รีแคสเน็ต:การลดความไม่เข้ากันในกรอบระบบการตรวจจับเซลล์ขณะไมโทสิสโดยอัตโนมัติแบบสองขั้นตอน

นายชวาล เพียรสัตยานนท์

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญา
วิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2565

RECASNET: REDUCING MISMATCH WITHIN THE TWO-STAGE MITOSIS DETECTION FRAMEWORK

Mr. Chawan Piansaddhayanon

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Engineering Program in Computer Engineering
Department of Computer Engineering
Faculty of Engineering
Chulalongkorn University
Academic Year 2022

Thesis Title	RECASNET: REDUCING MISMATCH WITHIN THE TWO-STAGE MITOSIS DETECTION FRAMEWORK
By	Mr. Chawan Piansaddhayanon
Field of Study	Computer Engineering
Thesis Advisor	Ekapol Chuangsuwanich, Ph.D.

Accepted by the Faculty of Engineering, Chulalongkorn University in Partial Fulfillment of the Requirements for the Master's Degree

. . . . Dean of the Faculty of Engineering
(Professor Supot Teachavorasinskun, D.Eng.)

THESIS COMMITTEE

. . . . Chairman
(Punnarai Siricharoen, Ph.D.)

. . . . Thesis Advisor
(Ekapol Chuangsuwanich, Ph.D.)

. . . . Thesis Co-Advisor
(Sira Sriswasdi , Ph.D.)

. . . . Examiner
(Professor Shanop Shuangshoti , M.D.)

. . . . External Examiner
(Itthi Chatnuntawech , Ph.D.)

ชวาล เพียรสัตยานนท์: รีแคสเน็ต:การลดความไม่เข้ากันในกรอบระบบการตรวจจับเซลล์ขณะไมโทสิสโดยอัตโนมัติแบบสองขั้นตอน. (RECASNET: REDUCING MISMATCH WITHIN THE TWO-STAGE MITOSIS DETECTION FRAMEWORK) อ.ที่ปรึกษาวิทยานิพนธ์หลัก : ดร. เอกพล ช่วงสุวนิช, 75 หน้า.

การนับจำนวนเซลล์ขณะไมโทสิสนั้นเป็นตัวแปรที่สำคัญในทางพยาธิวิทยาสำหรับการวินิจฉัยและตรวจระดับโรคมะเร็ง แต่ทว่าการจะได้มาซึ่งตัวแปรนี้โดยใช้แพทย์เป็นผู้ตรวจนั้นใช้เวลายาวนานและมีโอกาสผิดพลาดได้ ดังนั้นจึงมีระบบการเรียนรู้เชิงลึกจำนวนหนึ่งที่ได้ถูกเสนอมาเพื่อช่วยกระบวนการนี้โดยการตรวจจับเซลล์ขณะไมโทสิสทั้งหมดในภาพสไลด์ โดยระบบที่ถูกเสนอเหล่านี้เกือบทั้งหมดเป็นระบบการทำงานแบบสองขั้นตอนซึ่งประกอบไปด้วย ขั้นตอนตรวจจับ โดยจะมีแบบจำลองสำหรับตรวจจับวัตถุเพื่อเสนอตำแหน่งที่เซลล์ขณะไมโทสิสน่าจะอยู่ และ ขั้นตอนการจำแนก ซึ่งจะมีแบบจำลองสำหรับแยกแยะประเภทวัตถุมาปรับปรุงความมั่นใจของวัตถุจากขั้นตอนที่แล้วโดยละเอียด ถึงแม้กระนั้นการแก้ปัญหาด้วยวิธีนี้ก็นำมาซึ่งปัญหาใหม่เช่นเดียวกัน เนื่องจากขั้นตอนการจำแนกนั้นประสบปัญหาจากการทำงานที่ไม่สม่ำเสมอของขั้นตอนตรวจจับ และความแตกต่างของการกระจายตัวของชุดข้อมูลฝึกสอน ดังนั้น งานนี้จึงได้เสนอ ระบบการปรับปรุงคุณภาพแบบเป็นขั้นตอน (รีแคสเน็ต) ซึ่งเป็นกระบวนการที่ออกแบบมาเพื่อบรรเทาปัญหาที่กล่าวมาก่อนหน้า โดยงานนี้ได้เสนอการพัฒนาจากระบบเดิมขึ้นมาสามประการ ประการแรกคือเปลี่ยนกระบวนการย้ายหน้าต่างเพื่อให้ผลการทำนายคุณภาพต่ำที่ถูกเสนอโดยขั้นตอนตรวจจับลดลง ประการที่สองคือการปรับปรุงตำแหน่งศูนย์กลางของวัตถุ โดยมีแบบจำลองอีกตัวเพื่อเสนอตำแหน่งศูนย์กลางที่แท้จริงของวัตถุ ประการที่สามคือการปรับปรุงการเลือกข้อมูลมาฝึกสอนของขั้นตอนการจำแนกเพื่อให้การกระจายตัวของชุดข้อมูลฝึกสอนของทั้งสองขั้นตอนลดลง ทั้งนี้ระบบที่เสนอมานั้นได้ถูกนำมาวัดผลในฐานข้อมูลมะเร็งเต้านมและผิวหนังสุนัขขนาดใหญ่ เพื่อพิสูจน์ประสิทธิภาพของระบบ โดยการศึกษาพบว่าวิธีที่เสนอในงานนี้ได้ทำให้ค่า F1 เพิ่มจากจะระบบเดิมที่มีอยู่มากขึ้นสูงสุดถึงร้อยละ 4.8 โดยสัมบูรณ์ และทำให้ความผิดพลาดของการนับจำนวนเซลล์ขณะไมโทสิสลดลงสูงสุดร้อยละ 28.2 โดยงานที่เสนอมานั้นควรจะสามารถนำไปใช้ได้ทั่วไปในระบบการทำงานแบบสองขั้นตอน และทำให้ประสิทธิภาพโดยรวมของระบบการเรียนรู้เชิงลึกในงานด้านพยาธิวิทยานั้นสูงขึ้น

ภาควิชา วิศวกรรมคอมพิวเตอร์ ลายมือชื่อนิสิต ....
สาขาวิชา วิศวกรรมคอมพิวเตอร์ ลายมือชื่อ อ.ที่ปรึกษาหลัก ....
ปีการศึกษา 2565

6372025021: MAJOR COMPUTER ENGINEERING
KEYWORDS: MITOTIC COUNT / WHOLE SLIDE IMAGE / OBJECT DETECTION / IMAGE CLASSIFICATION / MULTI-STAGE DEEP LEARNING

CHAWAN PIANSADDHAYANON : RECASNET: REDUCING MISMATCH WITHIN THE TWO-STAGE MITOSIS DETECTION FRAMEWORK. ADVISOR : EKAPOL CHUANGSUWANICH, Ph.D., 75 pp.

Mitotic count (MC) is an important histological parameter for cancer diagnosis and grading, but the manual process to obtain this metric is tedious and not fully reproducible across different pathologists. To mitigate this problem, several deep learning models have been utilized to speed up the process. Typically, the problem is formulated as a two-stage deep learning pipeline: the detection stage for proposing the potential candidates for mitotic cells and the classification stage for refining prediction confidences from the former stage. However, this paradigm can lead to inconsistencies in the classification stage due to the poor prediction quality of the detection stage and the mismatches in training data distributions between the two stages. This thesis proposes a Refine Cascade Network (ReCasNet), an improved deep learning pipeline that introduces three improvements to alleviate the aforementioned problems. First, window relocation was used to suppress poor-quality false positive boxes produced by the detection stage around the sliding window border. Second, we proposed an additional deep learning model to align the poorly centered objects to the true object center. Third, additional data were queried from the training slides to teach the classification stage to bridge the training distribution gap between the two stages. We evaluated the performance of ReCasNet on two public large-scale mitotic figure recognition datasets, canine cutaneous mast cell tumor (CCMCT) and canine mammary carcinoma (CMC). By using our proposed pipeline, we achieved up to 4.8% F1 improvements for mitotic cell detection performance and 44.1% reductions in mean absolute percentage error (MAPE) for MC prediction. Techniques that underlie our proposed method can be generalized to other detection and classification algorithms and should contribute to improving the performances of deep learning models in broad digital pathology applications.

Department:	Computer Engineering	Student's Signature	....
Field of Study:	Computer Engineering	Advisor's Signature	....
Academic Year:	2022		

Acknowledgements

Foremost, I would like to express my sincere gratitude to my advisor Prof. Ekapol Chuangsuwanich for introducing me to the computer vision field and encouraging me to advance further throughout my master's study. I am also very grateful for being invited to cooperate with NVAITC, which gives me professional research experience. I also would like to express my gratitude to Prof. Sira Sriswasdi, my co-advisor, for giving me a lot of helpful suggestions and introducing me to the medical imaging field.

Besides my advisors, I also would like to thank Prof. Shanop Shuangshoti and Sakun Santisukwongchote for giving me crucial insights related to pathology imaging and responding to some of my questionable questions. Dr. Qingyi Tao for giving me very informative and helpful advice that highly contributed to the findings of the manuscript. Prof. Punnarai Siricharoen and Itthi Chatnuntawech for providing a valuable critique that greatly improves the quality of the manuscript.

CONTENTS

Page

Page

LIST OF TABLES

Table **Page**

LIST OF FIGURES

Chapter I

INTRODUCTION

This chapter describes the problem statement containing the rationale for automatic mitosis detection, the topic primarily discussed throughout this thesis, along with the thesis objective and scope of work. The problem statement first explains the importance of mitotic count on tumor grading and its challenge. Then, deep learning systems that were proposed to automate the process were introduced along with the potential issue with the current paradigm. Lastly, the method for mitigating the aforementioned issue was proposed along with a brief result to demonstrate its effectiveness. The thesis objective and scope of work then state the main objective of this thesis, which is proposing an improvement of the existing two-stage mitosis detection pipeline by modifying the existing automatic mitosis detection system. Part of this thesis was published in Artificial Intelligence in Medicine.

1.1 Problem statement

Cancer is the major leading cause of death worldwide, and early detection of this disease greatly increases the survivorship rate of the patients (Hawkes, 2019). Typically, during the cancer diagnosis process, the pathologist is responsible for identifying the important histologic parameter of the tissue samples from the patient. One of the most important parameters is a Mitotic Count (MC), an integer that indicates the number of tumor cells during the division process (mitosis) at the area with the highest mitotic density (hotspot). To obtain this value, the pathologist has to manually scan through the tumor tissue and find the area that has the highest mitosis cell concentration and count all of them within the 10 high-power microscopic fields region. With the increasing use of digital pathology, whole slide images (WSI) that could store the whole tissue inside a single file is now routinely generated in several pathology laboratories. However, the method for obtaining mitotic count is still relatively unchanged as the process just changes from observing from the microscope to a computer image. In addition, an acquisition of this metric (MC) is tedious and could not be fully reproducible (Veta et al., 2016). This is because the identification of cells during mitosis is subjective across different pathologists. Thus, several studies (Pan et al., 2021) have proposed an algorithm to assist pathologists by automatically recognizing mitotic figures in the WSI and identifying the hotspot area. The current trend in automatic mitosis detection is the use of deep learning as it has demonstrated a highly promising image recognition performance and is now widely used in a wide range of medical imaging applications, including histopathological image analysis (Srinidhi et al., 2021).

Despite the significant improvement in the machine learning field, the model is still not flawless, as prediction errors are still being made. For the mitosis detection task, a lack of availability of clean data collection is a major obstacle to model improvement. This is because, as mentioned before, the identification of mitotic figures is subjective across different pathologists. First, the mitosis figure itself could also be divided into several substages, namely prophase, metaphase, anaphase, and telophase. The identification of cells during prophase is difficult as there is no clear visual difference from the normal cell. This problem leads to drastically different mitotic counts reported by the experts (Bertram et al., 2019). Second, the WSI acquisition process is flawed since the slide is only scanned on a single focal plane that could not be refocused. This results in some objects becoming out-of-focus, leading to poor texture information. Despite these problems, automated mitosis detection is still an important task in histopathological image analysis and is an active area of research.

To develop an automatic mitosis detection system, a dataset has to be created so that model could learn to distinguish mitotic figures from other cells. Thus, many competitions, such as the ICPR MITOS-2012 (Roux et al., 2013), AMIDA 2013 (Veta et al., 2014), ICPR MITOS-ATYPIA-2014 (Racoceanu et al., 2014), and TUPAC16 (Veta et al., 2019), provided an annotation of mitosis cell location on several high power fields (HPF) and organized the competition to improve mitosis detection performance. However, the annotations are only provided on the HPF level, but not on the whole slide level. This leads to the model not understanding the full context of the WSI, as the vast majority of the area is still not annotated. In addition, the number of annotated mitosis cells provided in these challenges is low, often fewer than one thousand objects each. Therefore, canine cutaneous mast cell tumor (CCMCT) (Aubreville et al., 2019) and the canine mammary carcinoma (CMC) (Aubreville et al., 2020a) dataset that provides a complete annotation on the WSI are later introduced into the field. These datasets allow the model to learn from the greatly increased diversity, which greatly contributes to a significant increase in model performance (Aubreville et al., 2019). Nonetheless, these datasets are not flawless since they are annotated with a fixed-size bounding box with a radius of 25 pixels, a stark difference from the realistic setting as the mitosis cell does not have a static cell area.

The formulation of deep learning approaches to solve the task, in addition to imperfections in data acquisition and annotation, has a significant impact on how well the model performs. In most deep learning systems for mitosis detection, the task of mitosis recognition is frequently divided into two steps: the detection and classification stages (Chen et al., 2016; Li et al., 2018; Alom et al., 2020). One of the main causes of this is that the model cannot operate directly on WSI due to its enormous size. Instead, the WSI must be divided into smaller patches using a sliding window, from which the inference is then carried out on by the detection stage on each window to determine where mitotic figures are located using deep object detection or segmentation model. The classification stage then refines the prediction results by first extracting the position of each predicted mitotic figure and revising the corresponding image patch to center it around the mitotic figure and ensure that only one mitotic figure is contained within the patch. After that, each revised image patch is fed into a deep object classifier to generate a confidence score. The classification stage significantly improves mitotic figure recognition performance by overcoming the disadvantage of the detection stage, which must handle a much broader variety of image patches, some with no mitotic figure and others with multiple mitotic figures.

Despite the advantages mentioned above, the classification stage of a multi-stage pipeline suffers from inconsistent input data received from the detection stage and a mismatch in training distribution. The outputs of inference at the detection stage would inevitably include inaccurate object locations and poor bounding boxes, resulting in inconsistently positioned objects at the image patch of the following stage. The inconsistency results in classification stage performance degradation because most convolutional neural networks do not possess the shift-invariant property to properly handle the changes in distributions of object locations and bounding boxes produced by the detection stage (Engstrom et al., 2017). The use of a sliding window makes the situation even worse because it may split an object into pieces across multiple patches, increasing the number of low-quality false positives. It is also not negligible that the training data distributions differ between the two stages. The classification stage mainly observes mitotic figures and other objects with similar appearances, whereas the detection stage learns the entire data distribution of the WSI. When the classification stage receives inputs without a mitotic figure, this training distribution mismatch results in an out-of-distribution problem. By using all predictions, even those with low confidence, from the detection stage to train the classification stage, DeepMitosis (Li et al., 2018) alleviates this issue. On large datasets, however, where the detector suggests hundreds of thousands of objects, this method is impractical.

To address all of these issues, we propose Refine Cascade Network (ReCasNet), an improved deep learning pipeline for improving mitosis recognition performance on large-scale mitotic figure recognition datasets. Our pipeline improves classification stage performance by increasing the consistency of input data distribution and exposing the model to more informative data. First, we propose Window Relocation, a simple but effective method for overcoming the weakness of an overlapping sliding window by removing objects near the window border and re-evaluating them as the center of newly extracted patches. This method attempts to eliminate bad bounding boxes while requiring less computation than the overlapping sliding window. Second, we present an Object Center Adjustment Stage, a deep learning model that bridges the gap between the classification and detection stages. To reduce the variance in input translation, it generates new image patches that center on mitotic figures predicted by the detection stage and feeds them to the classification stage. Third, we improve the DeepMitosis verification model's training data sampling process (i.e., classification stage) by focusing on a subset of informative samples from the proposed objects on which the detector and the classifier disagree the most.

We demonstrated the effectiveness of ReCasNet on the CCMCT and CMC datasets. Our proposed pipeline achieved 83.2% test F1 on the CCMCT dataset and 82.3% test F1 on the CMC dataset, which is a +1.2 and +4.8 percentage point improvements over the baseline work, respectively (Aubreville et al., 2019, 2020a). We also benchmarked the performance on the slide level on both datasets to compare the HPF and mitotic count (MC) produced by our pipeline to the ground truth. It was also found that our method resulted in up to 44.1%, and 49.3% less mean absolute percentage error (MAPE) compared to the baseline under the fully automated and human-in-the-loop mitotic count, respectively.

1.2 Objective

The primary objective of this thesis is to propose a new method to mitigate the existing problem of the two-stage object detection pipeline on large-scale mitosis detection tasks, which suffer from poor-quality bounding boxes produced by the detector and inconsistencies in training distribution across multiple stages. The experiments were also conducted to verify the effectiveness of our method.

1.3 Scope of work

This thesis is mainly focused on improving the mitotic recognition performance on large-scale mitosis detection tasks where a large quantity, complete bounding box annotation of mitotic figures over the whole slide image could be obtained.

Chapter II

BACKGROUND

This chapter describes the background knowledge necessary for the thesis. The chapter is divided into three subchapters: deep convolutional neural network (CNN), deep objection detection, translation variance in the CNN, and mitotic figure.

2.1 Deep convolutional neural network

This subchapter explains the progression of convolutional neural network (CNN) architecture spanning from LeNet-5 (Lecun et al., 1998) to ConvNext (Liu et al., 2022), a CNN architecture currently used in our work.

LeNet-5

LeNet-5 is one of the first proposed CNN architectures. It introduced the concept of the Convolution-Activation-Subsampling scheme to recognize handwritten digit characters. First, the convolution layer was used to extract features from the image then activation was performed to add a non-linearity to the optimization program. Since the network architecture in this age is still shallow, Tanh activation is still commonly used as the function is zero-centered. After that, the subsampling operation (average pooling) was applied to reduce the feature size to decrease the computation cost and observe a wider receptive field. By using this architecture, it achieved less than 1% test set error on the MNIST character recognition dataset. Figure 2.1 illustrates the overview of LeNet-5 architecture.

