Tada, K., Endo, Y., Etoh, T., Nakashima, M., & Inomata, M. (2021). Development of an artificial intelligence system using deep learning to indicate anatomical landmarks during laparoscopic cholecystectomy. *Surgical Endoscopy*, *35*(4), 1651–1658. https://doi.org/10.1007/s00464-020-07548-x

Tong, K., Wu, Y., & Zhou, F. (2020). Recent advances in small object detection based on deep learning: A review. In *Image and Vision Computing*. https://doi.org/10.1016/j.imavis.2020.103910

Tsai, J.-Y., Hung, I. Y.-J., Guo, Y. L., Jan, Y.-K., Lin, C.-Y., Shih, T. T.-F., Chen, B.-B., & Lung, C.-W. (2021). Lumbar Disc Herniation Automatic Detection in Magnetic Resonance Imaging Based on Deep Learning. *Frontiers in Bioengineering and Biotechnology*, *9*. https://doi.org/10.3389/fbioe.2021.708137

Tsuneki, M. (2022). Deep learning models in medical image analysis. *Journal of Oral Biosciences*.

Uka, A., Tare, A., Polisi, X., & Panci, I. (2020). FASTER R-CNN for cell counting in low contrast microscopic images. *2020 International Conference on Computing, Networking, Telecommunications & Engineering Sciences Applications (CoNTESA)*, 64–69.

Van Eck, N., & Waltman, L. (2010). Software survey: VOSviewer, a computer program for bibliometric mapping. *Scientometrics*, *84*(2), 523–538.

Varanda, A. B., Martins-Lograd o, A., Ferreira, M. G., & Fior, R. (2020). Zebrafish Xenografts Unveil Sensitivity to Olaparib beyond BRCA Status. *Cancers*, *12*(7), 1769.

Viola, P., & Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, *1*, I–I.

Voulodimos, A., Doulamis, N., Doulamis, A., & Protopapadakis, E. (2018). Deep learning for computer vision: A brief review. *Computational Intelligence and Neuroscience*, *2018*.

Wallace, M. B., Sharma, P., Bhandari, P., East, J., Antonelli, G., Lorenzetti, R., Vieth, M., Speranza, I., Spadaccini, M., Desai, M., Lukens, F. J., Babameto, G., Batista, D., Singh, D., Palmer, W., Ramirez, F., Palmer, R., Lunsford, T., Ruff, K., … Hassan, C. (2022). Impact of Artificial

Intelligence on Miss Rate of Colorectal Neoplasia. *Gastroenterology*, *May*, 1–10. https://doi.org/10.1053/j.gastro.2022.03.007

Wan, J., Chen, B., & Yu, Y. (2021). Polyp Detection from Colorectum Images by Using Attentive YOLOv5. *Diagnostics (Basel, Switzerland)*, *11*(12). https://doi.org/10.3390/diagnostics11122264

Wang, C. Y., Mark Liao, H. Y., Wu, Y. H., Chen, P. Y., Hsieh, J. W., & Yeh, I. H. (2020). CSPNet: A new backbone that can enhance learning capability of CNN. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, *2020-June*, 1571–1580. https://doi.org/10.1109/CVPRW50498.2020.00203

Wang, Q., Bi, S., Sun, M., Wang, Y., Wang, D., & Yang, S. (2018). Deep learning approach to peripheral leukocyte recognition. *PLoS ONE*, *14*(6). https://doi.org/10.1371/journal.pone.0218808

Wang, Y., Liu, Z., & Deng, W. (2019). Anchor generation optimization and region of interest assignment for vehicle detection. *Sensors (Switzerland)*. https://doi.org/10.3390/s19051089

Wu, L, Xu, M., Jiang, X., He, X., Zhang, H., Ai, Y., Tong, Q., Lv, P., Lu, B., Guo, M., Huang, M., Ye, L., Shen, L., & Yu, H. (2022). Real-time artificial intelligence for detecting focal lesions and diagnosing neoplasms of the stomach by white-light endoscopy (with videos). *Gastrointestinal Endoscopy*, *95*(2), 269-280.e6. https://doi.org/10.1016/j.gie.2021.09.017

Wu, Lingyun, Hu, Z., Ji, Y., Luo, P., & Zhang, S. (2021). Multi-frame Collaboration for Effective Endoscopic Video Polyp Detection via Spatial-Temporal Feature Transformation. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *12905 LNCS*, 302–312. https://doi.org/10.1007/978-3-030-87240-3_29

Wu, Y., Kirillov, A., Massa, F., Lo, W.-Y., & Girshick, R. (2019). *Detectron2*.

Xi, P., Guan, H., Shu, C., Borgeat, L., & Goubran, R. (2020). An integrated approach for medical abnormality detection using deep patch convolutional neural networks. *Visual Computer*, *36*(9), 1869–1882. https://doi.org/10.1007/s00371-019-01775-7

Xia, J., Xia, T., Pan, J., Gao, F., Wang, S., Qian, Y.-Y., Wang, H., Zhao, J., Jiang, X., Zou, W.-B., Wang, Y.-C., Zhou, W., Li, Z.-S., & Liao, Z. (2021). Use of artificial intelligence for detection of gastric lesions by magnetically controlled capsule endoscopy. *Gastrointestinal Endoscopy*, *93*(1), 133-139.e4. https://doi.org/10.1016/j.gie.2020.05.027

Xiang, Y., Sun, W., Pan, C., Yan, M., Yin, Z., & Liang, Y. (2020). A novel automation-assisted cervical cancer reading method based on convolutional neural network. *Biocybernetics and Biomedical Engineering*, *40*(2), 611–623. https://doi.org/10.1016/j.bbe.2020.01.016

Xiao, Y., Tian, Z., Yu, J., Zhang, Y., Liu, S., Du, S., & Lan, X. (2020). A review of object detection based on deep learning. *Multimedia Tools and Applications*, *79*, 23729–23791.

