

**School of Computer Science**

FACULTY OF ENGINEERING AND  
PHYSICAL SCIENCES



**UNIVERSITY OF LEEDS**

---

# **Deep Learning for Improving Cell Segmentation and Classification in Histology Images**

**Hongbin Guo**

**Submitted in accordance with the requirements for the degree of  
MSc Advanced Computer Science (Artificial Intelligence)**

2025–2026

The candidate confirms that the following have been submitted.

| Items               | Format | Recipient(s) and Date              |
|---------------------|--------|------------------------------------|
| Deliverable 1, 2, 3 | Report | SSO (DD/MM/YY)                     |
| Deliverable 4       | URL    | Supervisor, Assessor<br>(DD/MM/YY) |

Type of project: Empirical Investigation \_\_\_\_\_

The candidate confirms that the work submitted is their own and the appropriate credit has been given where reference has been made to the work of others.

I understand that failure to attribute material which is obtained from another source may be considered as plagiarism.

(Signature of Student) Hongbin Guo \_\_\_\_\_

## Summary

This dissertation investigates the effects of backbone and decoder choice in CellViT models for nuclei instance segmentation and classification in histopathology images. Four configurations are evaluated by combining two backbones, ViT-256 and SAM-H, with two decoder architectures, HoVer-Net and CPP-Net. The study first reproduces CellViT experiments on PanNuke and then evaluates the corresponding trained models on MoNuSeg without additional training or fine-tuning.

The PanNuke reproduction closely matches the published CellViT results where direct comparison is available, supporting the reliability of the experimental pipeline. Across the reproduced experiments, SAM-H achieves higher aggregate performance than ViT-256 under both decoders, while HoVer-Net achieves higher aggregate bPQ and mPQ than CPP-Net.

On MoNuSeg, SAM-H again performs better than ViT-256 across the principal instance-segmentation metrics under both decoder architectures. Decoder performance is less consistent: CPP-Net performs better on some metrics, whereas HoVer-Net performs better on others. Paired statistical comparisons similarly provide stronger evidence for backbone differences than for a consistent overall advantage of either decoder.

Qualitative, oracle, and routing analyses show that decoder performance can vary between images and local regions, but simple routing strategies do not consistently outperform the best fixed decoder.

Overall, the results indicate that backbone choice has a more consistent effect on CellViT segmentation performance, whereas decoder performance is more dependent on dataset characteristics and evaluation metrics.

## **Acknowledgements**

I would like to express my sincere gratitude to my supervisor, Dr Arash Rabbani, for his guidance, feedback, and support throughout this dissertation project. His advice helped me refine the research direction, develop the experimental methodology, and critically interpret the results of the study.

I would also like to thank my assessor, Dr Duygu Sarikaya, for her valuable feedback and guidance on my research, which helped me further develop and refine the project. Finally, I would like to thank my family and friends for their encouragement and personal support throughout my MSc studies and dissertation project.