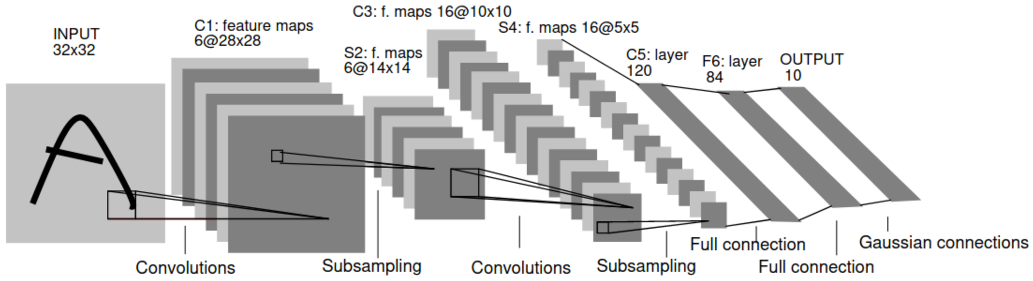


Figure 2.1: Illustration of LeNet-5 architecture. The image is taken from (Lecun et al., 1998).

AlexNet

AlexNet (Krizhevsky et al., 2012) is a CNN architecture proposed in 2012. It achieved a vastly superior classification performance over the traditional method on the ImageNet dataset (Deng et al., 2009) in that year. To accommodate a change in a greatly increased image size of 224 × 224 compared to the MNIST dataset which has a resolution of 28 × 28, it had proposed several improvements over LeNet-5. First, the proposed network was scaled up with the size of the image. AlexNet had increased the filter size, the number of filters, and the number of layers from 5 to 8. As a result, the use of Tanh as an activation function was becoming untenable as it suffered from a vanishing gradient after multiple layers. Therefore, a non-saturated Relu was used instead to avoid this problem. Second, it changed the non-overlapping mean filter used in LeNet-5 to an overlapping max filter to increase the size of the receptive field and capture a sharp input signal. Third, it introduced the concept of multi-GPU training to hasten the training process. Figure 2.2 illustrates the overview of the AlexNet architecture.

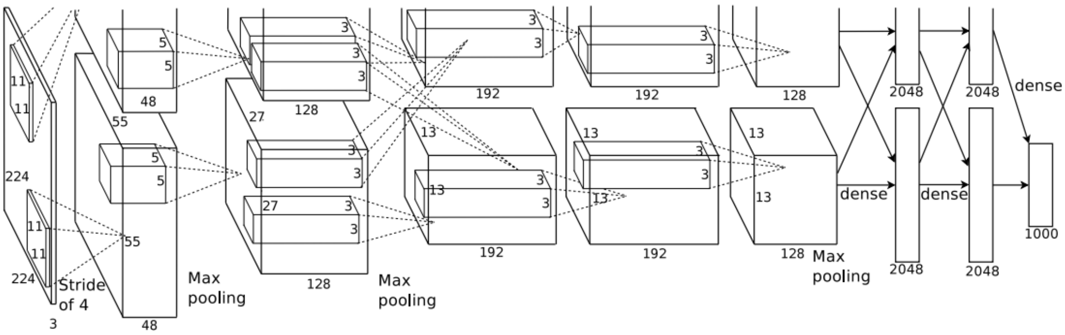


Figure 2.2: Illustration of AlexNet architecture. The image is taken from (Krizhevsky et al., 2012).

VGG

VGG (Simonyan and Zisserman, 2015) is a CNN architecture proposed in 2014 and achieved second place in the 2014-ImageNet competition. It changed the paradigm of CNN architectural design by prioritizing the depth of the network and the number of filters over the size of the filter. Compared to AlexNet, which has the largest filter size of 11 × 11, the filter size in VGG architecture was no larger than a mere 3 × 3 filters, and the number of layers was instead increased to compensate for the decreased filter size. With this approach, the number of convolution layers that could be stacked was increased to 19 layers. Nonetheless, further scaling this network was significantly becoming more difficult due to gradient exploding and vanishing problems.

ResNet

ResNet architecture (He et al., 2016) overcame the problem of vanishing gradient problem by introducing a skip connection to bypass multiple layers so that the gradient could be backpropagated to the top of the network. Combined with the inclusion of batch normalization proposed by Inception-v2 (Ioffe and Szegedy, 2015), it allowed the number of layers (depth) of the network to be greatly increased to 152, leading to a significantly better image classification performance while not suffering major training instability. This paper also adopted the use of bottleneck block to reduce computation cost while maintaining the same classification performance. Figure 2.3 illustrates the architecture of ResNet-34.

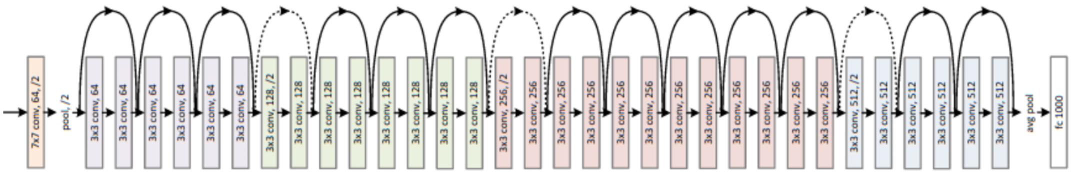


Figure 2.3: Illustration of ResNet-34 architecture. The image is taken from (He et al., 2016).

SENet

Squeeze-and-Excitation Networks (SENet) (Hu et al., 2018) adopted a popularly used attention mechanism in the natural language processing field (Vaswani et al., 2017) to the existing CNN architecture, allowing it better understand the global image context. This work proposed a Squeeze-and-Excitation (SE) block that performed a global average pooling to compress a $H \times W \times C$ feature map into a $1 \times 1 \times C$. Then, it was further compressed into a $1 \times 1 \times \frac{C}{r}$ tensor and got uncompressed back to the original $1 \times 1 \times C$ feature map. After that, the uncompressed feature map was multiplied by the original feature. This forced the model to learn the importance of each channel by interacting with other channels as the information was compressed twice so that only important feature was retained. The design of this block is similar to the attention mechanism which had interaction with all other word embeddings in the whole input sequence. An illustration of the SE block is shown in Figure 2.4.

EfficientNet

One way to improve the performance of the CNN architecture is by increasing the network depth, the number of filters (network width), or input image resolution. Nevertheless, this method is not well-scaled as the performance gain diminishes as the model size keeps increasing. EfficientNet (Tan and Le, 2019) conducted an

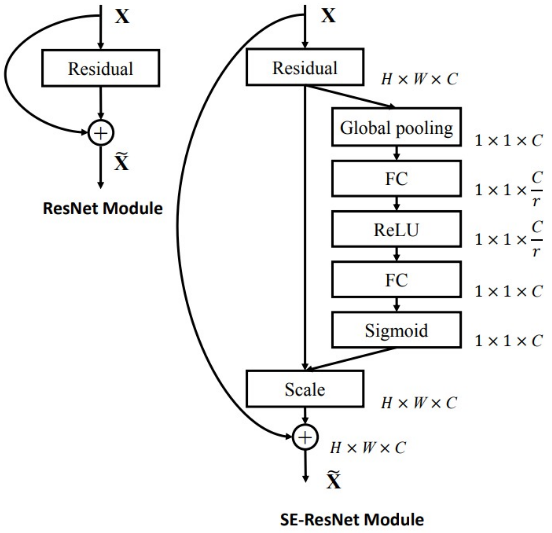


Figure 2.4: Illustration of Squeeze and Excitation block on the convolution block of the ResNet architecture. The image is taken from (Hu et al., 2018).

extensive experiment to examine the relationship between the breadth, depth, and resolution of the CNN by searching for an optimal relation between them. It was found that the breadth, depth, and resolution obtained through the search were more effective than scaling only one parameter and could also generalize across different network architectures. This proposed architecture also adopted SE block and mobile inverted bottleneck block (Sandler et al., 2018) as a base convolution block to utilize global information and efficiently learned the representation.

Vision Transformer

Even though convolution neural networks are typically used in image recognition architecture because they can capture spatial correlation, Vision Transformer (Dosovitskiy et al., 2021) chose to abandon this presumption and instead use a purely attention-based model. Vision Transformer was inspired by BERT (Devlin et al., 2019), which outperformed Bidirectional LSTM (Huang et al., 2015), which operated the input (word) sequentially, by using self-attention (Vaswani et al., 2017) and a feedforward layer as key network components. The interaction between the words in the sentence was represented by self-attention, and the feedforward layer was then utilized to aggregate the collected data. The two layers were then grouped into a building block and used it to stack the block many times to increase the model complexity. However, despite the improvement in Natural Language Processing, this framework could not be directly applied in computer vision as the image is an extremely long sequence of pixels.

Vision Transformer circumvented the aforementioned problem by breaking down the whole image into a sequence of small image patches. Each flattened patch was then projected into a low-dimension input and fed into a Transformer Encoder that returned the object class as an output. The attention in this setting was repurposed into a module that observed the interactions between each patch throughout the whole image, albeit at the cost of a highly inefficient amount of model parameters compared to the standard CNN architecture. Despite its shortcoming, Vision Transformer achieved a competitive result to a state-of-the-art CNN during the published time. Figure 2.5 illustrates the architecture of Vision Transformer.

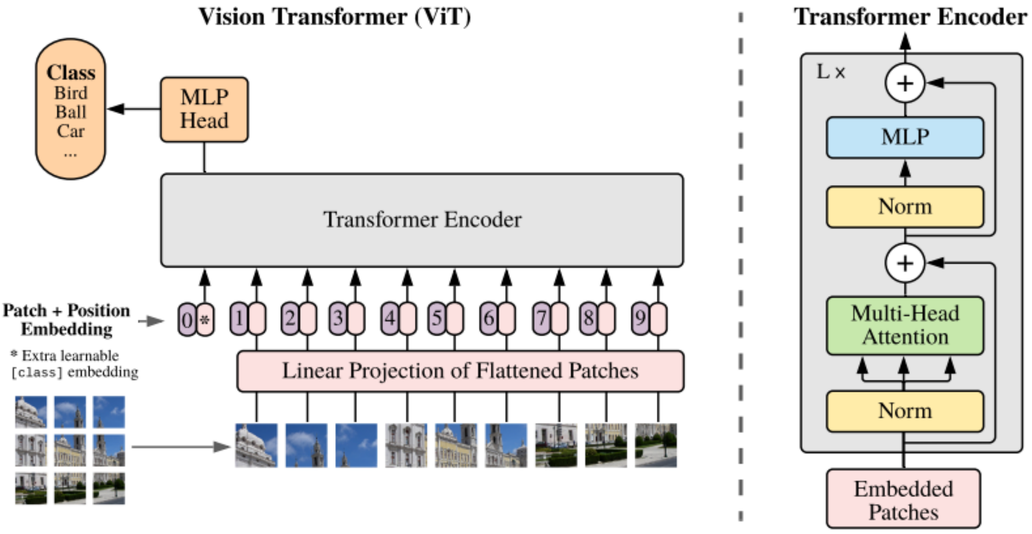


Figure 2.5: Overview of Vision Transformer. The image is taken from (Dosovitskiy et al., 2021).

Swin Transformer

Despite the use of a Transformer leading to great success in the image recognition task, it came at the cost of being extremely parameter inefficient since it could not efficiently harness the spatial information like CNN. Moreover, both memory and computation complexity also scaled quadratically with the image size as the number of tokens was also quadratically multiplied, leading to drastically increased computation cost in the self-attention layer. Therefore, Swin Transformer (Liu et al., 2021) was proposed to mitigate the aforementioned problems. Instead of using attention to learn the interaction between each patch in the whole image, Swin Transformer opted for a hierarchical structure where each transformer layer could only attend to a small area within the image. The limited window was then gradually merged and overlapped with neighboring patches as the number of layers was increased to expand the receptive field. This allowed the model to be more computationally efficient as its mechanism could utilize local information more effectively while reducing the computation complexity when computing self-attention to linearly scale with the image size. The architecture was also shown to be highly effective in other vision tasks, such as object detection and semantic segmentation. An architectural overview of the Swin Transformer is shown in Figure 2.6.

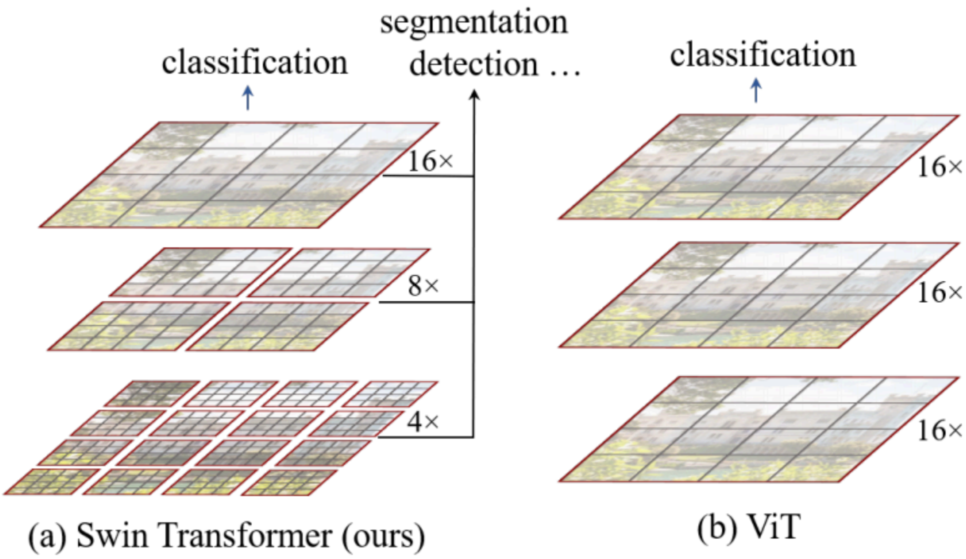


Figure 2.6: Swin Transformer architecture compared to standard Vision Transformer. The image is taken from (Liu et al., 2021).

ConvNext

ConvNext (Liu et al., 2022) proposed an alternative to modern Vision Transformers by examining the factors that contributed to transformers' success and applying them to the standard CNN architecture. Their work suggested that the concept of patching the convolutional filter in the network stem, increasing kernel size, and shifting from standard Batchnorm-ReLU normalization to a GELU-Layernorm scheme (Hendrycks and Gimpel, 2016; Ba et al., 2016) that mimics the behavior of the transformer, along with proper computationally efficient convolutional block design and scaling (Xie et al., 2017; Sandler et al., 2018), led to a network with higher image recognition performance. Their work also showed that under the same computational budget, when used as a network backbone, ConvNext yielded better performance than the transformer counterparts in several major computer vision fields.

2.2 Deep object detector

A deep object detector is a deep learning model that receives an image as an input and returns a set of bounding boxes $\{(x_1, y_1, w_1, h_1, S_1)$, ..., $(x_n, y_n, w_n, h_n, S_n)\}$, where each tuple in the set represents the center of the predicted object, object width, object height, positive object confidence, respectively. Typically, modern object detectors use a CNN to extract image features, which are used to feed to the classification and regression output head that return the confidence of each class and the size of the bounding box, respectively. Since the predicted bounding might sometimes be overlapped with each other, non-maximum suppression was performed on the overlapped predictions, retaining only the one with the highest confidence.

The subchapters explain the progression of deep object detector model starting from Overfeat (Sermanet et al., 2013) to YOLOF (Chen et al., 2021), an object detector currently used in our work.

Overfeat

Overfeat is one of the first works that applied a deep convolutional neural network to an object detection task. This work tackled the problem in a very straightforward approach by using the CNN trained on an image classification task to perform inference in a sliding window fashion over multiple scales. However, simply applying this method does not work since the objects have different sizes, and some patches might not even contain any object at all. Therefore, this work modified the existing network to be suitable for the object detection task by training the model to distinguish between foreground and background objects and introducing another regressor to adjust the size of the predefined bounding box. Nevertheless, this method is extremely computationally extensive since the inference has to be performed multiple times over multiple scales on a single image.

R-CNN

Regions with CNN features (R-CNN) (Girshick et al., 2014) offered a considerable performance gain over Overfeat by removing the sliding window element. Instead of using a sliding window to perform inference across the whole image, R-CNN used a selective search to propose the regions that were likely to contain a foreground object (region of interest). A CNN was then applied to the proposed region to extract the features and the prediction head to recognize object class and adjust the size of the bounding box. This method had a significant advantage over a sliding window approach as the number of inferences was now limited to the number of the proposed regions. Nonetheless, an inference still had to be performed on a single image several times to extract features from all proposals. In addition, since the proposal could be any shape, every proposal had to be resized into a single, fixed-size image. Figure 2.7 showed an overview of R-CNN pipeline.

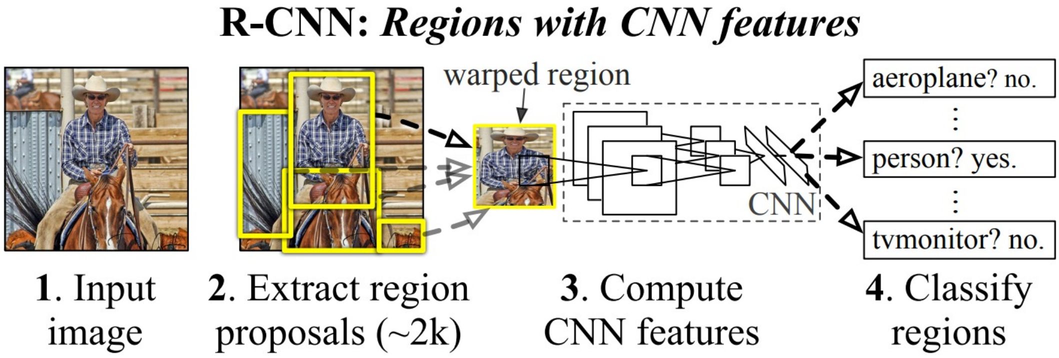


Figure 2.7: Overview of R-CNN pipeline. The image is taken from (Girshick et al., 2014).

Fast R-CNN

Fast Region-based Convolutional Network (Fast R-CNN) (Ren et al., 2015b) significantly improved the detection performance over R-CNN as an inference was now only performed once to obtain the detection result. Fast R-CNN used a selective search to propose the region of interest on the extracted CNN features from the image instead of directly proposing it from the raw input. Moreover, the ROI pooling was also proposed to convert the raw feature maps of different sizes into a fixed-size feature vector without the use of image resizing. This improvement allowed the training of Fast R-CNN to be 10 times faster than R-CNN and 200 times faster at test time. However, this method still relied on a selective search algorithm to generate object proposals, leading to a significant time bottleneck during training and testing.