Xie, W., Noble, J. A., & Zisserman, A. (2018). Microscopy cell counting and detection with fully convolutional regression networks. *Computer Methods in Biomechanics and Biomedical Engineering: Imaging and Visualization*. https://doi.org/10.1080/21681163.2016.1149104

Xie, Y., Wu, Z., Han, X., Wang, H., Wu, Y., Cui, L., Feng, J., Zhu, Z., & Chen, Z. (2020). Computer-Aided System for the Detection of Multicategory Pulmonary Tuberculosis in Radiographs. *Journal of Healthcare Engineering*, *2020*. https://doi.org/10.1155/2020/9205082

Xu, C., Li, M., Li, G., Zhang, Y., Sun, C., & Bai, N. (2022). Cervical Cell/Clumps Detection in Cytology Images Using Transfer Learning. *Diagnostics*, *12*(10). https://doi.org/10.3390/diagnostics12102477

Xu, X., Zhou, F., Liu, B., Fu, D., & Bai, X. (2019). Efficient Multiple Organ Localization in CT Image Using 3D Region Proposal Network. *IEEE Transactions on Medical Imaging*, *38*(8), 1885–1898. https://doi.org/10.1109/TMI.2019.2894854

Xue, Z., Novetsky, A. P., Einstein, M. H., Marcus, J. Z., Befano, B., Guo, P., Demarco, M., Wentzensen, N., Long, L. R., Schiffman, M., & Antani, S. (2020). A demonstration of automated visual evaluation of cervical images taken with a smartphone camera. *International Journal of Cancer*, *147*(9), 2416–2423. https://doi.org/10.1002/ijc.33029

Yan, C., Wang, L., Lin, J., Xu, J., Zhang, T., Qi, J., Li, X., Ni, W., Wu, G., Huang, J., Xu, Y., Woodruff, H. C., & Lambin, P. (2022). A fully automatic artificial intelligence–based CT image analysis system for accurate detection, diagnosis, and quantitative severity evaluation of pulmonary tuberculosis. *European Radiology*, *32*(4), 2188–2199. https://doi.org/10.1007/s00330-021-08365-z

Yan, K., Cai, J., Zheng, Y., Harrison, A. P., Jin, D., Tang, Y., Tang, Y., Huang, L., Xiao, J., & Lu, L. (2021). Learning from Multiple Datasets with Heterogeneous and Partial Labels for Universal Lesion Detection in CT. *IEEE Transactions on Medical Imaging*, *40*(10), 2759–2770. https://doi.org/10.1109/TMI.2020.3047598

Yan, K., Wang, X., Lu, L., & Summers, R. M. (2018). DeepLesion: Automated mining of large-scale lesion annotations and universal lesion detection with deep learning. *Journal of Medical Imaging*, *5*(3). https://doi.org/10.1117/1.JMI.5.3.036501

Yang, R., & Yu, Y. (2021). Artificial convolutional neural network in object detection and semantic segmentation for medical imaging analysis. *Frontiers in Oncology*, *11*, 638182.

Yap, M. H., Hachiuma, R., Alavi, A., Brüngel, R., Cassidy, B., Goyal, M., Zhu, H., Rückert, J., Olshansky, M., Huang, X., Saito, H., Hassanpour, S., Friedrich, C. M., Ascher, D. B., Song, A., Kajita, H., Gillespie, D., Reeves, N. D., Pappachan, J. M., … Frank, E. (2021). Deep learning in diabetic foot ulcers detection: A comprehensive evaluation. *Computers in Biology and Medicine*, *135*. https://doi.org/10.1016/j.compbiomed.2021.104596

Younas, F., Usman, M., & Yan, W. Q. (2022). *A deep ensemble learning method for colorectal polyp classification with optimized network parameters*.

Yu, H., Yang, L. T., Zhang, Q., Armstrong, D., & Deen, M. J. (2021). Convolutional neural networks for medical image analysis: state-of-the-art, comparisons, improvement and perspectives. *Neurocomputing*, *444*, 92–110.

Yu, W., Xue, Y., Knoops, R., Yu, D., Balmashnova, E., Kang, X., Falgari, P., Zheng, D., Liu, P., Chen, H., Shi, H., Liu, C., & Zhao, J. (2021). Automated diatom searching in the digital scanning electron microscopy images of drowning cases using the deep neural networks. *International Journal of Legal Medicine*, *135*(2), 497–508. https://doi.org/10.1007/s00414-020-02392-z

Yurtsever, E., Lambert, J., Carballo, A., & Takeda, K. (2020). A survey of autonomous driving: Common practices and emerging technologies. *IEEE Access*, *8*, 58443–58469.