# Contents

|          |                                                             |           |
|----------|-------------------------------------------------------------|-----------|
| <b>1</b> | <b>Introduction</b>                                         | <b>3</b>  |
| 1.1      | Project Aim . . . . .                                       | 3         |
| 1.2      | Objectives . . . . .                                        | 4         |
| 1.3      | Deliverables . . . . .                                      | 4         |
| 1.4      | Ethical, Legal, and Social Issues . . . . .                 | 5         |
| <b>2</b> | <b>Background Research</b>                                  | <b>7</b>  |
| 2.1      | Literature Survey . . . . .                                 | 7         |
| 2.1.1    | Histology Nuclei Segmentation . . . . .                     | 7         |
| 2.1.2    | CNN-Based Nuclei Segmentation Methods . . . . .             | 7         |
| 2.1.3    | Transformer-Based Nuclei Segmentation Methods . . . . .     | 8         |
| 2.1.4    | Foundation Models for Medical Image Analysis . . . . .      | 9         |
| 2.1.5    | Research Gap . . . . .                                      | 9         |
| 2.2      | Methods and Techniques . . . . .                            | 11        |
| 2.2.1    | CellViT as a Modular Segmentation Framework . . . . .       | 11        |
| 2.2.2    | Benchmark Reproduction and External Evaluation . . . . .    | 11        |
| 2.2.3    | Evaluation Metrics . . . . .                                | 12        |
| 2.2.4    | Statistical Analysis . . . . .                              | 12        |
| 2.2.5    | Qualitative Analysis and Oracle Diagnostics . . . . .       | 13        |
| 2.3      | Choice of Methods . . . . .                                 | 13        |
| <b>3</b> | <b>Datasets and Experimental Design</b>                     | <b>15</b> |
| 3.1      | Datasets . . . . .                                          | 15        |
| 3.1.1    | PanNuke . . . . .                                           | 15        |
| 3.1.2    | MoNuSeg . . . . .                                           | 15        |
| 3.2      | Experimental Design . . . . .                               | 16        |
| 3.2.1    | Research Questions and Experimental Overview . . . . .      | 16        |
| 3.2.2    | Implementation and Model Configuration . . . . .            | 17        |
| 3.2.3    | PanNuke Reproduction Protocol . . . . .                     | 18        |
| 3.2.4    | MoNuSeg External Evaluation Protocol . . . . .              | 19        |
| 3.2.5    | Evaluation Protocol . . . . .                               | 20        |
| 3.2.6    | Statistical, Qualitative, and Diagnostic Analysis . . . . . | 21        |
| <b>4</b> | <b>Results of the Empirical Investigation</b>               | <b>23</b> |
| 4.1      | PanNuke Reproduction and Configuration Results . . . . .    | 23        |
| 4.1.1    | Reproduction against Published CellViT Results . . . . .    | 23        |

|          |                                                         |           |
|----------|---------------------------------------------------------|-----------|
| 4.1.2    | Backbone Results . . . . .                              | 25        |
| 4.1.3    | Decoder Results . . . . .                               | 25        |
| 4.2      | MoNuSeg External Evaluation Results . . . . .           | 26        |
| 4.2.1    | Overall Performance . . . . .                           | 27        |
| 4.2.2    | Backbone Results . . . . .                              | 28        |
| 4.2.3    | Decoder Results . . . . .                               | 28        |
| <b>5</b> | <b>Validation of Results</b>                            | <b>31</b> |
| 5.1      | Statistical Validation . . . . .                        | 31        |
| 5.1.1    | PanNuke Statistical Analysis . . . . .                  | 31        |
| 5.1.2    | MoNuSeg Statistical Analysis . . . . .                  | 32        |
| 5.2      | Qualitative Comparison of Decoder Predictions . . . . . | 33        |
| 5.3      | Decoder Routing Analysis . . . . .                      | 36        |
| 5.3.1    | Oracle Analysis . . . . .                               | 36        |
| 5.3.2    | Routing Using Global Image Features . . . . .           | 37        |
| 5.3.3    | Local Decoder Comparison . . . . .                      | 38        |
| 5.4      | Summary of Validation Findings . . . . .                | 40        |
| <b>6</b> | <b>Conclusions and Future Work</b>                      | <b>41</b> |
| 6.1      | Conclusions . . . . .                                   | 41        |
| 6.2      | Future Work . . . . .                                   | 42        |
|          | <b>References</b>                                       | <b>43</b> |
|          | <b>Appendices</b>                                       | <b>45</b> |
| <b>A</b> | <b>External Materials</b>                               | <b>47</b> |
| A.1      | CPP-Net Compatibility for MoNuSeg Inference . . . . .   | 47        |

# List of Figures

|     |                                                              |    |
|-----|--------------------------------------------------------------|----|
| 4.1 | Backbone comparison of PanNuke mPQ. . . . .                  | 25 |
| 4.2 | Decoder comparison of PanNuke mPQ. . . . .                   | 26 |
| 4.3 | PanNuke nuclei detection precision and recall. . . . .       | 27 |
| 4.4 | Backbone comparison of MoNuSeg bPQ. . . . .                  | 28 |
| 4.5 | Decoder comparison of MoNuSeg bPQ. . . . .                   | 29 |
| 5.1 | Full-context MoNuSeg decoder comparison under SAM-H. . . . . | 34 |
| 5.2 | Zoomed MoNuSeg decoder comparison under SAM-H. . . . .       | 35 |