Faster R-CNN

Faster R-CNN (Ren et al., 2015b) completely removed the use of a selective search to propose an object of interest and replaced it with a region proposal network (RPN), resulting in a fully convolutional network for the object detection task. The RPN divides the image into multiple 32 × 32 grids where each grid contains k fixed-size box named anchor. The objective of the RPN is to learn whether there exists a foreground object in each anchor and perform a regression to adjust the anchor size. If the anchor predicted that it contains a foreground, the fixed-size vector is then extracted from the feature map generated by the backbone using ROI pooling and fed to another CNN, which predicts the exact object class and refines the bounding box generated by the FPN. Since this network required two CNNs to perform detection, this design paradigm could also be referred to as a two-stage detector. Figure 2.8 showed an overview of Faster R-CNN.

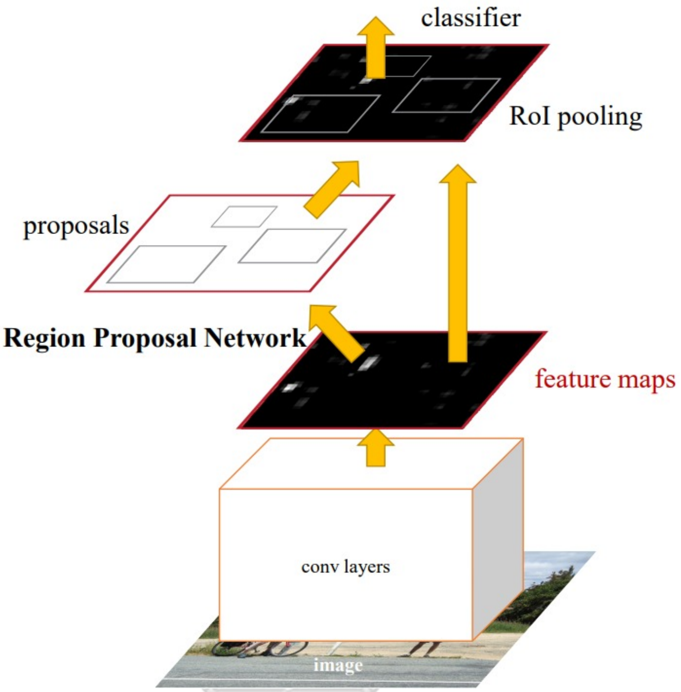


Figure 2.8: Overview of Faster R-CNN. The image is taken from (Ren et al., 2015a).

Cascade R-CNN

Though a fully convolutional network was viable with the advent of Faster R-CNN, it also came up with some heuristics to make it functional. One important heuristic is the usage of an anchor, a set of rough, pre-defined bounding boxes that were used as guidance for the RPN to distinguish whether there was an object that overlapped with the anchor. When the ground truth highly intersects with the anchor, the anchor is considered a positive object and then fed to the second stage to adjust the pre-generated box. Otherwise, it is treated as a negative anchor (background). However, the process of determining whether there was an object close to the anchor is also another heuristic since it used Intersection-over-Union (IOU), which was fixed to a static threshold of 0.5, as a criterion. Cascade R-CNN (Cai and Vasconcelos, 2018) stated in their work that the process of selecting an IOU threshold is indeed a non-negligible problem. Choosing a higher IOU threshold caused the model to perform better at a high IOU evaluation threshold but worse at a low IOU evaluation threshold, and vice versa. This indicated that testing at different IOU evaluation thresholds from the fixed training value caused performance degradation due to distribution mismatch.

Cascade R-CNN, therefore, proposed a method that would allow the model to be trained at multiple IOU thresholds by iteratively performing bounding box correction with progressive IOU threshold. As shown in Figure 2.9, Faster R-CNN used RPN to generate rough bounding boxes (B0, left) and then performed an ROI pooling, which converted each box into a fixed-size feature vector and fed it to a second stage CNN (H1, left) to provide a box correction (B1, left). Cascade R-CNN instead performed the correction process in an iteratively bootstrapped manner. The corrected box (B1, right) was then further fed to another pooling layer and second-stage CNN (H1, right) again, and the process was iteratively repeated until satisfied (H3). Each bounding box correction CNN was also not matched at the same level of the IOU threshold and was set to be progressively increased for the latter one. This simple modification improved the detection performance by 4% absolute performance improvement over the Faster R-CNN.

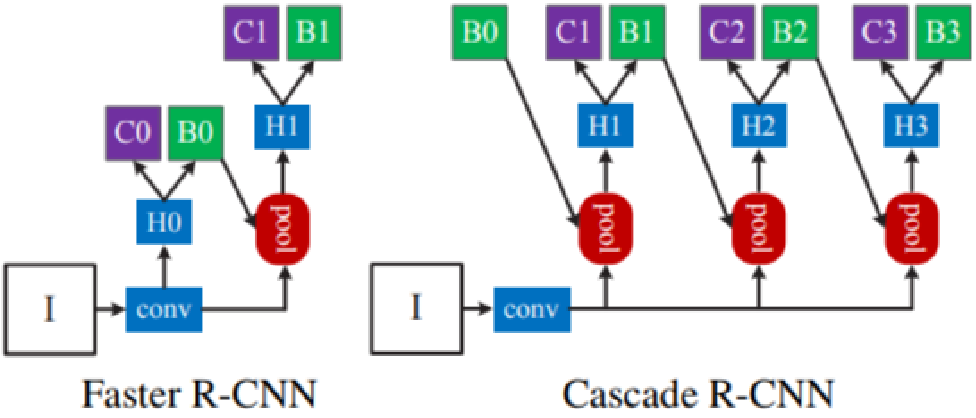


Figure 2.9: A comparsion between the second stage of Faster-RCNN and Cascade-RCNN. The image is taken from (Cai and Vasconcelos, 2018).

Single Shot MultiBox Detector (SSD)

Single Shot MultiBox Detector (SSD) (Liu et al., 2016) was among the first object detector to get recognized as a one-stage detector: a standalone CNN that can generate bounding boxes and their respective class without the need for the external module. SSD removed ROI pooling, along with the second convolution module responsible for correcting the bounding boxes coarsely generated by the first detection stage, and instead relied on the first stage itself to adjust the prediction. However, this inevitably led to a decrease in detection performance as the model itself also had to handle input scale variance while finding possible objects. Thus, SSD mitigated these problems by allowing the feature maps from other backbone stages to be proposed by the anchor, allowing the model to predict the object at different scales. In addition, along with distinguishing foreground from background objects, the object proposal (anchor) generated by the RPN also had to correct the position and size in case the proposed object is not a background. These changes led to severe training instability due to greatly increased task complexity, and many techniques had to be used to stabilize the training process. By using their proposed framework, SSD yielded a comparable performance to Faster-RCNN on the pascal VOC dataset (Everingham et al., 2010) while being able to achieve a real-time inference (> 30FPS). YOLOv2 and YOLOv3 (Redmon et al., 2016; Redmon and Farhadi, 2017) could also be considered a successor of SSD as they further stabilized the training process and formalized all main training heuristics. Figure 2.10 showed a complete architecture of SSD.

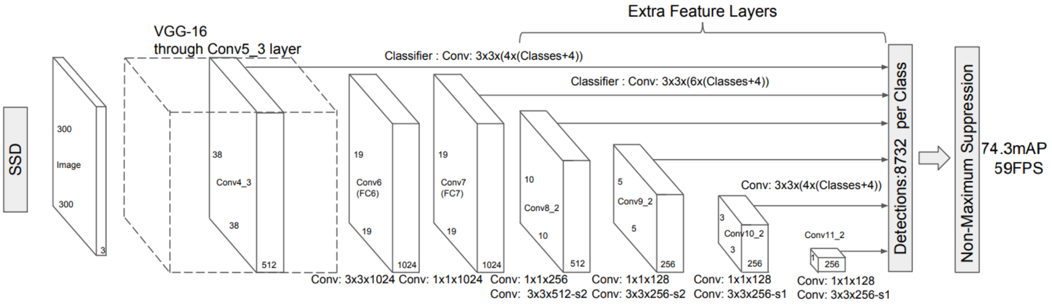


Figure 2.10: A complete architecture of SSD. The image is taken from (Liu et al., 2016).

Feature Pyramid Network (FPN)

One of the prominent issues when using a CNN-based object detector is its inability to truly achieved scale invariance. This result in the model not producing the same output even if the object is the same but scaled differently. Consequently, the detector's effectiveness is generally correlated with object size, as the class that contains a lot of large objects is often easier to be detected than the medium or small one. Therefore, Feature Pyramid Network (FPN) (Lin et al., 2017a) was among the first to recognize this issue and propose a method to mitigate this problem. Figure 2.11 illustrates the detection network when incorporated with FPN.

Their work pointed out that the scale problem could easily alleviate if the model could detect objects at different scales. A straightforward way to do this is performing inference on the same at different scales in a pyramidal manner. However, this method is computationally extensive since it has to perform an inference on the image several times. Thus, instead of using the original image, they used the representation learned during the forward pass of the network at different downsampled levels as a scaled version of an input. The selected representation of each downsampled level was then projected with additional convolutional blocks and combined with upscaled information of the lower stage to generate the bounding boxes. This method allowed the detector to generate output at different scales while only incurring marginal extra computation cost, resulting in a greatly increased detection performance on the COCO object detection dataset (Lin et al., 2014). Due to its simplicity, the FPN was widely adopted in most detectors, even in Transfomer-based backbones that required fewer inductive biases than CNN. Thus, it is still an active research field on how to effectively improve representation at different scales.

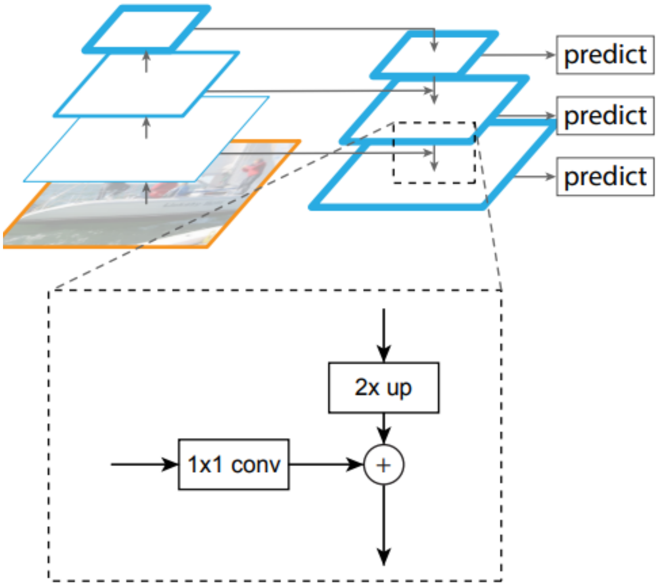


Figure 2.11: An example of FPN when used with a detection network and its upscaling module. The image is taken from (Lin et al., 2017a).

RetinaNet

Despite the success of the one-stage detector that yields a promised detection performance, this approach also introduced a new major challenge: it coupled the classification and regression task. The two-stage detectors separated the task of proposing the object and correcting position into two distinct networks, allowing the classification and regression task to be separately optimized, eventually leading to a stabilized training process. On the other hand, both objectives are being optimized at the same time in the one-stage detector, resulting in a severe vanishing classification loss. This is because most boxes generated by the anchor are background objects, causing an average classification loss to be very low as the positive object loss is overwhelmingly weighted out by the negative ones. Ultimately, this causes the ratio between regression and classification to be drifted from the intended value and makes positive objects harder to classify.

RetinaNet (Lin et al., 2017b) thus proposed focal loss to mitigate the vanishing classification loss. The focal loss is a parameterized version of standard cross entropy loss by multiplying the standard loss with $(1-p_t)^\gamma$. Due to the nature of the exponential function, the additional term allows the loss of an object with low confidence to be scaled down stronger than high confidence one, resulting in foreground objects getting significantly more loss weight than the background one. The parameter γ was used to control the extremeness of the exponential and should be adjusted proportionally to the ratio between foreground and background object. Combining with FPN, RetinaNet yielded a considerable performance improvement compared to SSD, YOLOv2, and Faster-RCNN. Figure 2.12 illustrates the change in classification loss when varying γ.

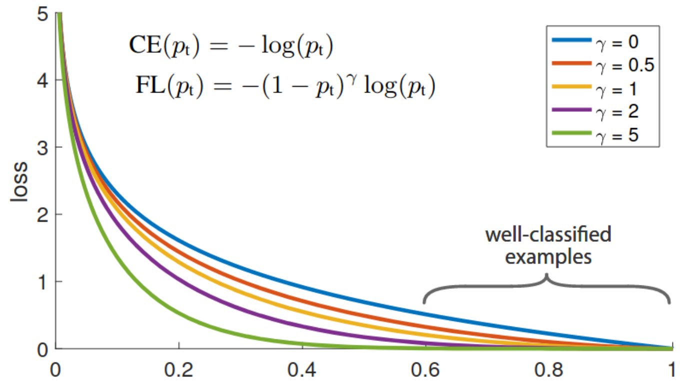


Figure 2.12: A plot showing a change in classification loss when varying focal loss parameter γ. The image is taken from (Lin et al., 2017b).

You Only Look One-level Feature (YOLOF)

You Only Look One-level Feature (YOLOF) (Chen et al., 2021) revisited the issue of FPN on modern detector. Initially, most works in the object detection field believed that the success of FPN largely came from its ability to utilize multi-scale feature and perform prediction at multiple scales. Thus, many works had been working toward finding better way to represent multi-scale feature (Liu et al., 2018; Ghiasi et al., 2019; Tan et al., 2020). This work showed that this common belief was not entirely correct. As shown in Figure 2.13, when removing the multiple-scale feature element from the FPN, the performance only dropped by 0.9 % (b) while removing the multi-scale prediction resulted in 12.0 % in detection performance (c), suggesting that the prediction at multiple scales was way more important than good multi-scale information. This finding was also held largely true in a large Vision Transformer model (Li et al., 2022).

This work also proposed that removing the FPN while retaining the same level of performance was possible. They introduced a Dilated Encoder network neck,

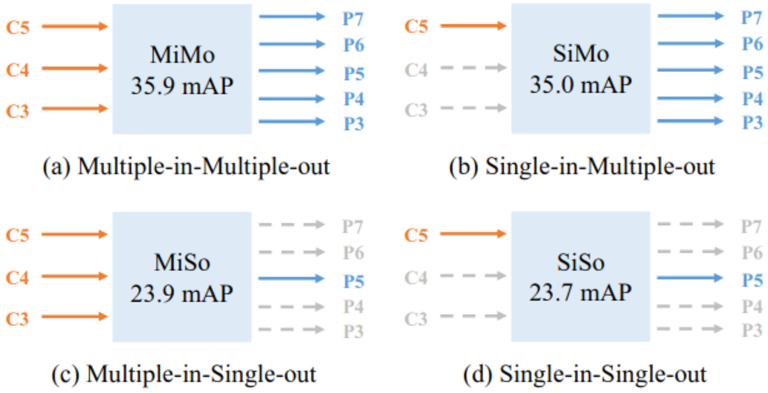


Figure 2.13: A detection performance comparison on different FPN design. The image is taken from (Chen et al., 2021).

as illustrated in Figure 2.14, to learn features at different scales by stacking dilated convolution to simulate input to varying scales and residual connections to preserve the information from the previous downsampled level. However, the proposed neck also raised a new issue as the number of anchors was significantly reduced compared to standard FPN. This caused the proportion of positive anchors to be greatly increased, leading to class imbalance as the standard MaxIoU anchor matching algorithm preferred a large object over the small one since it assigned the box that is highly overlapped with the ground truth to be a positive object. Thus, they proposed to perform Uniform Matching that had the same number of positive anchors regardless of object sizes instead of using standard IOU matching. It was found that YOLOF achieved better detection performance than RetinaNet and also performed an inference three times faster.

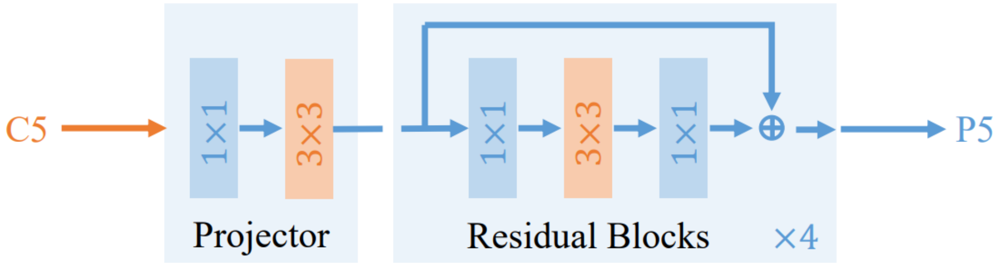


Figure 2.14: An illustration of Dilated Encoder proposed in YOLOF. The image is taken from (Chen et al., 2021).

2.3 Translation variance in the CNN

Even though the convolution neural network is now widely used in computer vision tasks, its robustness to real-world situations is still questionable. One main reason is that CNN is prone to adversarial example (Goodfellow et al., 2015). For example, the confidence of the classifier could be drastically different just by applying small noise to the image. Even without such a complicated attack, CNN's ability to handle input translation variance is also in question.

Engstrom et al. (Engstrom et al., 2017) showed that the CNN is also still prone to adversarial attacks caused by simple geometric transformations like rotation and translation operation, and the use of translation and rotation is not sufficient to resolve this problem. This is because even though CNN could tolerate variance in input translation due to a presence of a downsampling layer, it does not possess a shift-invariant property. This topic is highly related to our thesis since the positive object (mitotic figure) resides closely with many background objects. Figure 2.15 shows an example of an adversarial attack caused by simple geometric transformations.

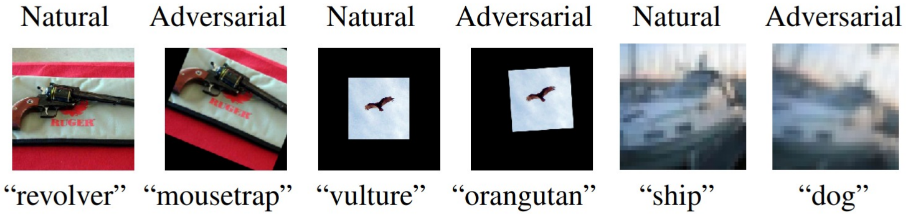


Figure 2.15: Example of adversarial attack caused by simple geometric transformations. The image is taken from (Engstrom et al., 2017).

2.4 Mitotic figure

A mitotic figure (MF) is a cell undergoing the division process (mitosis). It is defined as a group of a nucleus containing short rods or spikes of chromosomes getting unbounded by a nuclear membrane (Donovan et al., 2020). The phase of mitosis could be further divided into four main phases: Prophase, Metaphase, Anaphase, and Telophase. The cell division process starts in Prophase where the chromosomes begin to undergo a condensation process, resulting in reduced chromosome length and increased thickness. The condensed chromosomes then align themselves at the center of the cell in a linear plate, band, or ring shape during Metaphase. After that, in Anaphase, the chromosomes will start splitting into two equal clusters and then moving to opposite ends of the cell at the Telophase stage. Finally, the two clusters are physically separated into two different cells in the process named Cytokinesis. Photograph examples of the mitotic figure at different stages and their characteristics are shown in Figure 2.16.