Zaidi, S. S. A., Ansari, M. S., Aslam, A., Kanwal, N., Asghar, M., & Lee, B. (2022). A survey of modern

deep learning based object detection models. *Digital Signal Processing*, 103514.

Zeng, X., Wen, L., Liu, B., & Qi, X. (2020). Deep learning for ultrasound image caption generation based on object detection. *Neurocomputing*, *392*, 132–141. https://doi.org/10.1016/j.neucom.2018.11.114

Zhang, B., Rahmatullah, B., Wang, S. L., Zhang, G., Wang, H., & Ebrahim, N. A. (2021). A bibliometric of publication trends in medical image segmentation: Quantitative and qualitative analysis. *Journal of Applied Clinical Medical Physics*, *22*(10), 45–65.

Zhang, K., Xiong, F., Sun, P., Hu, L., Li, B., & Yu, G. (2019). Double anchor R-CNN for human detection in a crowd. In *arXiv*.

Zhang, R., Zheng, Y., Poon, C. C. Y., Shen, D., & Lau, J. Y. W. (2018). Polyp detection during colonoscopy using a regression-based convolutional neural network with a tracker. *Pattern Recognition*, *83*, 209–219. https://doi.org/10.1016/j.patcog.2018.05.026

Zhang, S., Xu, S., Tan, L., Wang, H., & Meng, J. (2021). Stroke Lesion Detection and Analysis in MRI Images Based on Deep Learning. *Journal of Healthcare Engineering*, *2021*. https://doi.org/10.1155/2021/5524769

Zhang, Z., Zhang, C., Shen, W., Yao, C., Liu, W., & Bai, X. (2016). Multi-oriented text detection with fully convolutional networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 4159–4167.

Zhao, Z.-Q., Zheng, P., Xu, S., & Wu, X. (2019). Object detection with deep learning: A review. *IEEE Transactions on Neural Networks and Learning Systems*, *30*(11), 3212–3232.

Zheng, Y., Chen, Z., Zuo, Y., Guan, X., Wang, Z., & Mu, X. (2020). Manifold-Regularized Regression Network: A Novel End-to-End Method for Cell Counting and Localization. *ACM International Conference Proceeding Series*. https://doi.org/10.1145/3390557.3394299

Zhong, Z., Sun, L., & Huo, Q. (2019). An anchor-free region proposal network for Faster R-CNN-based text detection approaches. *International Journal on Document Analysis and Recognition (IJDAR)*, *22*, 315–327.

Zhou, H., Li, Z., Ning, C., & Tang, J. (2017). Cad: Scale invariant framework for real-time object detection. *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 760–768.

Zhou, S. K., Greenspan, H., Davatzikos, C., Duncan, J. S., Van Ginneken, B., Madabhushi, A., Prince, J. L., Rueckert, D., & Summers, R. M. (2021). A Review of Deep Learning in Medical Imaging: Imaging Traits, Technology Trends, Case Studies With Progress Highlights, and Future Promises. *Proceedings of the IEEE*, *109*(5), 820–838. https://doi.org/10.1109/JPROC.2021.3054390

Zoph, B., Vasudevan, V., Shlens, J., & Le, Q. V. (2018). Learning transferable architectures for scalable image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 8697–8710.

Zou, Z., Shi, Z., Guo, Y., & Ye, J. (2019). Object detection in 20 years: A survey. *ArXiv Preprint ArXiv:1905.05055*.