# List of Tables

|     |                                                                                                  |    |
|-----|--------------------------------------------------------------------------------------------------|----|
| 3.1 | CellViT configurations evaluated in this dissertation. . . . .                                   | 17 |
| 3.2 | Shared training settings used in the PanNuke reproduction experiments. .                         | 18 |
| 3.3 | PanNuke experimental assignments reproduced from the official CellViT<br>configuration. . . . .  | 19 |
| 4.1 | PanNuke nuclei detection reproduction results. . . . .                                           | 23 |
| 4.2 | PanNuke instance segmentation reproduction results. . . . .                                      | 24 |
| 4.3 | PanNuke class-wise PQ reproduction results. . . . .                                              | 24 |
| 4.4 | MoNuSeg performance across the four configurations. . . . .                                      | 27 |
| 5.1 | Descriptive paired $t$ -test results on PanNuke. . . . .                                         | 31 |
| 5.2 | Paired Wilcoxon test results on MoNuSeg. . . . .                                                 | 32 |
| 5.3 | SAM-H image-level oracle and routing results. . . . .                                            | 36 |
| 5.4 | ViT-256 image-level oracle and routing results. . . . .                                          | 37 |
| 5.5 | Correlations between density features and decoder performance differences<br>on MoNuSeg. . . . . | 38 |
| 5.6 | Local MoNuSeg oracle results. . . . .                                                            | 39 |
| 5.7 | Local MoNuSeg routing results. . . . .                                                           | 39 |



# Chapter 1

## Introduction

### 1.1 Project Aim

The aim of this project is to reproduce and empirically analyse models based on CellViT for nuclei instance segmentation in histology images, with particular emphasis on how backbone and decoder choice affect performance and whether the resulting patterns remain consistent across datasets.

Segmentation of nuclei instances is an important task in computational pathology because nuclear morphology, spatial distribution, and type provide information that can support quantitative tissue analysis. Reliable instance segmentation supports downstream applications such as analysis of cell types, tissue characterisation, and tumour microenvironment analysis [6, 8]. The task is challenging because nuclei may be crowded, touching or overlapping, morphologically irregular, or affected by staining and tissue variation.

CellViT combines Transformer image encoders with specialised nuclei segmentation decoders within a common segmentation framework [8]. This modular structure provides an opportunity to study the effects of backbone and decoder choice separately. Rather than asking only which complete model achieves the highest score, this project investigates whether performance differences are associated with the backbone, the decoder, or their interaction with the dataset and evaluation metric.

This project evaluates four configurations formed by combining two CellViT backbones, ViT-256 and SAM-H, with the HoVer-Net and CPP-Net decoder designs evaluated in the original CellViT study [8]. The empirical investigation is organised into two stages. First, the four configurations are evaluated on PanNuke [5] to reproduce the CellViT benchmark setting and establish a reliable reference point. Second, the same four configurations are evaluated on the MoNuSeg test set [11] without additional training or fine-tuning, allowing the backbone and decoder trends observed on PanNuke to be examined on a different external histopathology dataset.

The project does not aim to introduce a new segmentation architecture. Instead, it provides a controlled empirical investigation of reproduction fidelity, backbone effects, decoder effects, precision and recall behaviour, the Detection Quality (DQ) and Segmentation Quality (SQ) components of Panoptic Quality, evaluation across datasets, and exploratory decoder routing diagnostics.

## 1.2 Objectives

1. To review nuclei instance segmentation methods, medical image segmentation approaches using Transformers, and applications of foundation models relevant to CellViT.
2. To reproduce four CellViT configurations on PanNuke using ViT-256 and SAM-H backbones with HoVer-Net and CPP-Net decoders.
3. To compare the reproduced PanNuke results with the corresponding values reported in the original CellViT study and to analyse the effects of backbone and decoder choice using aggregate metrics, results for individual classes, precision and recall behaviour, and the decomposition of Panoptic Quality into DQ and SQ.
4. To evaluate the same four configurations on the MoNuSeg test set of 14 images without additional training or fine-tuning, and to assess whether the backbone and decoder trends observed on PanNuke remain visible on a different external histopathology dataset using quantitative, statistical, and qualitative analysis.
5. To investigate decoder complementarity using exploratory image-level and patch-level oracle and routing diagnostics.