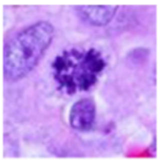
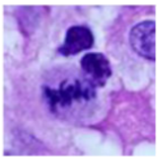
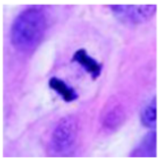
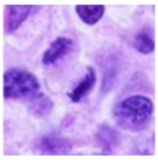
Photomicrograph example	Structure	Characteristics
	Prometaphase MF	Dark aggregate (cluster or ring shape) with spikes/projections
	Metaphase MF	Dark aggregate (linear or ring shape) with spikes/projections
	Anaphase MF	Two separated aggregates variable distances apart; linear with spikes/projections
	Telophase MF	Two separated aggregates at opposite ends of the cell; cleavage furrow

Figure 2.16: Characteristics of the mitotic figure. The image is taken from (Donovan et al., 2020).

Chapter III

RELATED WORK

This chapter describes the prior works that are closely related to our problem setup. First, we review the traditional machine learning approach for mitosis detection that requires expert guidance to design an informative feature that could describe the visual property of the mitosis cell. Next, we move on to the deep learning approach, which primarily uses CNN instead to create a set of convolutional filters that could distinguish mitotic figures from other objects, and their modifications to improve the pipeline performance. Finally, we focus on the works and their findings that come along with the advent of large-scale mitosis recognition datasets.

Many detection algorithms have been proposed to solve the problem of automatic mitosis detection. Hand-crafted object detection was once a popular approach for automatic mitosis detection (Veta et al., 2013; Khan et al., 2012; Sommer et al., 2012; Paul and Mukherjee, 2015; Tek, 2013; Huang and Lee, 2012; Nateghi et al., 2017; Paul et al., 2015). Prior to the resurgence of the deep learning approach, it was also widely used in general computer vision tasks. In this method, the object candidates were proposed first by assigning the probability of each pixel being a mitotic figure using traditional computer vision techniques, and then a threshold was applied. Following that, the pathologist's knowledge was used to extract the shape, texture, and statistical features of the mitotic figure candidates. Finally, the extracted features were fed into a classifier to differentiate between objects of interest and the background. On the ICPR MITOS-2012, AMIDA 2013, and ICPR MITOS-ATYPIA-2014 datasets, this approach outperformed deep object detection. However, this approach would likely to struggle on large-scale datasets because manually handcrafting the features that could represent hundred of thousands of mitotic figures would be extremely laborious and might not even be well-generalized to new datasets.

Deep learning is another solution to the problem. This paradigm achieves cutting-edge performance on a wide range of general computer vision tasks, including image classification, object detection, and semantic segmentation. Furthermore, it could be used for medical imaging tasks, leading to widespread adoption (Litjens et al., 2017). Malon et al. (Malon and Cosatto, 2013) proposed the location of the candidate for mitotic cells using image processing, and then recognized mitotic figures using hand-crafted and CNN features. Cireşan et al. (Cireşan et al., 2013) used CNN to train a single-stage pixel-level classifier to recognize mitotic figures on an image patch and performed inference in a sliding window fashion, eliminating the need for hand-crafted features. CasNN (Chen et al., 2016) was among the first to detect mitosis using a two-stage pipeline. The first stage involved training a semantic segmentation network to suggest the location of mitotic cells. Following that, the classification network was used to fine-tune the prediction result. DeepMitosis (Li et al., 2018) switched the first stage's detection algorithm from semantic segmentation to object detection, resulting in a significant performance improvement. A semantic segmentation network was used to estimate the bounding box in the dataset without pixel-level annotation. MitosisNet (Alom et al., 2020) altered the first stage by framing the problem as multi-task learning, in which both segmentation and detection tasks were trained concurrently. Despite significant progress, benchmarks are mostly performed on small-scale datasets.

The introduction of a large-scale mitosis detection dataset (Aubreville et al., 2019, 2020a) allowed for the evaluation of model performance on a per-slide basis. Aubreville et al. (Aubreville et al., 2020b) compared three deep learning-based methods for identifying the mitotic density in the WSI of canine cutaneous mast cell tumor (CCMCT). A two-stage pipeline with a dedicated object detector achieved the highest correlation between the predicted and ground truth mitotic count. Furthermore, the models' predictions outperformed individual experts in most cases. Bertram et al. (Bertram et al., 2021) later demonstrated that using a model to assist a human expert by pre-selecting the region of interest resulted in a consistently more accurate mitotic count. In terms of speed, Fitzke et al. (Fitzke et al., 2021) proposed a high-throughput deep learning system that could detect mitosis on the WSI in 0.27 minutes per slide. Most importantly, when compared to human expert evaluation, their system resulted in a change in tumor grading in some cases.

Chapter IV

METHOD

In this chapter, we explain each component of our proposed pipeline in full detail. The organization of this chapter starts by explaining an overview of the proposed pipeline. Then, it delves into the detail of each component, starting from the rationale for each component, their advantage our the existing baseline and their mechanism, and some implementation details.

An overview of our pipeline is shown in Figure 4.1. The pipeline is divided into four stages. First, a detection stage employs an object detector to propose the location of potential candidates for the mitotic figures in the WSI by splitting the WSI into many small sliding window grids and performing inference on each of them. Following that, a window relocation algorithm discards all low-quality false positive predictions around the image's border and re-evaluates them at the newly created patch. The extracted object is then getting refined in an object center adjustment stage to be more aligned with the image patch center. Finally, a classification stage re-evaluates the confidence of every object by observing each cell separately. An additional data sampling technique is further used to enhance the classification stage by selecting additional informative training examples from the WSI using disagreement between the detection and classification stages as a criterion. Each subchapter describes each stage in full detail.

4.1 Detection stage

The detection stage is the first stage of the pipeline and is in charge of locating every mitosis cells in the image. It is a deep object detector that takes an image as input and returns a set of bounding boxes $\{(x_1, y_1, w_1, h_1, S_1), ..., (x_n, y_n, w_n, h_n, S_n)\}$, where each tuple in the set represents the predicted object's center, width, height, and positive object confidence, in that order. Because of the sheer size of the WSI, the slide is segmented into smaller patches in a sliding window fashion. The sliding window algorithm breaks down the slide with the dimension of $W \times H$ into $\lceil \frac{W}{K} \rceil \times \lceil \frac{H}{K} \rceil$ image patches (window) with the window size of $K \times K$. The detection stage then infers on each patch to determine the location of the mitotic figures within it. To train the detector, we use the CCMCT and CMC baseline data sampling strategies but slightly alter the training process by sampling training images beforehand rather than querying them on the fly to stabilize the training process.

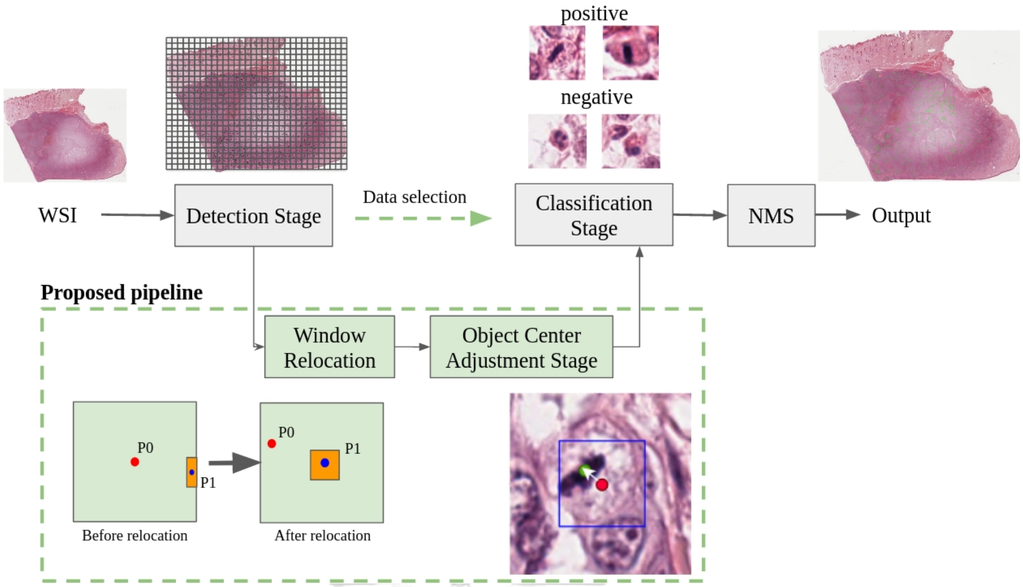


Figure 4.1: A summary of our proposed pipeline. The green dashed box highlights our contributions. Our pipeline now includes two more stages: window relocation and object center adjustment. Window Relocation is used to remove unnecessary low-quality predictions from the sliding window borders. The object center adjustment stage is in charge of aligning the estimated positive class object's center from the detection stage to the image patch center. In the classification stage, data selection is used to select additional training samples from the vast area of the WSI in order to improve the model performance.

Despite being able to perform an inference on the WSI, the use of the sliding window algorithm also cause a large quantity of low-quality false positive predictions to be generated during the inference process. This is due to the fact that the object near the window boundary may be partially split into multiple objects in multiple sliding windows, causing then to become multiple poor-quality false positive bounding boxes. Therefore, an overlapping sliding window is often used to alleviate this issue by allowing the windows to overlap with the previous one. As a result, partially split boxes around the window border are fully covered, but redundant predictions are also still being produced in excess, but fewer in quantity. As a result, non-maximum suppression (NMS) is employed as a post-processing technique to eliminate redundant objects. NMS suppresses the bounding box when there exist nearby bounding boxes of which an intersection over union (IOU) is over a certain threshold and has higher confidence. By removing the low-quality, low-confidence boxes while keeping the high-quality, high-confidence ones, the use of NMS greatly reduces the likelihood of false-positive predictions. Despite the advantage, the overlapping windows increased the number of patches to perform inference to $\lceil\frac{W}{K(1-\sigma)}\rceil \times \lceil\frac{H}{K(1-\sigma)}\rceil$, where σ is an overlapping ratio. Moreover, though

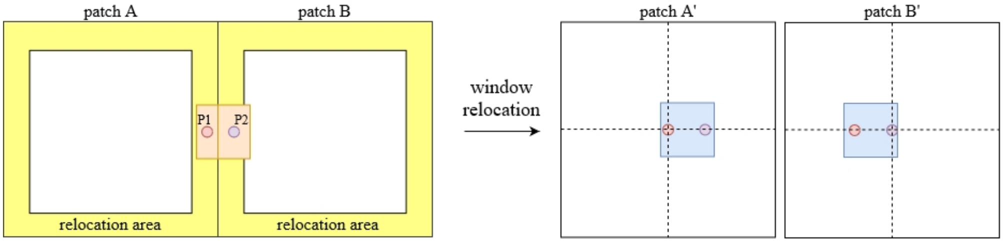


Figure 4.2: An illustration of the window relocation algorithm. Within the non-overlapping sliding windows A and B, there is an object of interest (orange box). As a result, patches A and B each produce a low-quality box with the center at points P1 and P2, respectively. Because they are in the relocation area, both centers are viable candidates for relocation. The window relocation algorithm begins by discarding the two boxes in both patches. Then, patches A' and B' are created from scratch, with patch center points P1 and P2. The newly created patches are then fed into the detector, which returns two blue boxes.

the problem is mitigated, this method does not guarantee good performance at the borders.

4.2 Window relocation

Window Relocation is a simple algorithm for removing poor quality predictions near the sliding window border. This method aims to eliminate the overlapping sliding window's two main flaws. The first flaw is that when the IOU is not high enough for NMS to suppress, poor quality predictions around the window border persist, resulting in an increased number of false positives during the final evaluation. Another flaw is that the computation resource is wasted when the window and its surroundings do not contain any object, which is especially problematic for this task because mitotic figures are frequently sparsely distributed across the WSI.

The window relocation algorithm is depicted in Figure 4.2. Window relocation addresses both issues in three steps. First, a relocation area is defined around the border of each patch (the yellow area in Figure 4.2). All positive objects whose center resides in the area are then discarded. After that, for each discarded object, the new window whose center is the center of the discarded object is created (patch A' and B' in Figure 4.2). Finally, the detector performs inference on the newly created windows. By following these steps, the object's focus is shifted from the window border to the newly created window center. This algorithm provides us with three advantages. First, it would reduce the poor quality predictions around the window border as most of them are removed. Second, having a relocated object positioned at the window center results in a more consistent detection result. Third, this method does not increase computation costs in areas with no objects. Though this method may result in redundant predictions, it has little impact on the overall pipeline because the new consistently produced boxes can be easily removed using NMS.

Next, we define a clear definition of a relocation area. If the condition below is met, the ith object in each window may be considered to be in the relocation area.

$$(min(x_i, y_i, K - x_i, K - y_i) \leq M) \wedge (S_i \geq D)) \tag{4.1}$$

In other words, the center of the object that is fewer than equal to M pixel from the window border in any axis and has higher positive object confidence than D is in a relocation area and is eligible for window relocation.

M is a hyperparameter determining a distance threshold from a window border, affecting the number of re-observed objects. If M is set to a low value, window relocation would act as a non-overlapping sliding window. In contrast, a high value of M would allow more objects to be re-scored. Setting M to a high value would also come with a trade-off because it would result in an increased computation cost since the detector has to re-infer more objects. Nevertheless, the use of window relocation is expected to have less computation costs than the overlapping sliding window. This is because it would only try to re-inference the objects around the window border, and the objects in the datasets are often sparse. D is a positive confidence threshold used by the detector to discard obvious negative objects. For both datasets, it is set to 0.05.

Since we know beforehand during the annotation process that the mitotic figure often has a form of circular shape with a radius around 25 pixels, we also follow this assumption and set M to 25 pixels. It should be noted that this method would be ineffective for general object detection tasks because the object shape could not be known ahead of time.

4.3 Object center adjustment Stage

Although many false-positive samples around the sliding window's border are reevaluated after window relocation, poor-quality bounding boxes continue to cause input inconsistency at the classification stage. Because of the input translation variance, the extracted object may not be positioned at the image patch center, resulting in classification stage performance degradation. As a result, after window relo-

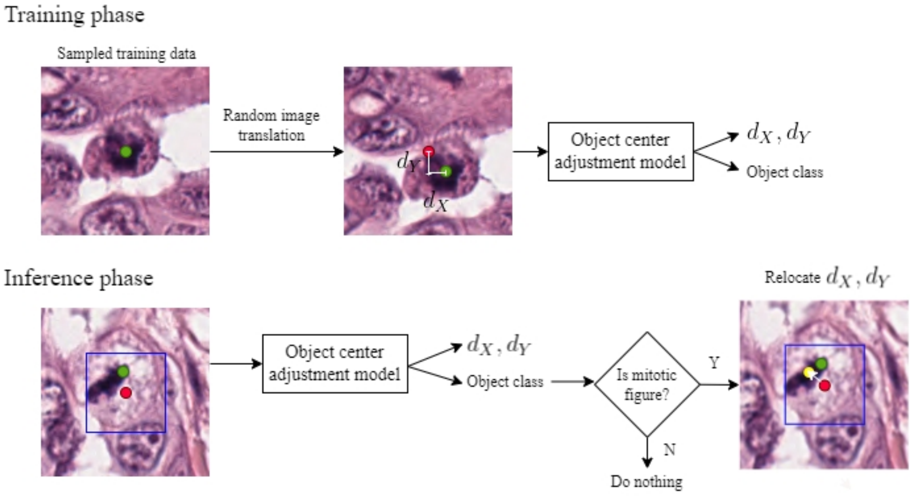


Figure 4.3: An illustration of the object center adjustment stage. The object center adjustment stage learns to estimate the distance between the extracted patch center (red dot) and the true positive class object center (green dot) and its class. The model estimated the location of the object center (yellow dot) during inference and generates a new image patch at the predicted location if the predicted object is recognized as a positive class. The blue box is a detection stage-predicted bounding box.

cation, we introduce an object center adjustment stage as a refinement process to reduce position inconsistency of the positive class objects in the image patch by making the object center more aligned to the image patch center to reduce input translation variance. The object center adjustment stage is a model that learns to find the positive object's center by estimating the distance between the image patch center and the ground truth positive class object center. Then, during an inference, it predicts the object center location and creates a new patch with the predicted location as the center if the object class is positive. Because the concept of object center is ambiguous for non-cell background and broad tissue texture areas, the negative class objects are not adjusted. The object center adjustment stage is depicted in Figure 4.3.

To train the model to estimate the position of the object center, we generate the data representing the object center at different locations in the patch as an input to the model. The generation process starts by randomly sampling positive and negative objects from the dataset and extracting them in an image patch. By doing so, the image center of the sampled object is always at the same position as the ground truth object center. Then, random geometric transformations, which are random image shifting, flipping, rotation, are applied to the sampled image. As a result, the ground truth center is shifted from the image center by (d_X, d_Y) pixels. After the image is transformed, the model learns to predict the position of the object center by predicting (d_X, d_Y). The value of d_X, d_Y is drawn from a normal distribution and is limited to a small value ($d_X, d_Y \leq 12$ pixels) because we assume that the center of the predicted object should be close to the ground truth object center.

Since the objective of this stage is to relocate the center of the positive object, the class of the object has to be known beforehand, which is not practical in a real-world situation. Therefore, the object class has to be inferred from the model. We could straightforwardly obtain the class by using object confidence from the detection stage. The detected object could be inferred as a positive class when the confidence is above a certain threshold. However, using detector confidence might not be ideal as the confidence produced by the poor bounding boxes might be inaccurate. Therefore, we added an auxiliary task for the object center adjustment stage to classify the object class. Since the input to this stage is just an extracted patch, it allows the model to observe a single object at a time, removing an unnecessary distraction from other objects. As a result, the confidence produced by this improvement should be superior to the detector confidence because it inherits the advantage of the limited observation like the classification stage, and it also has information of the annotated object center.