7. APPENDIXES

APPENDIX 1 – ARCHITECTURE OF INCEPTION RESNET V2

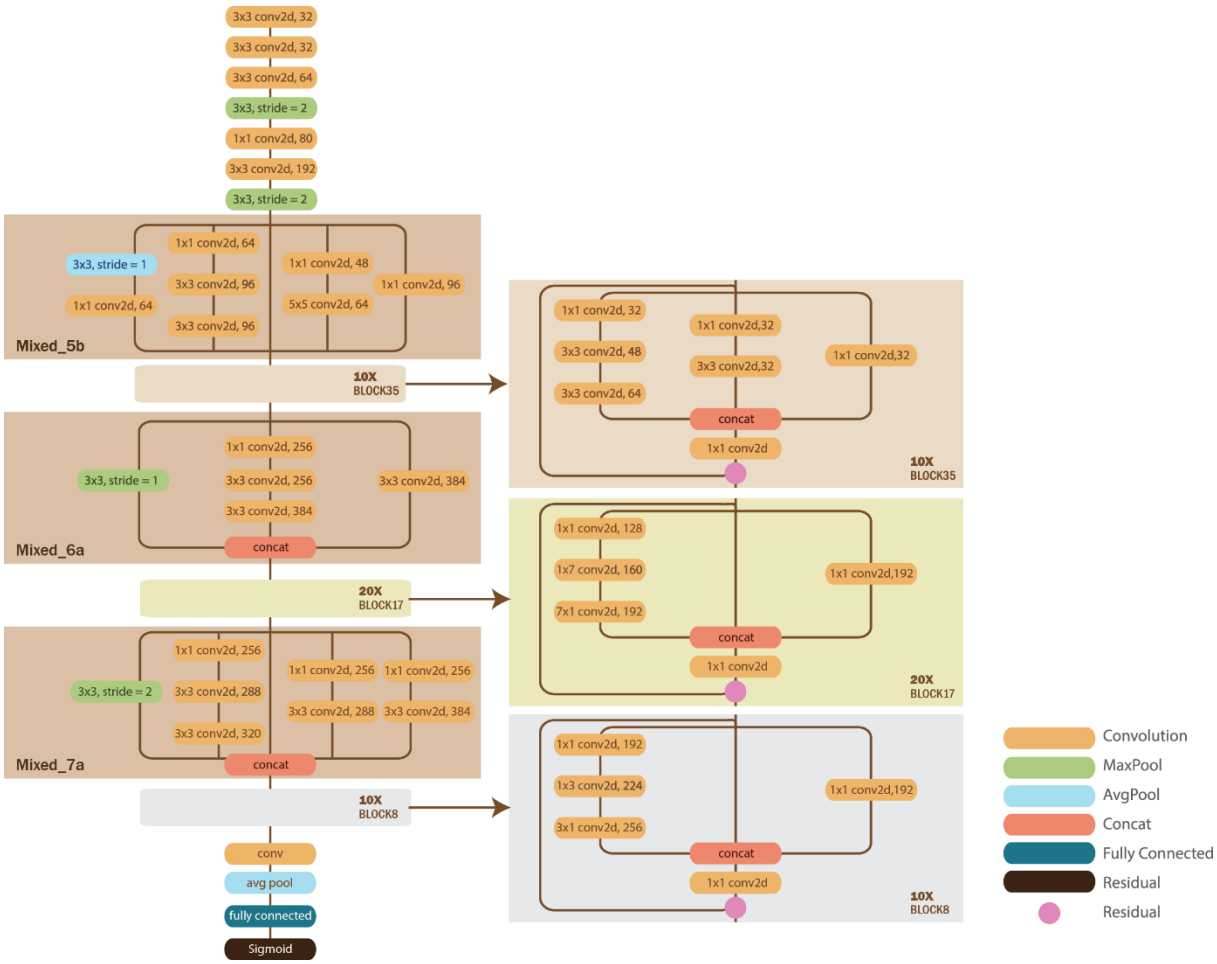

Figure 7.1 – Architecture of Inception ResNet V2, developed during the course of this research

APPENDIX 2 – COLORECTAL ZEBRAFISH XENOGRAFTS

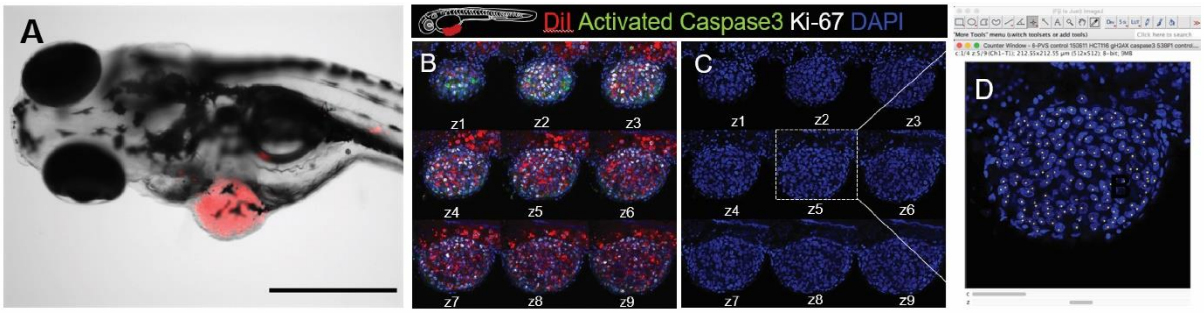

Figure 7.2 – Colorectal zebrafish xenografts

Note. Colorectal cancer cells were labeled using a lipophilic die (DiI-red) and injected into two-days - post-fertilization zebrafish embryos and fixed for immunofluorescence and confocal imaging at four days post injection. Immunofluorescence was performed to detect apoptosis (Activated Caspase3-green) and proliferation (Ki-67-white), and nuclei were counterstained with DAPPI (blue). A-B. Montage of a confocal z stack of CRC zebrafish xenograft. C. Manual counting of one slice using the manual Cell Counter plugin in Fiji.

APPENDIX 3 – REGRESSION METRICS AND MAP VALUES FOR THE VARIOUS MODELS: TRAINING AND DATA AUGMENTATION.

Table 7.1 – Regression metrics and mAP values for the various models: Training and data augmentation.