## 1.3 Deliverables

1. A reproducible evaluation pipeline for four CellViT configurations on PanNuke and MoNuSeg.
2. A complete empirical evaluation of the four configurations, including PanNuke reproduction results and external inference evaluation on MoNuSeg.
3. A comparative analysis of backbone and decoder behaviour using quantitative metrics, statistical testing, qualitative visualisation, and exploratory routing diagnostics.
4. Reproducibility documentation describing the implementation adaptations required to evaluate CPP-Net configurations through the MoNuSeg inference pipeline.
5. The source code, experiment configurations, evaluation scripts, and generated results required to support the empirical investigation.

## 1.4 Ethical, Legal, and Social Issues

This project uses publicly released histopathology datasets and does not collect new patient data or involve direct interaction with human participants. PanNuke and MoNuSeg are established research datasets for nuclei segmentation and related evaluation tasks [5, 11]. The data are used only for academic research, model inference, and evaluation, and the results are not interpreted as evidence of clinical readiness. Public datasets, source code, and pretrained model checkpoints are used in accordance with their stated licence terms, research conditions, and citation requirements. From a broader social perspective, accurate nuclei segmentation can support quantitative analysis of nuclear morphology and tissue characterisation, whereas inaccurate segmentation may lead to misleading downstream measurements. The dissertation therefore emphasises responsible data use, reproducible evaluation, transparent reporting, and cautious interpretation of model performance.



# Chapter 2

## Background Research

### 2.1 Literature Survey

#### 2.1.1 Histology Nuclei Segmentation

Nuclei instance segmentation is an important task in computational pathology because nuclear morphology, spatial organisation, and cell type provide information that supports quantitative analysis of histological tissue. Accurate instance segmentation supports downstream analyses of nuclear morphology, nuclei classification, and tissue-level characterisation [6, 8].

The task is challenging because individual nuclei must be identified and separated even when they are touching, overlapping, densely distributed, or have blurry boundaries. Nuclear appearance can also vary substantially across cell types, disease types, and tissue regions, further complicating robust instance segmentation [6, 3].

A major development in biomedical image segmentation was U-Net [14]. Its contracting path captures contextual features, while its expansive path combines upsampled representations with high-resolution features from the contracting path to support precise localisation. The original U-Net also introduced a weighted loss that assigns greater importance to narrow borders between touching cells, encouraging the network to learn boundaries between neighbouring objects.

Nevertheless, the core U-Net formulation performs pixel-wise segmentation rather than directly encoding the identity or geometry of individual instances. Later nuclei segmentation methods therefore introduced more explicit instance representations that describe how pixels belonging to adjacent nuclei should be separated.

#### 2.1.2 CNN-Based Nuclei Segmentation Methods

One approach to explicit instance prediction is detection-based segmentation. Mask R-CNN extends object detection with a parallel branch that predicts a segmentation mask for each detected object instance [7]. This provides an explicit instance-level alternative to pixel-wise semantic segmentation.

Nuclei-specific methods have also introduced geometric representations designed to separate neighbouring instances. HoVer-Net predicts a nuclear pixel map together with horizontal and vertical distance maps and a nuclear classification output [6]. The horizontal and vertical maps encode the position of nuclear pixels relative to the centre of mass of the corresponding nucleus. Gradients derived from these maps are

subsequently used during post-processing to separate clustered or touching nuclei. A different geometric representation is used by StarDist [15]. For pixels belonging to an object, StarDist predicts radial distances to the object boundary along predefined directions, together with an object-probability map. These predictions define star-convex polygon proposals, from which high-confidence instances are selected using non-maximum suppression. This representation provides a direct mechanism for representing individual objects in crowded cellular images.