The object center adjustment stage is a CNN that outputs two prediction heads: the main regression head to estimate the distance from the image center to the ground truth center (d_X, d_Y), and the auxiliary classification head to predict the object class. The model is optimized using relocation loss L_{rel} as shown below.

$$L_{rel} = \lambda_{reg} L_{reg} + (1 - \lambda_{reg}) L_{cls}. \tag{4.2}$$

The relocation loss L_{rel} is a combination of the regression loss L_{reg} and classification loss L_{cls} weighted by the parameter λ_{reg}. The classification loss is a standard cross-entropy loss calculated between the predicted and the ground truth object class. The regression loss is an $L1$ loss calculated between the predicted and the ground truth object center distance. To prevent regression noise, the regression loss calculation is ignored when the ground truth class is negative. An illustration of the building block of the object center adjustment stage is shown in Figure 4.4.

During inference, the model receives an extracted object as input and returns the object class and location of its center as an output by estimating the distance from the object center to the patch center. If the predicted object confidence exceeds a certain threshold, the object is considered positive, and a new patch with the predicted location at its center is generated. If the object's confidence is lower than the threshold, the model does nothing.

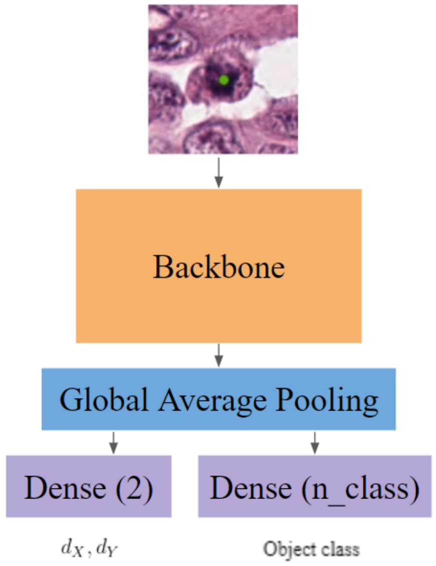


Figure 4.4: An overview of an object center adjustment stage. An illustration of the building block of the object center adjustment stage. The model outputs two prediction heads: a relocation head for predicting the distance from the image center, and a classification head for identifying the object class.

4.4 Classification stage

After the object center adjustment stage, the extracted object's center moves closer to the patch center and is ready to be fed to the classification stage. A classification stage is a model that looks similar to the object center adjustment stage but functions differently. This stage, in contrast to the previous one, is a CNN that only outputs a classification score. The classification stage takes an extracted object from the object center adjustment stage as input and returns the confidence of the object. This stage could be argued to be redundant because the object center adjustment stage could also return the confidence. However, t he main difference between this stage and the previous one is that the object is always centered in the image. This reduces the significance of having the model capture object translation variance. As a result, data augmentation strategies that could change the location of the object center are not used during training, resulting in improved training stability and recognition performance.

This stage's training procedure is similar to the object center adjustment model. First, positive and negative objects are sampled at random from the dataset in a separate area. The samples are then augmented and fed to the classifier, which predicts the object confidence. We follow DeepMitosis (Li et al., 2018) for the final object confidence calculation. The final object confidence S is weighted between the confidence produced by the detection stage S_{det} and the classification stage S_{cls}

using the weight ω as shown below.

$$S = \omega S_{det} + (1 - \omega) S_{cls}. \tag{4.3}$$

4.5 Active learning data selection

Even though the proposed pipeline performs well, the dataset is still underutilized. This is because the classification stage only looks at annotated objects and ignores the vast of majority of the unannotated areas. DeepMitosis (Li et al., 2018) addressed this issue by employing the detector to extract image regions from the original WSI in order to train the classification stage. However, in a large-scale dataset, this method became less effective because it generated an enormous number of objects from the negative class of WSIs. Inclusion of these additional data would result in not only a severe class imbalance, but also a problem with negative class uninformativeness. Therefore, we propose that only the informative subset of proposed objects should be chosen based on active learning approach.

We use an L1 distance between the detector's and classifier's positive class confidence to quantify the informativeness of a proposed object. This criterion provides us with two benefits. First, it would encourage the classifier to learn from the detector, which is generally better at filtering out negative objects. Second, it discourages the selection of noisy annotations because many objects in the positive class may not have been annotated as such. Both the detector and the classifier would return high positive class confidences in these cases and discard them. In this work, we select the top N (N = 20,000) negative objects with the highest informativeness as additional queries for retraining the classification model.

Chapter V

EXPERIMENTAL SETUP

This chapter explains the experimental setup used in our work, starting from the characteristic of the datasets used for the performance benchmarking. Then, the detailed training configuration of each component of the proposed method on each dataset, including other classification and detection algorithms used in this study. After that, we inform about the data sampling process and augmentation strategies for each model component. Finally, we define every quantitative evaluation protocol used in our thesis.

5.1 Dataset

Canine Mammary Carcinoma (CMC) Dataset

The first dataset chosen for benchmarking of our method was the CODAEL variant of the CMC (Aubreville et al., 2020a) dataset, a database of mitosis cells residing in dog mammary tumors. The prominent characteristic of this dataset was the availability of a complete mitotic figure annotation on the WSI level using algorithm-aided annotation and the consensus of experts. In addition, hard negative cells (mitosis figures lookalikes) were also annotated within the dataset, allowing the model to have more learning context. The CMC dataset contained an annotation of 13,907 mitotic figures on 21 WSIs, of which 7 of them were held out for testing. The CMC dataset consists of two classes: Mitosis, and Nonmitosis. The samples were collected from the Institute of Veterinary Pathology, Freie Universität Berlin, Berlin, Germany, and scanned using Aperio ScanScope CS2, Leica at a resolution of 0.25 microns per pixel (400X). Figure 5.1 shows an example of the scanned whole slide image from the CMC dataset.

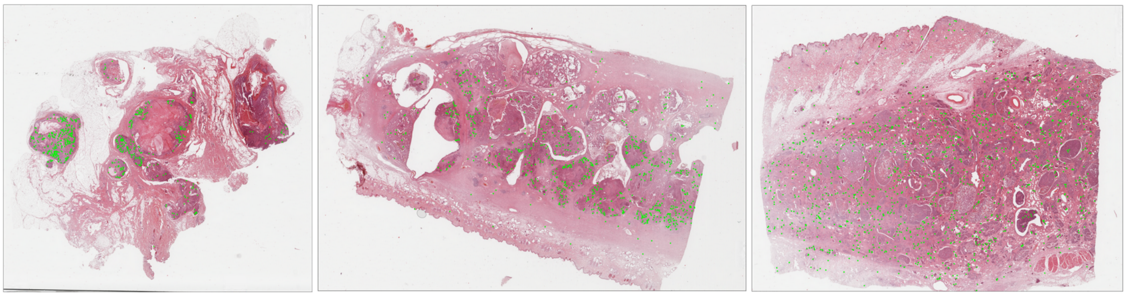


Figure 5.1: Example of the scanned image in the CMC dataset. Green dots indicate the position of mitotic figures.

Canine Cutaneous Mast Cell Tumor (CCMCT) Dataset

The second dataset chosen for benchmarking of our method was the ODAEL variant of the CCMCT (Aubreville et al., 2019) dataset, a database of mitosis cell resided in the most common skin tumor on dog. This dataset also provided a complete mitotic figure annotation on the WSI level using algorithm-aided annotation and the consensus of experts. Similar to the CMC dataset, hard negative objects (mitosis figures lookalikes) were also annotated with the inclusion of granulocytes and mast cells. The CMC dataset contained an annotation of 44,880 mitotic figures on 32 WSIs, of which 11 of them were held out for testing. The CCMCT dataset consists of four classes: Mitosis, Mitosis-lookalike, granulocytes, and mast cells. The samples were also collected from the Institute of Veterinary Pathology, Freie Universität Berlin, Berlin, Germany, and scanned using Aperio ScanScope CS2, Leica at a resolution of 0.25 microns per pixel (400X).

5.2 Detection stage

We used the same data preparation strategy as for the CCMCT and CMC baselines. 50% of the cropped patches were obtained at random from the training slide. 40% of the cropped images contained at least one mitotic figure. 10% of cropped images contained at least one mitotic figure-lookalike (class NonMitosis in the CMC dataset and Mitosis-lookalike in the CCMCT dataset). According to this strategy, 5,000 cropped patches were chosen from each training slide, for a total of 105,000 and 70,000 patches for the CCMCT and CMC datasets, respectively. With more data sampled using this strategy, we did not see a significant improvement in detection performance.

The training was conducted using Faster R-CNN (Ren et al., 2015a) with ResNet-50 (He et al., 2016) as a network backbone with an input training resolution of 512×512. The network backbone was initialized using ImageNet pre-trained weights (Deng et al., 2009). We did not modify the base detection algorithm except for the number of output classes. We sampled 5,000 image patches from each training slide using the same data sampling strategy as the baseline. The training framework was based on an object detection framework MMDetection (Chen et al., 2019). The model was trained with a batch size of 4 and the SGD optimizer. The model was trained for 8 epochs with an initial learning rate of 10^{-3}, which was divided by 10 after 5 and 7 epochs. During training, only random flip and standard photometric augmentation were used.

Apart from YOLOF, every detection model was trained using the same training schedule and data augmentation strategy as described in Faster R-CNN in the main methods. For YOLOF, we trained the model with a batch size of 16 and followed the original works' augmentation strategy. YOLOF was trained with an initial learning rate of 1.5×10^{-2} for 16 epochs which were divided by 10 after 10 and 14 epochs.

Our re-implementation of the training process of the detection stage is similar to the baseline works (Aubreville et al., 2019, 2020a) except for a few details. To improve the reproducibility and speed of the training processes, we pre-sampled a fixed number of training windows per slide instead of querying them on the fly. Second, whereas the original work used a complex training schedule with a super-convergence scheme and early stopping based on validation performance, we chose a simple training schedule with a fixed number of epochs and a standard step decay. This modification also improved the training's stability and reproducibility. The detection stage's input size was also increased from 256×256 to 512×512.

Despite the mentioned discrepancy in the detection stage, they have little to no impact on the performance of the whole pipeline. We demonstrated this claim by performing inference with our whole pipeline on the prediction result from the detection stage of the baseline works instead of our trained detector. This yielded 82.3% and 81.6% test F1 on the CCMCT and CMC datasets, respectively, which is very close to the one reported in Table 6.1 (82.4% and 81.6% in row 7). Please note that the comparison was conducted without the window relocation stage because we only had access to the prediction output and not the actual baseline detector model.

5.3 Object center adjustment stage

The training was conducted using EfficientNet-B4 (Tan and Le, 2019) as a network backbone with an input training resolution of 128×128. The network backbone was initialized using ImageNet (Deng et al., 2009) pre-trained weights. The model was trained using a batch size of 64 and Adam as an optimizer. The model was trained with an initial learning rate of 10^{-4} for 30,000 iterations which were divided by 10 after 22,500 and 27,000 iterations. λ_{reg} was set to 0.95 for every experiment. During training, random image geometric and standard photometric augmentation were used. The positive class threshold was set to 0.2 for the CMC dataset and 0.5 for the CCMCT dataset.

5.4 Classification stage

The training was conducted using EfficientNet-B4 (Tan and Le, 2019) as a network backbone with an input training resolution of 128×128. The network was initialized using ImageNet pre-trained weights. The model was trained on the CCMCT dataset with an initial learning rate of 5×10^{-4} for 30,000 iterations, which were divided by 10 after 22,500 and 27,000 iterations. The model was trained on the CMC dataset without data selection with an initial learning rate of 5×10^{-4} for 15,000 iterations, which were divided by 10 after 10,000 and 13,000 iterations. The model was trained on the CMC dataset with data selection with an initial learning rate of 5×10^{-4} for 24,000 iterations, which were divided by 10 after 15,000 and 21,000 iterations. Random image geometric and standard photometric augmentation except for random translation were used during training.

For other classification networks, they were trained using the same training schedule and data augmentation strategy as described for EfficientNet-B4 in the main methods, except for ConvNext-S. When ConvNext-S was trained on the CMC dataset, the initial learning rate was set at 5×10^{-4} for 16,000 iterations and were divided by 10 after 8,000 and 14,000 iterations.

5.5 Data augmentation strategies

Table 5.1 shows a detailed list of augmentation strategies of the object center adjustment and classification stages. Random rotation was still allowed in the classification stage because the relocated object center patch can still rotate.

Table 5.1: List of augmentation strategy of the the classification and object center adjustment stage.

Augmentation strategy	**Classification stage**	**Object center adjustment stage**	**Intensity**
	probability		
Random flip	0.5	0.5	-
Random brightness	0.5	0.5	(0.8, 1.2)
Random contrast	0.5	0.5	(0.8, 1.2)
Random gaussian blur	0.25	0.25	(3×3), (5×5)
Random hue	1	1	(-0.1, 0.1)
Random rotation	1	1	(-90, 90)
Random translation	0	1	$d_x, d_y \sim N(0, 6^2)$

5.6 Quantitative evaluation protocol

The performance of our proposed system is quantitatively evaluated using F1 as primary metrics for mitotic figure recognition. The TP, FP, and FN refer to a true positive, false positive, and false negative, respectively. The predicted objects are considered a true positive when the center of the predicted object is fewer than 25 pixels from the ground truth object center. Otherwise, it is considered a false positive. If there does not exist a predicted object with 25 pixels from the ground truth object, the object is considered a false negative. The F1 score is calculated using the following formula:

$$F1 - score = \frac{TP}{TP + \frac{1}{2}(FP + FN)} \tag{5.1}$$

The performance of the proposed pipeline on mitotic count prediction was evaluated using mean absolute error (MAE), and mean absolute percentage error (MAPE). The mean absolute error is an average absolute L1 distance between the predicted and ground truth mitotic count formulated as follow:

$$MAE = \frac{1}{n}\sum_{i=1}^{n} |y_i - \bar{y}_i| \tag{5.2}$$

where $y_i, \bar{y}_i, n$ refers to predicted mitotic count at slide i, ground truth mitotic count at slide i, and total number of test WSI, respectively. On the other hand, the mean absolute percentage error instead use proportion of error calculated as shown below:

$$MAPE = \frac{1}{n}\sum_{i=1}^{n} \left|\frac{y_i - \bar{y}_i}{y_i}\right| \tag{5.3}$$

Chapter VI

RESULTS

In this chapter, we report the performance of our proposed compared to the established baselines on the cell level which measures the competency of the proposed pipeline and image level to examine the generalizability across the whole WSI. It also includes multiple ablation studies to examine the robustness of our method when some components are removed or altered. The report also includes findings found using the proposed pipeline and an experiment under a realistic setup.

We verified the effectiveness of the proposed method on the CCMCT (Aubreville et al., 2019) and CMC (Aubreville et al., 2020a) datasets. We followed the prior works Aubreville et al. (2019) and used F1 (%) as a primary evaluation metric with precision and recall as a secondary metric and using the same train-test split. We reported an average of three along with standard deviations. The models used for evaluation were the checkpoints at the last training step.

The performance of our method was summarized in 6.1. Ultimately, the use of the proposed pipeline resulted in a significant increase in pipeline recognition performance from 82.0% to 83.2% on the CCMCT dataset and 77.5% to 82.3% on the CMC dataset. Data selection and object center adjustment stage were the main contributing factors for the pipeline improvement since they contributed 2.6% and 3.4% absolute performance improvement. The findings suggested that consistency of input and exposure to additional unannotated data during the classification stage were critical for performance improvement. Figure 6.1 depicts the qualitative improvement in WSI inference following the addition of each technique to the pipeline. After each stage, more mitotic figures (green dots) were correctly classified.

Despite a change in the detection algorithm, namely from Faster R-CNN-ResNet50 to RetinaNet-ResNet18 (Lin et al., 2017b), Cascade R-CNN-ResNet50 (Cai and Vasconcelos, 2018), and YOLOF-ResNet101 (Chen et al., 2021), the benefits of our method could still be clearly observed (Table 6.1). For instance, the detection performance gap between RetinaNet-ResNet18 and other algorithms was as large as 11.4% F1 on the CMC dataset; however, the whole-pipeline performance gap was reduced to only 0.5% F1. This also indicates that our method enables the use of a fast detection algorithm to accelerate inference time with minor classification performance cost.

Interestingly, the results in Table 6.1 also showed that the model that performed better on the COCO detection dataset (Lin et al., 2014) did not necessarily perform better on mitosis detection. This could be because mitotic objects, unlike objects in other datasets, are typically the same size, sparsely populated, and rarely overlap with each other. This pattern was also seen at the classification stage. As shown in Table 6.2, classification networks that performed well on the ImageNet dataset did not necessarily perform better on the mitosis detection task. ConvNext-S, for example, outperformed a larger ConvNext-B on both mitosis datasets, despite having the same architectural design. Hence , the search for a suitable detection and classification model for mitosis tasks is still ongoing area of research.

The method's performance was then evaluated solely on the detection part. The window relocation stage consistently improved detection performance on both CCMCT and CMC datasets, as shown in Table 6.3. The object center adjustment stage, on the other hand, had little to no effect. This is because the object center adjustment stage only slightly shifted the object center and had no effect on the detection's confidence score.

The mispredictions produced by our pipeline were then investigated by observing false-positive errors and categorizing them as easy or hard errors. The hard errors are hard-negative objects that are misidentified as positive classes, whereas the easy errors are misidentifications of the positive class with a non-hard negative object or background image. Figure 6.2 depicts a visualization of our method's false-positive errors. When compared to the baseline, our method significantly reduced the number of easy false positive predictions. Nonetheless, the distinction between positive and hard-negative samples still remained unresolved. This indicated that variance in input translation was not the only cause of the confusion between hard-negative and positive objects.

The subchapters below study an effect of the individual approach, an end-to-end evaluation, and a discussion of our method. Faster R-CNN-ResNet50 and EfficientNet-B4 were used as base detection algorithms and classification networks, respectively.

6.1 Effect of object center adjustment stage

In this subchapter, we study the effect of the object center adjustment stage on the proposed pipeline. First, we demonstrate that the presence of this stage improves the proposed object center quality. Then, to confirm our design choice, we provide ablation studies. $omega$ was set to zero for each experiment, and window relocation was not included.

Table 6.1: The test F1 (%) performance of the proposed method evaluated on the CCMCT and CMC datasets. ± denotes standard deviation. The baseline numbers in the CMC dataset were obtained from the erratum in their Github.