Parameter	Configuration	RMSE	MAE	MEDAE	mAP
Dimension	256x256	117.2	94.7	83.5	0.07
	512x512	52.9	33.4	17.5	0.60
	600x600*	21.1	15.5	11.5	0.74
	1024x1024	29.5	20.3	13.5	0.81
	1100x1100	**13.5**	**11.4**	**9.5**	**0.82**
Resize Method	Bilinear*	**13.5**	11.4	9.5	**0.82**
	Nearest Neighbor	15.6	11.5	8.5	0.77
	Bicubic	14.0	**10.6**	**7.0**	0.78
	Area	17.8	15.9	13.5	0.75
Optimizer	Momentum *	**7.0**	**5.8**	**4.5**	**0.84**
	Adam	139.2	129.8	122.0	0.03
	RMSProp	15.0	12.2	13.5	0.79
	Adam with exp. LR	9.6	8.5	9.0	0.83
Initial Learning Rate	0.0005	11.1	8.9	9.5	0.80
	0.0004	9.6	8.5	9.0	0.83
	0.0003*	**6.1**	**5.4**	**5.0**	0.83
	0.0002	11.5	9.9	10.0	**0.84**
	0.0001	7.1	6.2	7.0	0.83
Decay steps	25000	17.0	13.0	**7.5**	0.33
	50000	13.9	11.1	10.0	0.83
	100000	**11.5**	**9.9**	10.0	**0.84**
Data Augmentation	None	35.1	27.9	23.5	0.59
	Horizontal Flip *	12.0	9.1	7.5	0.82
	Vertical Flip	16.6	13.6	13.5	0.82
	Pixel Value Scale	12.3	9.9	9.5	0.81
	Image Scale	15.1	12.1	13.5	0.80
	RGB to GraY	8.5	7.8	9.0	0.81
	Adjust brightness	12.7	12.0	13.5	0.82
	Adjust Contrast	37.2	27.2	20.5	0.58
	Adjust Hue	15.5	12.6	9.0	0.78
	Adjust Saturation	12.5	8.4	**5.0**	0.81
	Distort color	16.6	13.6	12.5	**0.83**
	Jitter Boxes	13.0	9.9	6.0	0.76
	Crop Image	21.4	16.1	15.0	0.80
	Pad Image	135.3	122.9	116.5	0.04
	Crop pad Image	70.6	52.6	36.0	0.12
	Crop to aspect ratio	11.8	10.7	10.0	0.82
	Black Patches	17.9	14.0	13.0	0.80
	Rotation 90	15.1	11.9	9.0	0.80
	Top4	**7.2**	**5.9**	6.5	0.81
	All but (08)(13)(14)	15.5	11.9	11.5	**0.83**
	All	16.1	13.1	12.0	0.80

Note. Table with the resulting regression metrics and mAP values for various configurations of training and-data augmentation parameters. The asterisk (*) represents the default model for Faster R-CNN.

APPENDIX 4 – REGRESSION METRICS AND MAP VALUES FOR THE VARIOUS MODELS: FIRST STAGE

Table 7.2 – Regression metrics and mAP values for the various models: First stage

Parameter	Configuration	RMSE	MAE	MEDAE	mAP
Gradient Clipping	5	**12.0**	**9.5**	**8.5**	0.81
	10 *	15.5	11.9	11.5	**0.83**
	15	21.3	17.2	16.5	**0.83**
Feature Extractor Stride	8 *	**15.5**	**11.9**	**11.5**	**0.83**
	16	30.7	24.6	18.0	0.33
Feature Extractor Batch Norm	TRUE	139.2	129.8	122	0.11
	FALSE *	**15.5**	**11.9**	**11.5**	**0.83**
Aspect Ratios and Scales	[0.5,1.0,1.5,2.0]&[0.15,0.3,0.5,1.0]	**11.2**	**8.7**	**6.0**	**0.85**
	[0.5,1.0,2.0]&[0.15,0.3,0.5,1.0]	11.3	9.0	6.5	0.82
Atrous rate	1 (FALSE)	11.3	9.0	6.5	0.84
	2 (TRUE)*	**11.2**	**8.7**	**6.0**	**0.85**
1st Stage Regularizer	No regularizer*	**11.2**	**8.7**	**6.0**	**0.85**
	l2 0.5	12.7	9.6	9.0	0.82
	l2 1.0	17.3	14.2	9.0	0.80
	l1 0.5	33.4	28.4	30.5	0.61
	l1 1.0	41.0	34.2	29.5	0.63
Activation	NONE	14.9	11.7	8.5	0.82
	RELU*	**11.2**	**8.7**	**6.0**	**0.85**
	RELU_6	14.7	11.4	11.0	0.84
Batch Normalization	TRUE	25.2	17.4	11.0	0.79
	FALSE*	**11.2**	**8.7**	**6.0**	**0.85**
Regularize Depthwise	TRUE	73.1	38.9	20.0	0.80
	FALSE*	**11.2**	**8.7**	**6.0**	**0.85**
Kernel Size	3*	**11.2**	8.7	**6.0**	**0.85**
	2	11.7	**8.5**	6.5	0.82
	4	18.3	13.8	9.0	0.84
Depth	256	13.2	11.6	10.0	0.84
	512*	**11.2**	**8.7**	**6.0**	**0.85**
	1024	17.5	14.8	13.0	0.83
Minibatch Size	128	13.7	11.5	11.0	0.83
	256*	**11.2**	**8.7**	6.0	**0.85**
	512	13.1	9.0	**5.0**	0.83
Positive Balance Fraction	0.25	**9.9**	**7.8**	6.5	0.83
	0.5*	11.2	8.7	**6.0**	**0.85**
	0.75	18.5	15.0	11.5	0.82
NMS IoU threshold	0.3	14.2	11.4	8.0	0.84
	0.4	15.0	12.1	10.0	0.84
	0.5	11.5	10.1	10.0	0.84
	0.6	14.5	11.5	7.5	0.83
	0.7*	**11.2**	**8.7**	**6.0**	**0.85**
	0.8	13.7	10.9	9.5	0.83
Maximum Proposals	1000	11.9	8.9	6.5	0.83
	1500	15.3	11.2	8.5	0.83
	2000*	15.0	12.1	10.0	0.84
	2500	**11.1**	**8.6**	**6.0**	0.83
	3000	21.1	15.5	11.5	**0.85**
Loss weights	Localization 1.0 & Objectness 1.0	17.5	15.7	13.0	0.84
	Localization 2.0 & Objectness 1.0*	21.1	15.5	11.5	**0.85**
	Localization 1.0 & Objectness 2.0	**11.0**	**9.7**	**8.5**	0.84
Initial crop size	14	**11.3**	**8.7**	**7.0**	0.84
	17 *	21.1	15.5	11.5	**0.85**
Maxpool kernel & stride	Kernel 1 and Stride 1 *	21.1	15.5	11.5	**0.85**
	Kernel 3 and Stride 3	**10.4**	8.9	10.5	**0.85**
	Kernel 3 and Stride 1	11.3	**8.8**	**8.5**	0.84