CPP-Net builds on this polygon-based formulation [3]. It adopts StarDist-style initial radial distance predictions and refines them using contextual information from additional sampled points. A context enhancement mechanism and confidence-based weighting are used to combine these predictions, while a Shape-Aware Perceptual (SAP) loss provides additional supervision for nuclear shape. During post-processing, non-maximum suppression removes redundant polygon proposals, after which fine-grained post-processing further refines the resulting instance predictions.

HoVer-Net and CPP-Net are particularly relevant to this dissertation because they address the same nuclei instance segmentation problem using different geometric representations. HoVer-Net uses horizontal and vertical distance maps followed by gradient-based instance reconstruction, whereas CPP-Net uses radial distance prediction, contextual polygon refinement, and polygon-based instance reconstruction. Comparing the two within the same CellViT framework therefore allows decoder behaviour to be examined while holding the broader segmentation framework constant.

### 2.1.3 Transformer-Based Nuclei Segmentation Methods

Convolutional architectures construct image representations primarily through local operations, with larger receptive fields emerging as information is propagated through successive layers. Vision Transformer (ViT) introduced a different formulation in which an image is represented as a sequence of embedded patches processed by a Transformer encoder [4]. Global self-attention allows information to be integrated across image patches and introduces substantially less image-specific inductive bias than conventional convolutional architectures.

The original ViT study also demonstrated the importance of large-scale pretraining for effective transfer. This principle is particularly relevant to the ViT-256 backbone used by CellViT. ViT-256 originates from the first stage of the Hierarchical Image Pyramid Transformer (HIPT). This encoder is a ViT-S/16 operating on  $256 \times 256$  histology patches and was self-supervised using DINO on approximately 104 million histology patches from The Cancer Genome Atlas (TCGA) [2]. CellViT reuses this encoder pretrained on histopathology images as one of its backbone options [8].

Transformer representations have also been combined with specialised medical segmentation decoders. TransUNet provides one representative example, integrating

feature encoding using Transformers with a U-shaped medical segmentation architecture [1]. This illustrates the broader separation between feature extraction and downstream segmentation decoding.

This distinction is important for the present study. Different pretrained encoders may produce different feature representations, while the final instance segmentation also depends on how those representations are converted into individual nuclei. A controlled experimental design is therefore required to distinguish changes associated with backbone choice from those associated with decoder design.

### 2.1.4 Foundation Models for Medical Image Analysis

Large-scale visual pretraining has also produced models designed to transfer across segmentation tasks. The Segment Anything Model (SAM) is a promptable segmentation model consisting of a Vision Transformer image encoder, prompt encoder, and lightweight mask decoder [10]. SAM was trained using the large-scale SA-1B dataset, which contains approximately 11 million images and more than one billion segmentation masks.

SAM was trained for broad, general-purpose image segmentation rather than specifically for medical imaging. Subsequent work has investigated adapting SAM to medical segmentation. MedSAM, for example, adapts a pretrained SAM model using more than 1.5 million paired medical images and masks spanning multiple imaging modalities [13]. It therefore provides an example of large-scale adaptation of a pretrained segmentation foundation model to the medical domain.

The relevance of SAM to this dissertation is specifically its image encoder. CellViT incorporates pretrained SAM image encoders as alternative backbones while replacing SAM’s original prompt-based pipeline for mask generation with specialised decoders for nuclei segmentation [8]. The experiments in this dissertation therefore evaluate the SAM-H image encoder within the CellViT architecture rather than the complete promptable SAM system.

This creates a useful contrast between the two backbones evaluated in this study. ViT-256 is a relatively compact self-supervised encoder pretrained on histopathology images and originating from HIPT [2], whereas SAM-H is a substantially larger image encoder pretrained within the general-purpose SAM framework [10]. Whether these different pretraining and model design choices produce consistent performance differences in nuclei segmentation is an empirical question addressed by the experiments rather than assumed in advance.