Detector	**Method**	**CCMCT F1(%)**	**CMC F1(%)**
RetinaNet	Baseline (Detection stage) (Aubreville et al., 2020a, 2019)	62.8	72.6
RetinaNet	Baseline (Full pipeline) (Aubreville et al., 2020a, 2019)	82.0	77.5
RetinaNet	Detection stage	68.4 ± 0.5	59.0 ± 0.9
	+ Classification stage, (reproduced baseline, $\omega = 0$)	79.4 ± 0.2	77.3 ± 0.2
	+ Data selection	81.2 ± 0.2	80.3 ± 0.1
	+ Object center adjustment	82.3 ± 0.1	81.5 ± 0.1
	+ Weight confidence ($\omega = 0.4$)	82.4 ± 0.1	81.6 ± 0.1
	+Window relocation	**82.6 ± 0.1**	**81.8 ± 0.1**
Faster R-CNN	Detection stage	78.2 ± 0.5	70.4 ± 0.3
	+ Classification stage, (reproduced baseline, $\omega = 0$)	79.9 ± 0.3	78.4 ± 0.2
	+ Data selection	81.8 ± 0.1	80.3 ± 0.1
	+ Object center adjustment	82.5 ± 0.1	81.8 ± 0.1
	+ Weight confidence ($\omega = 0.4$)	83.0 ± 0.1	82.1 ± 0.1
	+Window relocation	**83.2 ± 0.1**	**82.3 ± 0.1**
Cascade R-CNN	Detection stage	75.8 ± 0.2	70.2 ± 0.6
	+ Classification stage, (reproduced baseline, $\omega = 0$)	79.9 ± 0.0	78.9 ± 0.1
	+ Data selection	81.7 ± 0.0	80.3 ± 0.1
	+ Object center adjustment	82.3 ± 0.1	81.3 ± 0.0
	+ Weight confidence ($\omega = 0.4$)	82.7 ± 0.1	81.5 ± 0.1
	+ Window relocation	**82.9 ± 0.1**	**81.9 ± 0.1**
YOLOF	Detection stage	69.4 ± 0.8	62.1 ± 0.1
	+ Classification stage, (reproduced baseline, $\omega = 0$)	79.8 ± 0.3	78.7 ± 0.6
	+ Data selection	81.2 ± 0.5	80.1 ± 0.4
	+ Object center adjustment	82.0 ± 0.1	81.2 ± 0.2
	+ Weight confidence ($\omega = 0.4$)	82.1 ± 0.1	81.1 ± 0.1
	+ Window relocation	**82.3 ± 0.1**	**81.4 ± 0.5**

Table 6.2: Effect of changing classification network on the performance of the whole pipeline. All models used Faster-RCNN-ResNet50 as a base detection algorithm. Our method still consistently improved the whole pipeline despite using different classification networks.

Classification network	**CCMCT Test F1(%)**	**CMC Test F1(%)**
ResNet-152	82.5±0.1	81.5±0.4
EfficientNet-B4	**83.2**±**0.1**	**82.3**±**0.1**
ConvNext-S (Liu et al., 2022)	82.4±0.1	81.7±0.1
ConvNext-B (Liu et al., 2022)	82.4±0.1	81.5±0.1

Table 6.3: Effect of the proposed method on the detection stage of the CMC and CCMCT dataset. The window relocation stage consistently improved the detection performance, while the object center adjustment stage had little to no effect.

Detection algorithm	**Method**	**CCMCT F1(%)**	**CMC F1(%)**
RetinaNet	Detection stage	68.4 ± 0.5	59.0 ± 0.9
	+ Object center adjustment	68.4 ± 0.5	59.0 ± 0.9
	+ Relocation stage	70.6 ± 0.4	64.1 ± 0.9
Faster-RCNN	Detection stage	78.2 ± 0.5	70.4 ± 0.3
	+ Object center adjustment	78.2 ± 0.5	70.4 ± 0.3
	+ Relocation stage	78.9 ± 0.3	72.4 ± 0.2
Cascade-RCNN	Detection stage	75.8 ± 0.2	70.2 ± 0.6
	+ Object center adjustment	75.8 ± 0.2	70.2 ± 0.6
	+ Relocation stage	76.9 ± 0.1	72.2 ± 0.5
YOLOF	Detection stage	69.4 ± 0.8	62.1 ± 0.1
	+ Object center adjustment	69.4 ± 0.8	62.1 ± 0.1
	+ Relocation stage	71.0 ± 0.6	65.1 ± 0.5

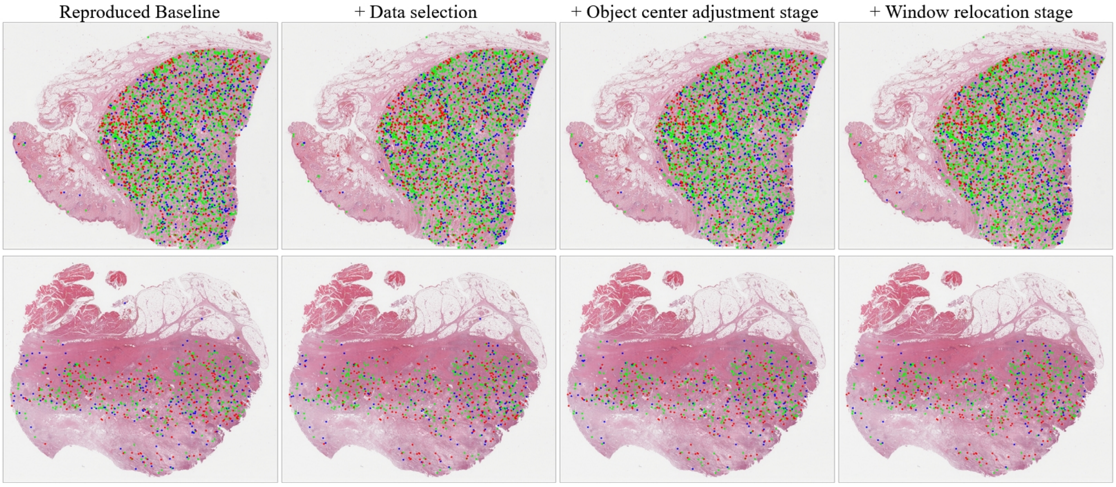
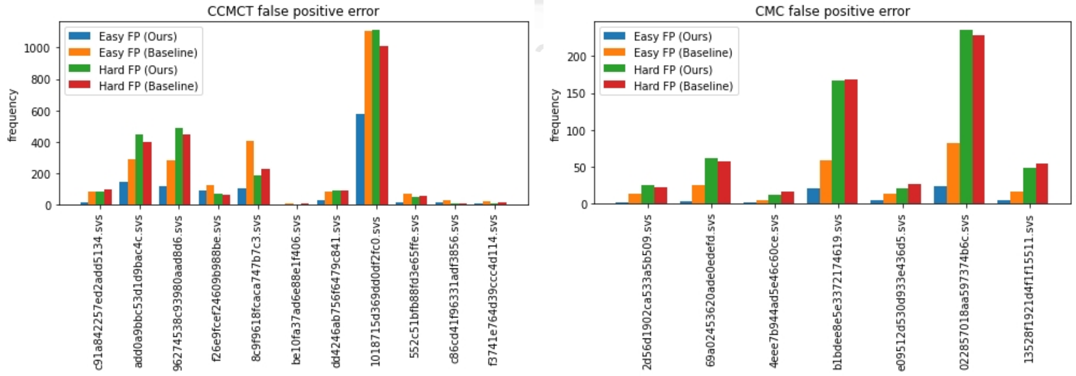


Figure 6.1: Example of whole pipeline detection results on the WSI. Red, blue, and green dots indicate false negative, false positive, and true positive, respectively. The detection threshold was set at the value that yielded the highest F1.



(a) (b)

Figure 6.2: Multiple Bar charts showing the frequency of easy and hard false positive (FP) errors on the CCMCT and CMC dataset. Our method greatly reduced the number of easy false positive predictions, yet confusion between positive and hard-negative samples still remained in high quantity.

One metric that can measure the performance of the object center adjustment stage is the distance between the patch center and the original location. This metric did not include false positives because they were irrelevant at this stage. On the CMC dataset, the object center adjustment stage reduced the average distance from 3.61 to 3.40. The results indicated that using the object center adjustment stage significantly reduces the input translation variance.

Next, we justify the exclusion of negative class in the regression loss and the presence of auxiliary head. Table 6.5 shows that the model performance reduced from 81.8% test F1 to 81.1% when the negative class was included in the regression loss. The results show that the ambiguity of the object center in the negative class object caused regression noise during training, resulting in lower performance. Furthermore, the auxiliary head improves model performance from 81.5 to 81.8 percent, demonstrating the importance of multi-task learning.

We also conducted ablation studies on the choice of pipeline design and the removal of data augmentation strategies that could change the location of the object center. Table 6.4 shows that translation augmentation improved the performance of the classification stage of the base pipeline. However, the object center adjustment training scheme is more efficient than data augmentation because it formulates the problem as a multi-task problem. We confirmed this by substituting an object center adjustment stage for a classification stage and producing object confidence with its classification head instead. It was discovered that by only using the object center adjustment stage, the overall pipeline performance improved from 80.5% to 81.3% on the CMC dataset. The performance was increased to 81.8% by further stacking the relocation and classification stages. However, having a translation augmentation in the stacked pipeline's classification stage degraded performance. The results also showed that when the translation variance of the object center was controlled, translation augmentation hampered performance.

6.2 Effect of window relocation

This subchapter sought to assess the impact of window relocation on the entire pipeline. The table 6.6 compares window relocation to the sliding window method. The use of overlapping sliding windows had no effect on pipeline performance because most of the overproduced samples could be removed using the object center adjustment stage and non-maximum suppression. On the CMC dataset, using window relocation improved the pipeline's performance over the non-overlapping sliding window and the overlapped one by 0.2% test F1 absolute improvement. The outcome suggested that some of produced errors could not be mitigated by solely

Table 6.4: The result of the ablation study on the importance of the object center adjustment stage conducted on the CMC dataset. The use of an object center adjustment stage outperformed the classification stage with translation augmentation. In addition, the removal of translation augmentation at the classifications stage was crucial for the performance improvement of the whole pipeline.

Method	**CMC test F1(%)**
Classification stage	80.3 ± 0.1
Classification stage w/ translation augmentation	80.5 ± 0.3
Object center adjustment stage	81.3 ± 0.1
Object center adjustment stage+Classification stage	**81.8** ± **0.1**
Object center adjustment stage+Classification stage, (w/ translation augmentation)	81.5 ± 0.1

Table 6.5: The result of the ablation study of the object center adjustment stage conducted on the CMC dataset. The use of an auxiliary head improved the stage performance while the inclusion of negative class for relocation loss resulted in reduced performance.

Negative class relocation loss	**Auxiliary head**	**CMC test F1(%)**
-	-	81.5 ± 0.2
✓	-	81.1 ± 0.1
-	✓	**81.8** ± **0.1**

using the center adjustment stage. This is because the overproduced object's center may be too far away for the object center adjustment stage to adjust back to the actual center. Furthermore, we found that window relocation only adds a small amount of inference time over non-overlapping sliding windows in a practical setting. This is due to the low density of mitotic figures in the WSI. Furthermore, unlike overlapping sliding windows, window relocation could ignore the majority of the background image because it did not contain any objects to begin with.

Since both window relocation and object center adjustment stage have a similar objective of improving poor quality predictions for the detection stage, we conducted an ablation study to observe the effect of each component separately. Table 6.7 shows a comparison of the two components on the CMC dataset. Window relocation improved the test F1 from 80.3% to 81.1%. Nevertheless, the performance was inferior to the object center adjustment stage, which achieved 81.8%. This is because window relocation mostly affects the objects positioned around the sliding

Table 6.6: A comparison of different sliding window algorithms on the CMC dataset. Window relocation outperformed overlapping sliding windows while incurring less computation cost.

Method	**CMC test F1(%)**	**Number of test inference window**
Non-overlapping sliding window	82.1 ± 0.1	211482 (+0%)
Overlapping sliding window	82.1 ± 0.1	261909 (+23.8%)
Window relocation	**82.3** ± **0.1**	217368 (+2.7%)

Table 6.7: A comparison between window relocation and object center adjustment stage on the CMC dataset. Window relocation could partially mitigate the problem of input translation variance.

Window relocation	**Object center adjustment stage**	**CMC test F1(%)**
-	-	80.3 ± 0.1
✓	-	81.1 ± 0.2
-	✓	81.8 ± 0.1
✓	✓	**82.1** ± **0.1**

window border.

6.3 Effect of data selection algorithm

In this chapter, we demonstrate that our informativeness criterion works well for this task. As a result, we provided a comparison of our method to three baselines. The first baseline is DeepMitosis (Li et al., 2018) query strategy, which queries every negative object proposed by the classification stage from the training slides. The second baseline is uncertainty sampling, a strong baseline in the Active Learning field (Settles, 2009). This method measures the uncertainty produced by the model as a selection criterion for data acquisition. We used entropy as an uncertainty measurement and used classification stage confidence to produce model uncertainty. The third baseline is K-Center-greedy (Sener and Savarese, 2018), a query strategy based on the core set approach. It aims to select the samples that provide the most coverage over the training distribution by minimizing the distance between a data point and its nearest chosen samples. We also follow their work by using the output after the last convolutional layer of the classification stage to represent the data point and L2 as a distance function. During the experiments, the window relocation and object center adjustment stages were not included and the data were collected using

Table 6.8: The effect of data selection algorithm on the performance of the pipeline on the CMC dataset.

Query method	**CMC test F1(%)**
Baseline (no query)	77.6 ± 0.2
DeepMitosis (query all)	80.0 ± 0.2
K-Center greedy	79.0 ± 0.1
Uncertainty sampling	79.8 ± 0.1
Disagreement (Ours)	**80.3** ± **0.1**

Table 6.9: The performance of the pipeline on the CMC dataset when varying the number of datapoints quired using the data selection algorithm.

Number of quiried datapoints	**CMC test F1(%)**
0 (no query)	77.6 ± 0.2
2,000	77.8 ± 0.3
5,000	78.4 ± 0.4
10,000	79.2 ± 0.2
20,000 (Ours)	**80.3** ± **0.1**
40,000	80.2 ± 0.1

the same classification model as an uncertainty estimator for every baseline.

The results of our experiment are shown in table 6.8. Our method outperformed DeepMitosis' querying strategy and Active Learning baselines, and every Active Learning baseline outperformed not selecting any data at all. The results supported our claim that overexposure to negative samples resulted in suboptimal performance but was still preferable to not querying any additional data at all.

Next, we examined the effect of the query size on model performance. The result in Table 6.9 indicated that the performance tended to increase as more datapoints were included in the labeled pool. However, its capability stagnated when over 20,000 samples were selected. This is because the criteria used for data selection were strictly based on informativeness. Thus, the sample became less informative as more data were queried, resulting in an overabundance of uninformative data and class imbalance.

6.4 End-to-End evaluation

We further evaluated our method in an end-to-end setting by comparing the mitotic count produced by our method to the ground truth for each WSI. We follow the definition of mitotic count in Meuten et al.(Meuten et al., 2016) by counting the number of mitotic figures in 10 high-power fields (HPF, 2.37 mm^2) with an aspect ratio of 4:3 surrounding the area of WSI with the highest density of mitotic figures. In other words, the HPF for calculating mitotic count was selected by identifying the rectangular window of size 7110×5333 pixels that contains the highest number of predicted mitotic figures (Bertram et al., 2019). Once an HPF for a WSI was selected, mitotic count was calculated under two settings: fully-automated and human-in-the-loop. Under the fully-automated setting, the number of predicted mitotic figures in the selected HPF was taken as the mitotic count. This setting mimics the situation where the models were used to obtain mitotic count without supervision. On the other hand, the number of annotated mitotic figures in the selected HPF was instead used as the mitotic count in the human-in-the-loop setting. This scenario simulates the situation in which the selected HPF is given to expert pathologists who can recognize the majority, if not all, mitotic figures. Furthermore, this setting emphasizes the model's ability to propose good HPF rather than its ability to predict individual mitotic figures. MAPE and MAE at the prediction threshold which yielded the lowest MAPE were reported in Table 6.10. The performances for the baseline method were calculated using the predictions provided in the authors' GitHub. This shows that our method significantly improved mitotic counts on both CCMCT and CMC datasets under both fully-automated and human-in-the-loop settings. Figure 6.3 shows the comparison of mitotic counts produced by our method and the baseline method on individual WSI. Our method clearly resulted in more accurate mitotic counts, especially when the mitotic density is high.

Table 6.10: The end-to-end performance of the proposed method evaluated on the CCMCT and CMC datasets. Our method consistently outperformed the baseline in both settings.

Dataset	**Method**	**Fully-Automated**		**Human-In-The-Loop**	
		MAPE	**MAE**	**MAPE**	**MAE**
CCMCT	Baseline	18.8	10.5	11.2	4.4
	Ours	**10.5**	**8.3**	**6.8**	**1.9**
CMC	Baseline	7.8	3.1	8.1	2.4
	Ours	**5.6**	**1.9**	**5.6**	**1.6**

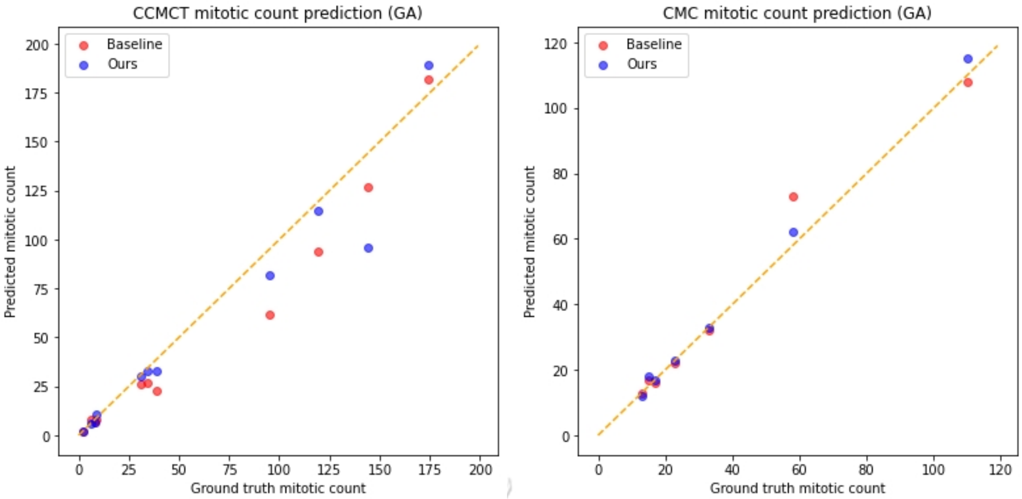


Figure 6.3: Scatter plots illustrating the predicted mitotic count and the ground truth on the CCMCT and CMC dataset under the fully-automated setting. Compared to the baseline, our method clearly changed the predicted MC when the object appeared in high density, though the effect become less noticeable on the slides with low mitotic figures.