Note. Table with the resulting regression metrics and mAP for various configurations of first-stage parameters. The asterisk (*) represents the default model for Faster R-CNN.

APPENDIX 5 – REGRESSION METRICS AND MAP VALUES FOR THE VARIOUS MODELS: SECOND STAGE

Table 7.3 – Regression metrics and mAP values for the various models: Second stage

Parameter	Configuration	RMSE	MAE	MEDAE	mAP
2nd Stage Regularizer	No regularizer *	21.1	15.5	11.5	**0.85**
	l2 0.5	13.8	12.0	11.0	0.83
	l2 1.0	10.7	8.3	6.0	**0.85**
	l1 0.5	10.6	9.2	8.5	**0.85**
	l1 1.0	**9.3**	**7.6**	**5.5**	0.83
Activation	NONE	17.4	13.2	12.5	0.84
	RELU *	21.1	15.5	11.5	**0.85**
	RELU_6	**11.3**	**9.7**	**9.5**	**0.85**
Batch Normalization	TRUE	**10.9**	**8.8**	**7.0**	0.81
	FALSE *	21.1	15.5	11.5	**0.85**
Regularize depthwise	TRUE	**8.7**	**7.9**	**7.5**	**0.85**
	FALSE *	21.1	15.5	11.5	0.84
Dropout	FALSE *	21.1	15.5	11.5	**0.85**
	0.5	17.1	14.1	15	0.82
	0.6	13.7	10.8	11.0	0.83
	0.7	17.3	13.1	8.5	0.81
	0.8	17.6	13.7	13.0	0.82
	0.9	**10.0**	**8.8**	**8.0**	0.83
NMS IoU threshold	0.4	10.3	9.2	8.0	0.83
	0.5	**9.3**	**7.8**	**7.5**	0.83
	0.6*	21.1	15.5	11.5	**0.85**
	0.7	18.8	14.3	8.5	0.84
	0.8	11.9	10.5	9.0	0.83
Maximum detections	700 *	21.1	15.5	11.5	**0.85**
	1000	**9.4**	**7.2**	**7.5**	0.83
	1500	22.6	17.3	15.0	0.86
Loss weights	Localization 1.0 & Classification 1.0	**7.9**	**5.7**	**3.5**	0.83
	Localization 2.0 & Classification 1.0 *	21.1	15.5	11.5	**0.85**
	Localization 1.0 & Classification 2.0	12.6	9.5	7.0	0.83
Classification Loss	Weighted sigmoid *	21.1	15.5	11.5	**0.85**
	Bootstrapped sigmoid	**12.7**	**11.5**	**11.0**	0.84
	Focal sigmoid	14.5	12.2	**11.0**	0.83

Note. Table with the resulting regression metrics and mAP for various configurations of second-stage parameters. The asterisk (*) represents the default model for Faster R-CNN.