### 2.1.5 Research Gap

The literature reviewed above highlights two architectural dimensions that are directly relevant to nuclei instance segmentation. First, specialised decoders use different

geometric representations to reconstruct individual nuclei, as illustrated by the horizontal and vertical distance representation of HoVer-Net and the radial polygon representation used by StarDist and CPP-Net [6, 15, 3]. Second, pretrained Transformer encoders provide alternative approaches to visual feature extraction, including the ViT-256 encoder pretrained on histopathology images and the substantially larger SAM-H image encoder [2, 10].

The original CellViT study brings these backbone and decoder designs together within a common nuclei segmentation framework and reports alternative HoVer-Net and CPP-Net decoder configurations on PanNuke, including configurations based on ViT-256 and SAM-H [8]. Therefore, the research gap is not simply the absence of decoder comparison within CellViT.

Instead, the evaluation across datasets remains less complete. The original CellViT MoNuSeg experiment evaluates CellViT models using HoVer-Net decoders but does not report the corresponding CPP-Net configurations [8]. Consequently, the consistency of decoder effects across PanNuke and MoNuSeg cannot be examined using the same complete set of backbone and decoder configurations from the published MoNuSeg results alone.

This dissertation addresses this gap by evaluating a controlled  $2 \times 2$  combination of two backbones, ViT-256 and SAM-H, and two decoders, HoVer-Net and CPP-Net. The four configurations are first evaluated under the PanNuke benchmark setting and are then applied to the MoNuSeg test set without additional training or fine-tuning. This design allows the consistency of both backbone-level and decoder-level performance patterns to be examined across the two datasets.

A further question concerns whether similar aggregate decoder performance implies similar predictions. Two decoders may achieve comparable overall scores while succeeding and failing on different nuclei or image regions. Qualitative disagreement analysis and oracle diagnostics can therefore provide additional evidence about whether alternative decoder predictions contain complementary information.

The contribution of this dissertation is consequently empirical rather than architectural. It does not introduce a new segmentation network. Instead, it uses reproduction, controlled comparison of backbone and decoder choices, evaluation across datasets, paired statistical analysis, qualitative disagreement visualisation, and exploratory oracle and routing diagnostics to investigate how backbone choice, decoder choice, dataset, and metric influence the observed CellViT performance patterns.

## 2.2 Methods and Techniques

### 2.2.1 CellViT as a Modular Segmentation Framework

CellViT provides a suitable framework for the experimental comparison because it combines Transformer encoders for feature extraction with specialised decoders for nuclei segmentation [8]. Alternative pretrained backbones can therefore be examined together with different instance segmentation decoder designs within a common model family.

Within this structure, the backbone extracts visual representations from the input image, while the decoder converts the extracted features into outputs used to reconstruct individual nuclei. This separation makes it possible to compare backbone and decoder choices without replacing the complete segmentation system for every experiment.

Such a framework is appropriate for the objective of this dissertation because the aim is not simply to identify the highest-scoring complete model. Instead, the experiments examine whether observed performance differences remain consistent when either the backbone or decoder changes while the broader experimental pipeline is retained.

### 2.2.2 Benchmark Reproduction and External Evaluation

Benchmark reproduction and external evaluation serve different purposes in this dissertation. Reproduction first establishes whether the selected CellViT configurations can be evaluated under the published PanNuke setting and whether comparable aggregate behaviour can be obtained. This provides a reference point before the same architectural comparisons are examined on a different dataset.

PanNuke [5] is a multi-tissue nuclei instance segmentation and classification dataset and serves as the primary benchmark used in the original CellViT study [8]. It therefore provides the primary setting for reproducing and analysing the four CellViT configurations.

MoNuSeg provides a separate multi-organ histopathology benchmark for nuclei segmentation [12, 11]. In this dissertation, only the MoNuSeg test set of 14 images is used for inference and evaluation. Models trained under the PanNuke setting are applied without MoNuSeg training, fine-tuning, or target-dataset checkpoint selection. MoNuSeg therefore serves as a different external histopathology setting in which the persistence of the PanNuke backbone and decoder patterns can be examined.