6.5 Algorithm-aided mitotic count

We aimed to observe the effect of our pipeline on a real use case by having a human expert perform a mitotic count on the HPF proposed by our pipeline on the CCMCT dataset. First, the expert received a large rectangle box representing the HPF proposed by our method. After that, the expert had to draw a bounding box on every mitotic figure found in the proposed field, and a mitotic count is then calculated from the number of annotated objects. Bounding boxes predicted by the model were intentionally hidden from the expert in this setting. We used Sliderunner (Aubreville et al., 2018) as an annotation tool. An annotation time was also measured during the experiment. It was measured starting from the first to the last annotated mitotic figures in the slide. Figure 6.4 showed the result of our experiment. The mitotic count produced by the human expert is significantly lower than the one purely proposed by the model when the ground truth when the mitotic figures appeared in high density. The relation between an annotation time and the ground truth mitotic count followed a linear trend. Surprisingly, the annotation time dropped sharply when the ground truth mitotic count was above a certain value. We believed that the error came from the fact that the diagnosis would not change even if more mitotic cells were to be counted in the HPF with high MC. The result also indicated that only providing the HPF was not enough for an accurate mitotic count, and disagreement among the selection of the region of interest for the mitotic count was not the only issue.

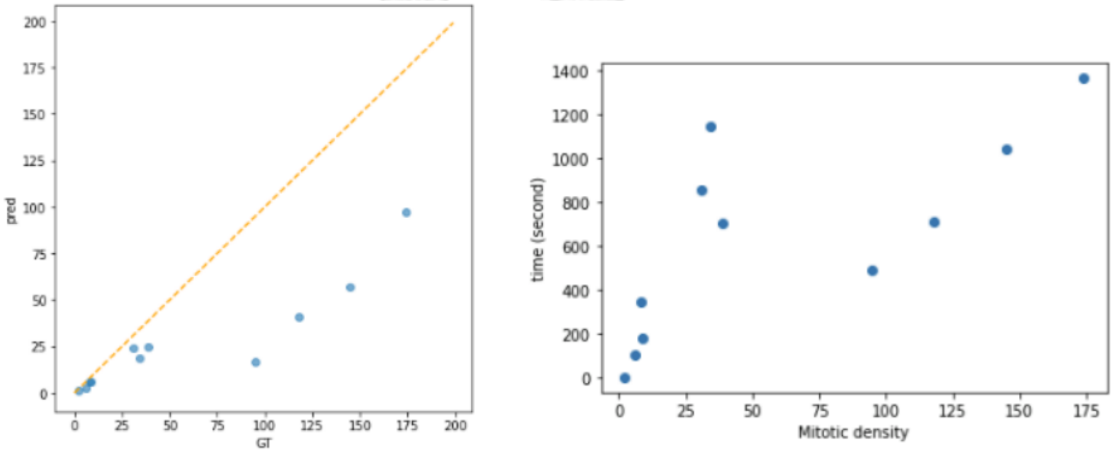


Figure 6.4: A result of algorithm-aided manual mitotic count. The MC produced by a human expert was significant lower than the one purely proposed by the model and the ground truth when the mitotic figures appeared in high density.

Chapter VII

DISCUSSION

This chapter provides insight into the results and finding reported in the previous chapter. First, it clarifies the difference between the proposed method and the former works. Then, it goes on each proposed component in detail on their advantage and shortcoming along with the quantitative result to verify the effectiveness of the proposed method. Lastly, it discusses some unresolved issues in the proposed pipeline, involving the quality of the dataset used for benchmarking and foreseeable problems in a real-world application when a pathologist is included in the loop.

Unlike most previous studies, which focused on improving mitosis detection pipeline performance by increasing model capability or data variety, our study looked at the interaction between different stages of the pipeline, a problem often overlooked in the field. Inconsistencies between the detection and classification stages, we argued, could lead to mispredictions due to a variety of mechanisms. First, poor-quality detection-stage bounding boxes that do not consistently center on object locations can confuse classification-stage training. Furthermore, a mismatch in the training distribution between the detection and classification stages will result in poor performance on out-of-distribution samples. To directly mitigate the aforementioned problems in a two-stage pipeline, three improvements were introduced: the window relocation stage, the object center adjustment stage, and improved data selection. The proposed pipeline outperformed previously reported baselines on two large-scale mitosis detection datasets at both individual mitotic object detection and mitotic count prediction (Tables 6.1, 6.10). The advantages of our method are also applied to other detection and classification algorithms (Tables 6.1, 6.2, and 6.3).

The window relocation stage, which is the first proposed component of our pipeline next to the detection stage, is a straightforward algorithm that enhances detection around a sliding window's boundary by re-inferencing the objects that are close to the border. It should be noted that this approach is more effective than an overlapping sliding window since only the areas around the window border were reobserved rather than creating additional windows. Thus, overall detection inference time was significantly decreased as a result (Table 6.6). In addition, window relocation also swaps out the poor predictions that were placed around the window's edge with newly created patches that solely focused on the objects. Figure 7.1 provides illustrations of instances where window relocation produces superior bounding boxes. However, the effectiveness of this approach significantly depends on the presumptions that mitotic figures were sparsely populated and around the same size on WSI, which are not always true in general object detection datasets.

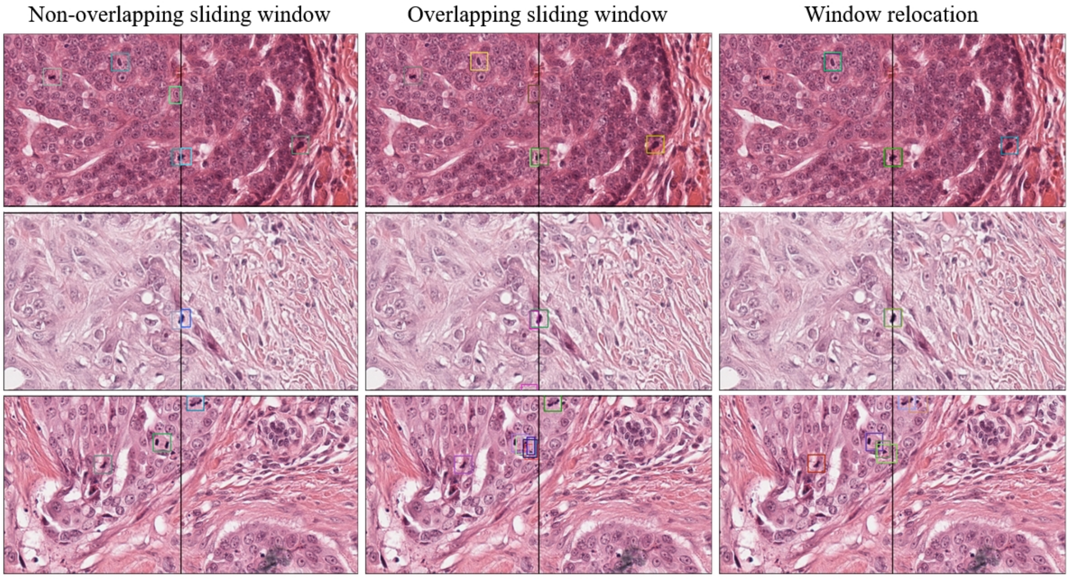


Figure 7.1: Illustrations of instances where window relocation produces superior bounding boxes. Black lines indicated sliding window boundaries, and bounding boxes were differently color-coded so that overlapping boxes were more visually distinguishable.

The object center adjustment stage is the second proposed component of our pipeline. This stage is in charge of moving the predicted object's center from the detection stage closer to the actual object's center, reducing translation variance for the classification stage. Translational variance would not normally be a significant issue in image classification. However, because mitotic figures are frequently found in close proximity to background objects, a slight shift in the input image patch is enough to confuse the classification model between positive and background objects. Figure 7.2 shows that a three-pixel shift can cause the classifier's confidence to drop from around 0.9 to 0.2, and that the object center adjustment stage is critical for stabilizing the classifier's confidence. The object center adjustment also produced better results than the conventional translation augmentation (Table 6.4).

Even though the object center adjustment stage (Figure 7.3(a)) may successfully shift the predicted object centers closer to the actual object centers, there were some inevitable errors (Figure 7.3(b)). A common cause of misalignment corresponds to mitotic objects in the late telophase stage during which the two daughter cells looked like two separate mitotic figures. This pattern made the model centers on one of the daughter cells rather than the actual center in the middle of the two cells. In other cases, the model was unable to make a proper adjustment because the initially anticipated object centers were too far from the actual center. Additionally, it should be mentioned that the object center adjustment method is suitable for mitosis detection due to the ability to precisely define and annotate the center of each unique cell. As a result, the technique should probably be able to apply to other pathology tasks that could clearly indicate the object center. However, general images contain many objects whose definition of a center may be ambiguous.

Finally, we reduced the training distribution mismatch between the detection and classification stages by querying additional training data for the classification stage based on the detector-classifier disagreement. This method offers an advantage over a query-all approach because the detector often generates too many more negative objects than positive objects, including a large number of uninformative samples. Furthermore, even though the aforementioned problems can be alleviated, the classifier will not be capable of learning the entire distribution of the WSI as long as it still relies solely on the detector to generate training samples.

Despite the improvement in the model's ability to recognize mitotic figures, its capability was still limited due to the lack of a gold standard label was still unresolved. Currently, most of the public datasets available were annotated by a consensus of broad-certified pathologists, which inevitably led to some mitotic figures not getting annotation, potentially leading to errors in the mitotic count. One prominent example for the former statement was the mitosis cell during prophase because the cell itself is difficult to identify with light microscopy (Donovan et al., 2020). This leads to the pathologists not counting them since they cannot be reliably distinguished, and therefore, consequently hindered the model for identifying all the mitotic figures in the whole slide image. It should also be noted that the bulk of performance gains through ReCasNet came from easy false positive objects (Figure 6.2). The issue of confusion between hard-negative and positive mitotic figures remains unresolved.

An improvement in mitotic count proposed by the model, even if performing better than the pathologist, also does not always directly lead to a more accurate diagnosis. This is because the availability of a method to achieve a more accurate mitotic count would violate the assumption of tumor grading protocol (Bloom and Richardson, 1957; Avallone et al., 2021) that is solely based on observation using a light microscopic image. The use of Phosphohistone H3 (PHH3), a cell proliferation marker, has been shown to increase the proportion of high-grade cancers during tumor proliferation assessment (van Steenhoven et al., 2020), and it is also likely that our model would have the same effect since the pathologists tend to undercount when the mitotic density in the slide is high. Despite the aforementioned downsides, the model should be able to assist the pathologist to distinguish different phases of mitosis cells, which might even change the grading protocol when AI-assisted tumor grading is widely adopted in the foreseeable future.

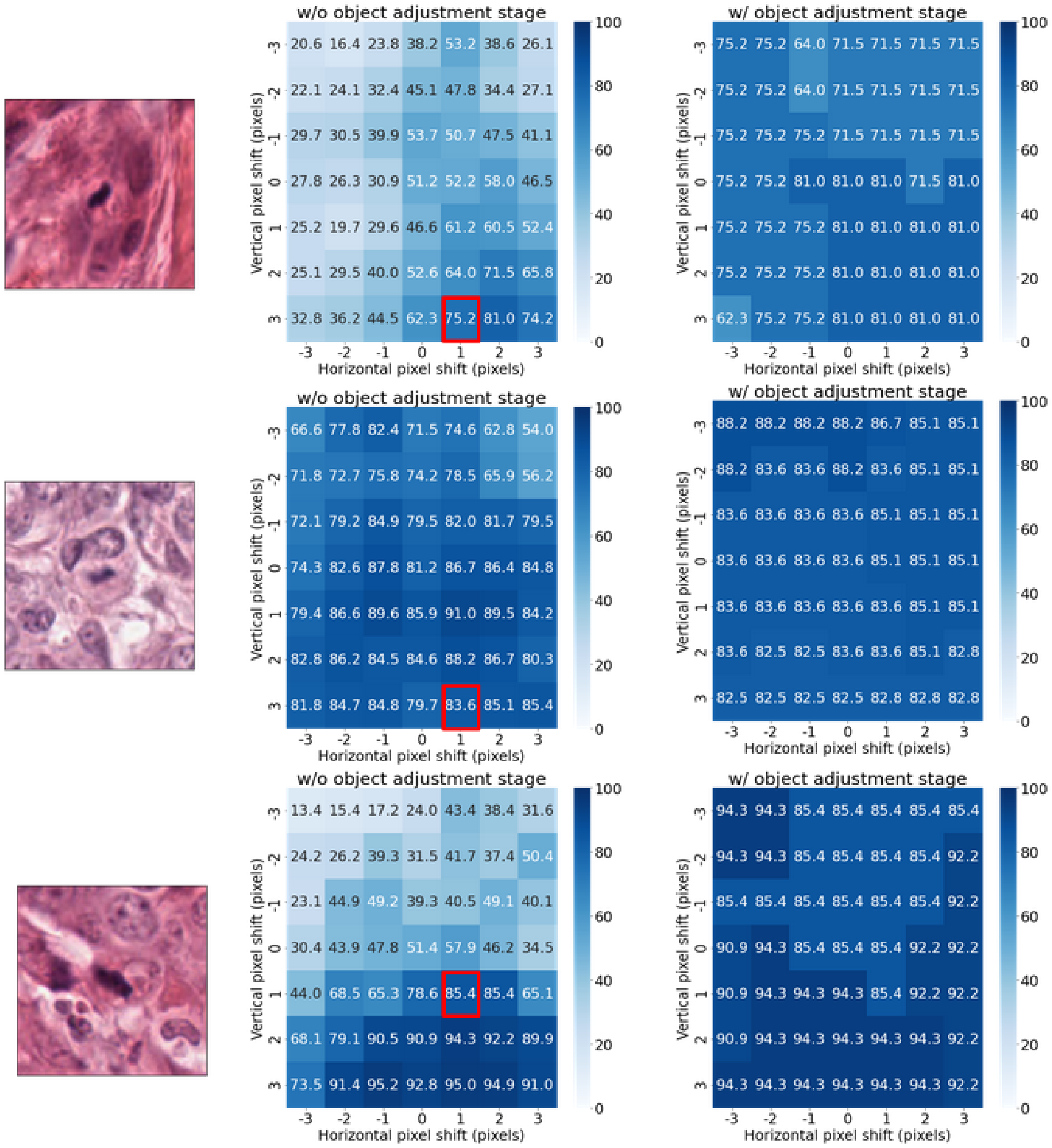
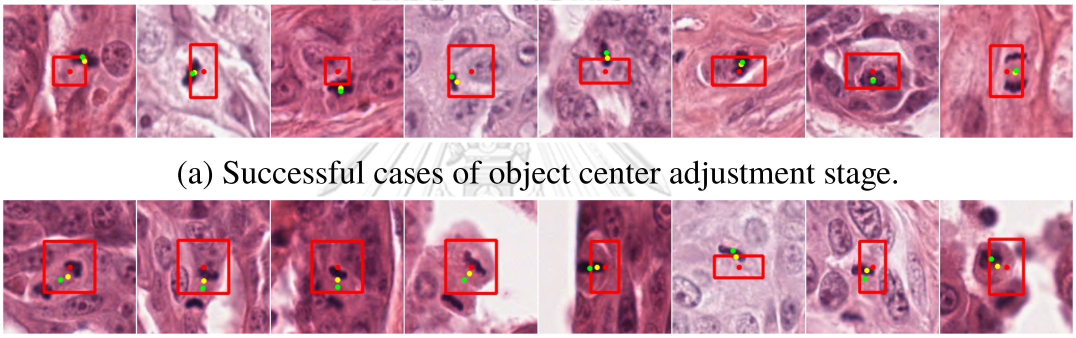


Figure 7.2: Confidence of objects produced by the detector being fed to the classifier when perturbed by image translation. The images on the left were sampled from the detection results, and the numbers in the center grid are positive object confidence produced by the classifier after shifting the image. On the other hand, the right grid showed object confidence when the object center adjustment stage was applied to the produced box before being fed to the classification stage. The confidence of the classification model without object center adjustment varies drastically compared to its counterpart. For example, when the top-left image was left-shifted up by three pixels, the classifier confidence without the object adjustment stage dropped from 51.2 to 27.8, while its counterpart only reduced from 81.0 to 75.2. when Red boxes indicate the position of the most frequently proposed adjusted object center.



(a) Successful cases of object center adjustment stage.

(b) Failure cases of object center adjustment stage.

Figure 7.3: Example prediction results produced by the object center adjustment stage on the CMC dataset. (a) shows successful examples. (b) shows failure examples. The first four images of are failures at the telophase stage. Red, yellow, and green dots indicate original, relocated, and ground truth object center, respectively. The red boxes are the bounding box produced by the detection stage.

Chapter VIII

CONCLUSION & FUTURE WORK

8.1 Conclusion

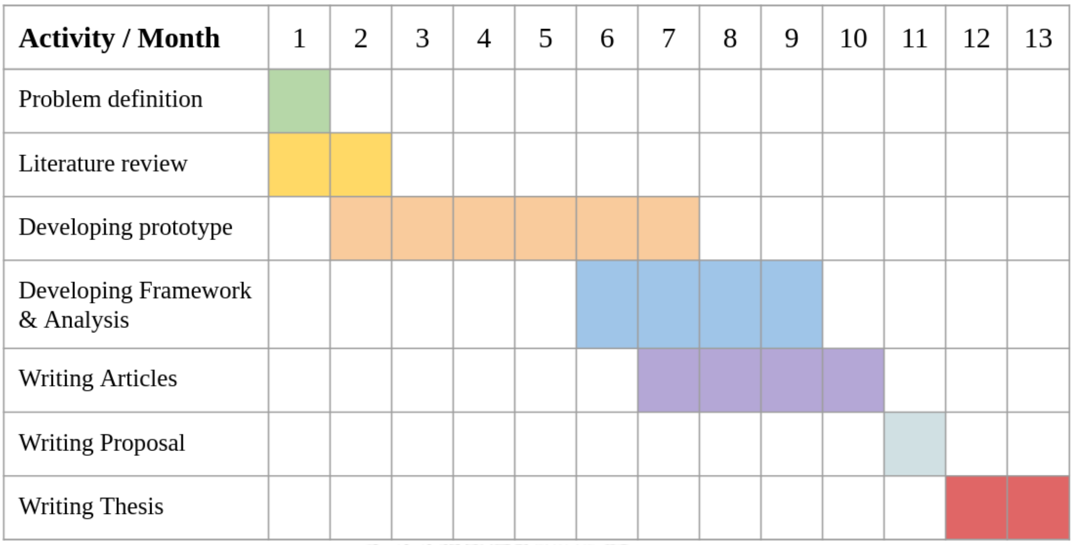
We propose ReCasNet, an enhanced deep learning pipeline that improves the two-stage mitosis detection pipeline in three ways. First, we presented window relocation, a method for reducing the number of false positives introduced by the sliding window algorithm by removing predictions around the window border and assigning them to a new window for re-inference. Second, we proposed the object center adjustment stage, which is a deep learning model in charge of adjusting the predicted center of the mitotic cell from the detection stage. This improves the consistency of the classification stage's inputs to be positioned at the image center. Third, we used an active learning technique to address inconsistencies in training data distribution by identifying additional informative examples based on the disagreement between the two stages in order to train the classification stage. On the CCMCT and CMC datasets, our proposed method significantly improves the overall pipeline performance in terms of both detection of individual mitotic figures and end-to-end region-of-interest proposal and mitotic count predictions.