APPENDIX 6 – GROUND-TRUTH LABELING AND INFERRED DETECTIONS FOR LOW-COMPLEXITY IMAGES

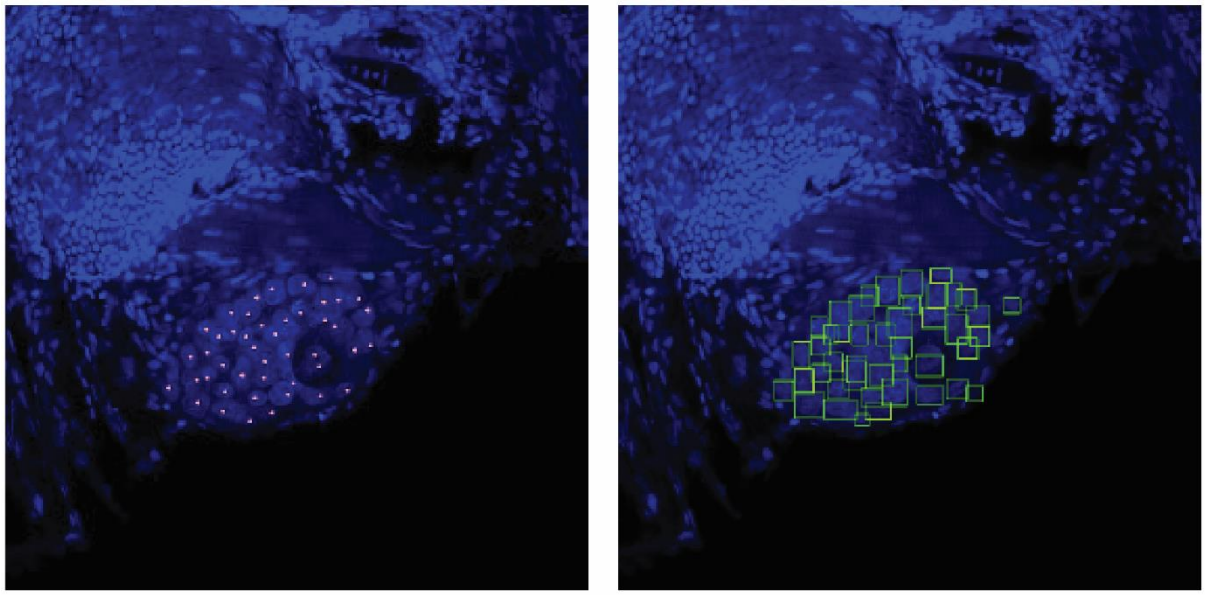

Figure 7.3 – Ground-truth labeling and inferred detections for low-complexity images

Note. Comparison between the ground truth and the inferred detections in low-complexity images. On the left is an image labeled by a medical expert. The results of the trained model are provided on the right.

APPENDIX 7 – GROUND-TRUTH LABELING AND INFERRED DETECTIONS FOR MEDIUM-COMPLEXITY IMAGES

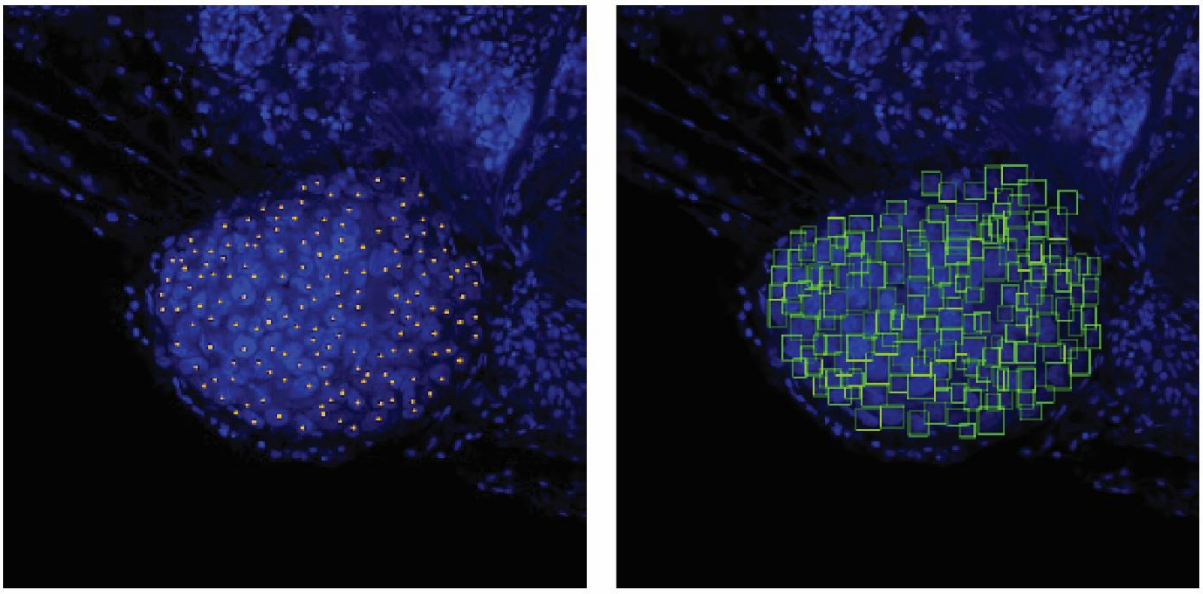

Figure 7.4 – Ground-truth labeling and inferred detections for medium-complexity images

Note. Comparison between the ground truth and the inferred detections in medium-complexity images. On the left is an image labeled by a medical expert. The results of the trained model are provided on the right.

APPENDIX 8 – GROUND-TRUTH LABELING AND INFERRED DETECTIONS FOR IMAGES WITH HIGH COMPLEXITY

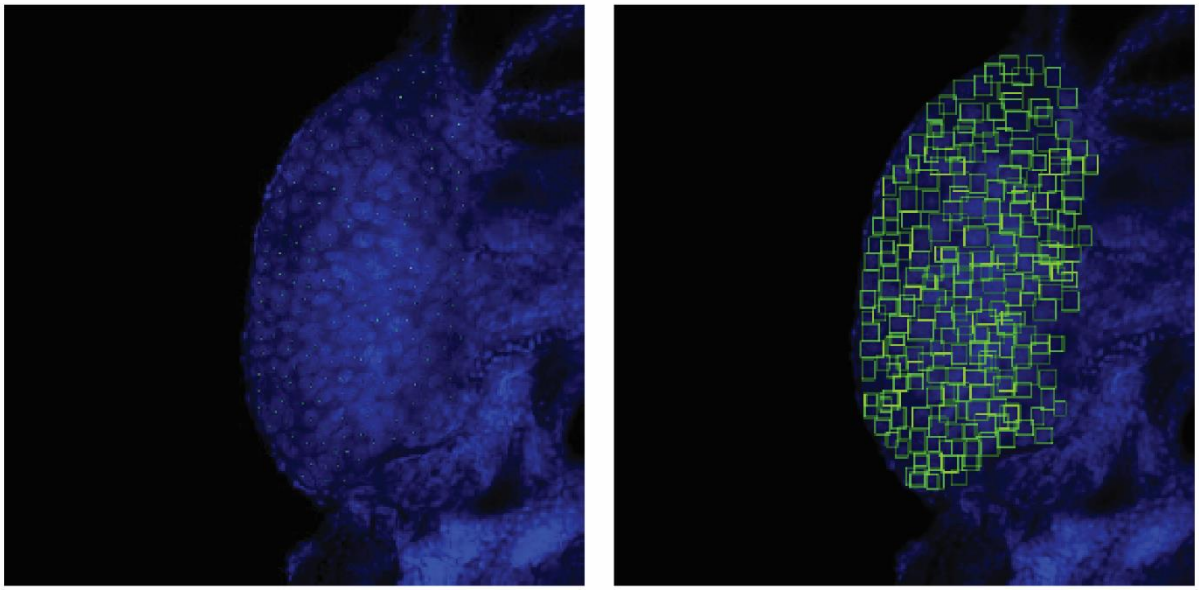

Figure 7.5 – Ground-truth labeling and inferred detections for images with high complexity

Note. Comparison between the ground truth and the inferred detections in high-complexity images. On the left is an image labeled by a medical expert. The results of the trained model are provided on the right.

APPENDIX 9 – OVERVIEW OF THE IMPLEMENTED OBJECT-DETECTION SYSTEM.

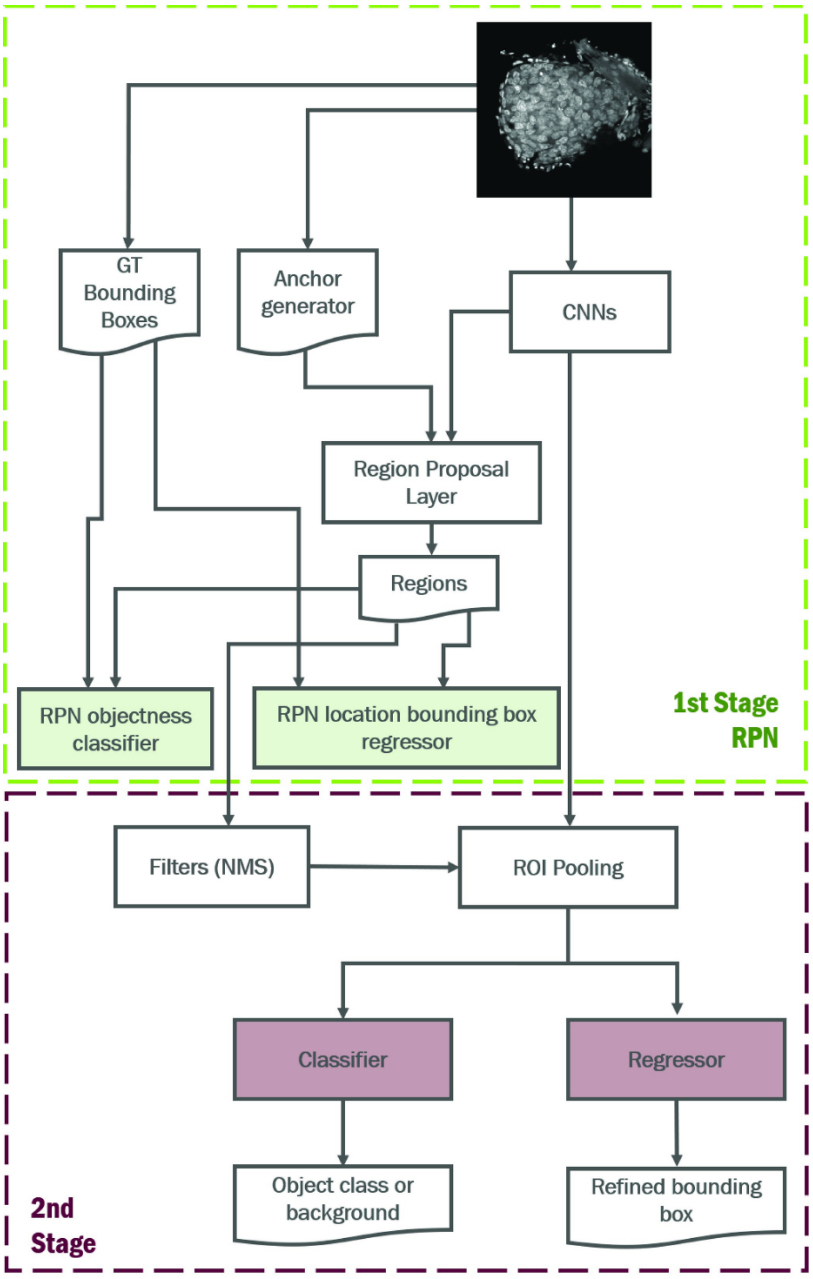

Figure 7.6– Overview of the Faster R-CNN object-detection system implemented in this research.

Note. This system includes a first stage (RPN) whose main purpose is to propose a set of regions where objects could be present, and a second phase in which the output of the first phase is used to detect and classify objects in those regions.

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação

Universidade Nova de Lisboa