8.2 Future work

Though the experiments on the CMC and CMMCT dataset in our work have been thoroughly conducted to verify the effectiveness of our method, the study is still limited to a single pathology scanner, and the model transferability from canine tissue to humans is still unexamined. We plan to further scale our method on different scanners with human tissue to observe the generalization of our work. We also aim to create a dataset with gold-standard information using PHH3 immunostaining to resolve the everlasting issue of confusion between mitosis cells and the other lookalike object.

8.3 Thesis timeline

Figure 8.1 shows the timeline of our thesis starting from October 2020.



Activity / Month	1	2	3	4	5	6	7	8	9	10	11	12	13
Problem definition	🟩												
Literature review	🟨	🟨											
Developing prototype		🟧	🟧	🟧	🟧	🟧	🟧						
Developing Framework & Analysis						🟦	🟦	🟦	🟦				
Writing Articles							🟪	🟪	🟪	🟪			
Writing Proposal											▨		
Writing Thesis												🟥	🟥

Figure 8.1: Thesis timeline.

REFERENCES

Alom, M. Z., Aspiras, T., Taha, T. M., Bowen, T., and Asari, V. K. 2020. Mitosisnet: End-to-end mitotic cell detection by multi-task learning. IEEE Access 8 (2020): 68695–68710.

Aubreville, M., Bertram, C., Klopfleisch, R., and Maier, A. 2018. Sliderunner. In Maier, A., Deserno, T. M., Handels, H., Maier-Hein, K. H., Palm, C., and Tolxdorff, T. (ed.), Bildverarbeitung für die Medizin 2018, pp. 309–314. Berlin, Heidelberg: Springer Berlin Heidelberg.

Aubreville, M., Bertram, C., Marzahl, C., Maier, A., and Klopfleisch, R. 2019. A large-scale dataset for mitotic figure assessment on whole slide images of canine cutaneous mast cell tumor. Scientific Data 6 (11 2019): 1–9.

Aubreville, M., Bertram, C., Donovan, T., Marzahl, C., Maier, A., and Klopfleisch, R. 2020a. A completely annotated whole slide image dataset of canine breast cancer to aid human breast cancer research. Scientific Data 7 (11 2020):

Aubreville, M., Bertram, C., Marzahl, C., Gurtner, C., Dettwiler, M., Schmidt, A., Bartenschlager, F., Merz, S., Fragoso, M., Kershaw, O., Klopfleisch, R., and Maier, A. 2020b. Deep learning algorithms out-perform veterinary pathologists in detecting the mitotically most active tumor region. Scientific Reports 10 (10 2020):

Avallone, G., Rasotto, R., Chambers, J. K., Miller, A. D., Behling-Kelly, E., Monti, P., Berlato, D., Valenti, P., and Roccabianca, P. 2021. Review of histological grading systems in veterinary medicine. Veterinary Pathology 58.5 (2021): 809–828.

Ba, J. L., Kiros, J. R., and Hinton, G. E. 2016. Layer normalization [Online]. Available from: https://arxiv.org/abs/1607.06450 [2016,].

Bertram, C., Aubreville, M., Gurtner, C., Bartel, A., Corner, S., Dettwiler, M., Kershaw, O., Noland, E., Schmidt, A., Sledge, D., Smedley, R., Thaiwong, T., Kiupel, M., Maier, A., and Klopfleisch, R. 2019. Computerized calculation of mitotic count distribution in canine cutaneous mast cell tumor sections: Mitotic count is area dependent. Veterinary Pathology 57 (2019): 214 – 226.

Bertram, C. A., Aubreville, M., Donovan, T. A., Bartel, A., Wilm, F., Marzahl, C., Assenmacher, C.-A., Becker, K., Bennett, M., Corner, S., Cossic, B., Denk, D., Dettwiler, M., Gonzalez, B. G., Gurtner, C., Haverkamp, A.-K., Heier, A., Lehmbecker, A., Merz, S., Noland, E. L., Plog, S., Schmidt, A., Sebastian, F., Sledge, D. G., Smedley, R. C., Tecilla, M., Thaiwong, T., Fuchs-Baumgartinger, A., Meuten, D. J., Breininger, K., Kiupel, M., Maier, A., and Klopfleisch, R. 2021. Computer-assisted mitotic count using a deep learning-based algorithm improves interobserver reproducibility and accuracy. Vet Pathol 59.2 (December 2021): 211–226.

Bloom, H. J. and Richardson, W. W. 1957. Histological grading and prognosis in breast cancer; a study of 1409 cases of which 359 have been followed for 15 years. Br. J. Cancer 11.3 (September 1957): 359–377.

Cai, Z. and Vasconcelos, N. 2018. Cascade R-CNN: delving into high quality object detection. In 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018, pp. 6154–6162. : Computer Vision Foundation / IEEE Computer Society.

Chen, H., Dou, Q., Wang, X., Qin, J., and Heng, P. 2016. Mitosis detection in breast cancer histology images via deep cascaded networks. Proceedings of the AAAI Conference on Artificial Intelligence 30.1 (Feb. 2016):

Chen, K., Wang, J., Pang, J., Cao, Y., Xiong, Y., Li, X., Sun, S., Feng, W., Liu, Z., Xu, J., Zhang, Z., Cheng, D., Zhu, C., Cheng, T., Zhao, Q., Li, B., Lu, X., Zhu, R., Wu, Y., Dai, J., Wang, J., Shi, J., Ouyang, W., Loy, C. C., and Lin, D. 2019. Mmdetection: Open mmlab detection toolbox and benchmark.

Chen, Q., Wang, Y., Yang, T., Zhang, X., Cheng, J., and Sun, J. 2021. You only look one-level feature. In IEEE Conference on Computer Vision and Pattern Recognition. :

Cireşan, D. C., Giusti, A., Gambardella, L. M., and Schmidhuber, J. 2013. Mitosis detection in breast cancer histology images with deep neural networks. In Mori, K., Sakuma, I., Sato, Y., Barillot, C., and Navab, N. (ed.), Medical Image Computing and Computer-Assisted Intervention – MICCAI 2013, pp. 411–418. Berlin, Heidelberg: Springer Berlin Heidelberg.

Deng, J., Dong, W., Socher, R., Li, L., Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 248–255. :

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1

(Long and Short Papers), pp. 4171–4186. Minneapolis, Minnesota: Association for Computational Linguistics.

Donovan, T., Moore, F., Bertram, C., Luong, R., Bolfă, P., Klopfleisch, R., Tvedten, H., Salas, E., Whitley, D., Aubreville, M., and Meuten, D. 2020. Mitotic figures-normal, atypical, and imposters: A guide to identification. Veterinary Pathology 58 (12 2020):

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. 2021. An image is worth 16x16 words: Transformers for image recognition at scale. In 9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021. : OpenReview.net.

Engstrom, L., Tsipras, D., Schmidt, L., and Madry, A. 2017. A rotation and a translation suffice: Fooling CNNs with simple transformations. ArXiv abs/1712.02779 (2017):

Everingham, M., Gool, L., Williams, C. K., Winn, J., and Zisserman, A. 2010. The pascal visual object classes (voc) challenge. Int. J. Comput. Vision 88.2 (jun 2010): 303–338.

Fitzke, M., Whitley, D., Yau, W., Rodrigues, F., Fadeev, V., Bacmeister, C., Carter, C., Edwards, J., Lungren, M., and Parkinson, M. 2021. Oncopetnet: A deep learning based ai system for mitotic figure counting on h&e stained whole slide digital images in a large veterinary diagnostic lab setting. ArXiv abs/2108.07856 (2021):

Ghiasi, G., Lin, T.-Y., and Le, Q. V. 2019. Nas-fpn: Learning scalable feature pyramid architecture for object detection. In 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. :

Girshick, R., Donahue, J., Darrell, T., and Malik, J. 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. In 2014 IEEE Conference on Computer Vision and Pattern Recognition, pp. 580–587. :

Goodfellow, I., Shlens, J., and Szegedy, C. 2015. Explaining and harnessing adversarial examples. In International Conference on Learning Representations. :

Hawkes, N. 2019. Cancer survival data emphasise importance of early diagnosis. BMJ 364 (2019):

He, K., Zhang, X., Ren, S., and Sun, J. 2016. Deep residual learning for image recognition. In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778. :

Hendrycks, D. and Gimpel, K. 2016. Gaussian error linear units (gelus). (2016):

Hu, J., Shen, L., and Sun, G. 2018. Squeeze-and-excitation networks. In 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7132–7141. :

Huang, C.-H. and Lee, H.-K. 2012. Automated mitosis detection based on exclusive independent component analysis. In Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012), pp. 1856–1859. :

Huang, Z., Xu, W., and Yu, K. 2015. Bidirectional LSTM-CRF models for sequence tagging. ArXiv abs/1508.01991 (2015):

Ioffe, S. and Szegedy, C. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Bach, F. and Blei, D. (ed.), Proceedings of the 32nd International Conference on Machine Learning, volume 37 of Proceedings of Machine Learning Research, pp. 448–456. Lille, France: PMLR.

Khan, A. M., El-Daly, H., and Rajpoot, N. M. 2012. A gamma-gaussian mixture model for detection of mitotic cells in breast cancer histopathology images. In Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012), pp. 149–152. :

Krizhevsky, A., Sutskever, I., and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. In Pereira, F., Burges, C., Bottou, L., and Weinberger, K. (ed.), Advances in Neural Information Processing Systems, volume 25. : Curran Associates, Inc.

Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P. 1998. Gradient-based learning applied to document recognition. Proceedings of the IEEE 86.11 (1998): 2278–2324.

Li, C., Wang, X., Liu, W., and Latecki, L. J. 2018. Deepmitosis: Mitosis detection via deep detection, verification and segmentation networks. Medical Image Analysis 45 (2018): 121–133.

Li, Y., Mao, H., Girshick, R., and He, K. 2022. Exploring plain vision transformer backbones for object detection [Online]. Available from: https://arxiv.org/abs/2203.16527 [2022,June].

Lin, T., Maire, M., Belongie, S. J., Bourdev, L. D., Girshick, R. B., Hays, J., Perona, P., Ramanan, D., Doll'a r, P., and Zitnick, C. L. 2014. Microsoft COCO: common objects in context. CoRR abs/1405.0312 (2014):

Lin, T.-Y., Dollár, P., Girshick, R. B., He, K., Hariharan, B., and Belongie, S. J. 2017a. Feature pyramid networks for object detection. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017): 936–944.

Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollár, P. 2017b. Focal loss for dense object detection. In 2017 IEEE International Conference on Computer Vision (ICCV), pp. 2999–3007. :

Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., van der Laak, J. A., van Ginneken, B., and Sánchez, C. I. 2017. A survey on deep learning in medical image analysis. Medical Image Analysis 42 (2017): 60–88.

Liu, S., Qi, L., Qin, H., Shi, J., and Jia, J. 2018. Path aggregation network for instance segmentation. In 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8759–8768. :

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., and Berg, A. C. 2016. Ssd: Single shot multibox detector. In Leibe, B., Matas, J., Sebe, N., and Welling, M. (ed.), Computer Vision – ECCV 2016, pp. 21–37. Cham: Springer International Publishing.

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., and Guo, B. 2021. Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 10012–10022. :

Liu, Z., Mao, H., Wu, C.-Y., Feichtenhofer, C., Darrell, T., and Xie, S. 2022. A convnet for the 2020s. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11976–11986. :

Malon, C. and Cosatto, E. 2013. Classification of mitotic figures with convolutional neural networks and seeded blob features. Journal of pathology informatics 4 (05 2013): 9.

Meuten, D., Moore, F., and George, J. 2016. Mitotic count and the field of view area: Time to standardize. Veterinary Pathology 53 (01 2016): 7–9.

Nateghi, R., Danyali, H., and Helfroush, M. 2017. Maximized inter-class weighted mean for fast and accurate mitosis cells detection in breast cancer histopathology images. Journal of Medical Systems 41 (08 2017):

Pan, X., Lu, Y., Lan, R., Liu, Z., Qin, Z., Wang, H., and Liu, Z. 2021. Mitosis detection techniques in H&E stained breast cancer pathological images: A comprehensive review. Computers & Electrical Engineering 91 (2021): 107038.

Paul, A. and Mukherjee, D. P. 2015. Mitosis detection for invasive breast cancer grading in histopathological images. IEEE Transactions on Image Processing 24.11 (2015): 4041–4054.

Paul, A., Dey, A., Mukherjee, D. P., Sivaswamy, J., and Tourani, V. 2015. Regenerative random forest with automatic feature selection to detect mitosis in histopathological breast cancer images. In Navab, N., Hornegger, J., Wells, W. M., and Frangi, A. (ed.), Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015, pp. 94–102. Cham: Springer International Publishing.

Racoceanu, D., Calvo, J., Attieh, E., Naour, G. L., and Gloaguen, A. 2014. Detection of mitosis and evaluation of nuclear atypia score in breast cancer histological images. :

Redmon, J. and Farhadi, A. 2017. Yolo9000: Better, faster, stronger. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6517–6525. :

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. 2016. You only look once: Unified, real-time object detection. In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 779–788. :

Ren, S., He, K., Girshick, R., and Sun, J. 2015a. Faster R-CNN: Towards real-time object detection with region proposal networks. In Cortes, C., Lawrence, N., Lee, D., Sugiyama, M., and Garnett, R. (ed.), Advances in Neural Information Processing Systems, volume 28. : Curran Associates, Inc.

Ren, S., He, K., Girshick, R., and Sun, J. 2015b. Faster r-cnn: Towards real-time object detection with region proposal networks. In Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1, NIPS'15, p. 91–99. Cambridge, MA, USA: MIT Press.

Roux, L., Racoceanu, D., Lomenie, N., Kulikova, M., Irshad, H., Klossa, J., Capron, F., Genestie, C., Le Naour, G., and Gurcan, M. 2013. Mitosis detection in breast cancer histological images an icpr 2012 contest. Journal of pathology informatics 4 (05 2013): 8.

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L.-C. 2018. Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). :

Sener, O. and Savarese, S. 2018. Active learning for convolutional neural networks: A core-set approach. In 6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings. : OpenReview.net.

Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., and Lecun, Y. 2013. Overfeat: Integrated recognition, localization and detection using convolutional networks. International Conference on Learning Representations (ICLR) (Banff) (12 2013):

Settles, B. 2009. Active learning literature survey. :

Simonyan, K. and Zisserman, A. 2015. Very deep convolutional networks for large-scale image recognition. CoRR abs/1409.1556 (2015):

Sommer, C., Fiaschi, L., Hamprecht, F. A., and Gerlich, D. W. 2012. Learning-based mitotic cell detection in histopathological images. In Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012), pp. 2306–2309. :

Srinidhi, C. L., Ciga, O., and Martel, A. L. 2021. Deep neural network models for computational histopathology: A survey. Medical Image Analysis 67 (Jan 2021): 101813.

Tan, M. and Le, Q. 2019. EfficientNet: Rethinking model scaling for convolutional neural networks. In Chaudhuri, K. and Salakhutdinov, R. (ed.), Proceedings of the 36th International Conference on Machine Learning, volume 97 of Proceedings of Machine Learning Research, pp. 6105–6114. : PMLR.

Tan, M., Pang, R., and Le, Q. V. 2020. Efficientdet: Scalable and efficient object detection. In 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 10778–10787. :

Tek, F. 2013. Mitosis detection using generic features and an ensemble of cascade adaboosts. Journal of pathology informatics 4 (05 2013): 12.

van Steenhoven, J. E. C., Kuijer, A., Kornegoor, R., van Leeuwen, G., van Gorp, J., van Dalen, T., and van Diest, P. J. 2020. Assessment of tumour proliferation by use of the mitotic activity index, and ki67 and phosphohistone H3 expression, in early-stage luminal breast cancer. Histopathology 77.4 (October 2020): 579–587.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. 2017. Attention is all you need. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (ed.), Advances in Neural Information Processing Systems, volume 30. : Curran Associates, Inc.

Veta, M., van Diest, P. J., and Pluim, J. P. W. 2013. Detecting mitotic figures in breast cancer histopathology images. In Gurcan, M. N. and Madabhushi, A. (ed.), Medical Imaging 2013: Digital Pathology, volume 8676, pp. 70 – 76. : SPIE.

Veta, M., Diest, P., Willems, S., Wang, H., Madabhushi, A., Cruz-Roa, A., González, F., Larsen, A., Vestergaard, J., Dahl, A., Cireşan, D., Schmidhuber, J., Giusti, A., Gambardella, L. M., Tek, F., Walter, T., Wang, C.-W., Kondo, S., Matuszewski, B., and Pluim, J. 2014. Assessment of algorithms for mitosis detection in breast cancer histopathology images. Medical Image Analysis (11 2014):

Veta, M., Diest, P., Jiwa, M., Al-Janabi, S., and Pluim, J. 2016. Mitosis Counting in Breast Cancer: Object-Level Interobserver Agreement and Comparison to an Automatic Method. PLOS ONE 11 (08 2016): e0161286.

Veta, M., Heng, Y. J., Stathonikos, N., Bejnordi, B. E., Beca, F., Wollmann, T., Rohr, K., Shah, M. A., Wang, D., Rousson, M., and et al. 2019. Predicting breast tumor proliferation from whole-slide images: The tupac16 challenge. Medical Image Analysis 54 (May 2019): 111–121.

Xie, S., Girshick, R., Dollár, P., Tu, Z., and He, K. 2017. Aggregated residual transformations for deep neural networks. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5987–5995. :

Chapter IX

LIST OF PUBLICATIONS

9.1 Journal

1. Piansaddhayanon, C., Santisukwongchote, S., Shuangshoti, S., Tao, Q., Sriswasdi, S., & Chuangsuwanich, E. (2022). ReCasNet: Improving consistency within the two-stage mitosis detection framework. Artificial Intelligence in Medicine, 102462. doi:10.1016/j.artmed.2022.102462

Biography

Chawan Piansaddhayanon was born in Bangkok on August 16, 1998. He graduated from high school at Assumption College in 2015. From 2016 to 2020, he studied for a bachelor's degree in computer engineering at the faculty of engineering, Chulalongkorn University. In 2021, he pursued a master's degree in computer engineering at the same university.