The two stages provide complementary evidence. PanNuke assesses reproduction and the effects of backbone and decoder choice under the original benchmark setting, whereas MoNuSeg evaluates whether the same relative patterns remain visible during inference on a different dataset.

### 2.2.3 Evaluation Metrics

Nuclei instance segmentation requires both the detection of individual nuclei and accurate delineation of their boundaries. The evaluation therefore uses metrics that distinguish these aspects of performance [6].

Panoptic Quality (PQ) was introduced by Kirillov et al. [9]. In its original formulation,

$$PQ = \underbrace{\frac{|TP|}{|TP| + \frac{1}{2}|FP| + \frac{1}{2}|FN|}}_{RQ} \times \underbrace{\frac{\sum_{(p,g) \in TP} IoU(p,g)}{|TP|}}_{SQ}, \quad (2.1)$$

where  $TP$ ,  $FP$ , and  $FN$  denote matched, false-positive, and false-negative instances, and  $IoU$  is the intersection over union between a matched prediction  $p$  and ground-truth instance  $g$ . In the original formulation, the first term is Recognition Quality (RQ), while the second term is Segmentation Quality (SQ). Matching uses an  $IoU > 0.5$  criterion, which yields unique one-to-one matches under the PQ formulation [9].

In the literature on nuclei instance segmentation, including HoVer-Net, the recognition term is commonly referred to as Detection Quality (DQ) [6]. This dissertation therefore uses the notation

$$PQ = DQ \times SQ, \quad (2.2)$$

where DQ reflects instance detection quality and SQ is the mean overlap quality of matched nuclei. Examining DQ and SQ separately helps identify whether a PQ difference arises primarily from instance detection or from the segmentation quality of matched nuclei.

PanNuke additionally distinguishes binary and multi-class variants of PQ [5]. Binary Panoptic Quality (bPQ) treats nuclei as a single class, whereas multi-class Panoptic Quality (mPQ) calculates class-specific PQ values across the annotated nuclei categories and averages them. This distinction makes bPQ suitable for class-agnostic comparison, including MoNuSeg, while mPQ is used when PanNuke nucleus type information is evaluated.

Precision, recall, and detection  $F_1$  are also reported to characterise false-positive and false-negative behaviour. Although DQ has the algebraic form of an  $F_1$  measure, the matching procedures used for PQ-based DQ and the reported detection metrics are distinct [6, 5]. The implementation-specific evaluation procedures used in this dissertation are therefore defined in Chapter 3.

### 2.2.4 Statistical Analysis

Aggregate averages alone do not indicate whether observed differences are consistent across paired experimental units. Paired statistical comparisons are therefore used as

supporting evidence where the same experimental settings or images are evaluated by multiple model configurations.

A paired  $t$ -test assesses the mean of paired differences [16]. In this dissertation, it is applied where the paired differences are approximately normally distributed.

The Wilcoxon signed-rank test provides a rank-based non-parametric approach for paired differences [17]. In this dissertation, it is used as an alternative where normality is not assumed, with the symmetry of the difference distribution considered when interpreting the test.

The statistical procedures used for PanNuke and MoNuSeg are selected according to their respective experimental units and are described in Chapter 3. Statistical results are interpreted together with the direction and magnitude of the observed performance differences rather than as the sole basis for model comparison.

### 2.2.5 Qualitative Analysis and Oracle Diagnostics

Quantitative metrics summarise overall segmentation behaviour but do not show where competing decoders agree or disagree. Qualitative analysis is therefore used to complement numerical evaluation.

Contour overlays, enlarged regions, and maps showing decoder disagreement are used to inspect missed nuclei, additional detections, boundary differences, and behaviour in crowded cellular regions. These visualisations are particularly useful when two configurations obtain similar aggregate scores while producing different local predictions. In this dissertation, an oracle diagnostic is additionally used to estimate the upper bound obtainable if the better of two decoder predictions could be selected using ground-truth performance. Because this selection relies on ground truth, it is not a deployable inference strategy and is not presented as a segmentation method.

A gap between the best fixed decoder and oracle performance indicates that the same decoder is not optimal for every evaluated observation. The size of this gap therefore provides a diagnostic measure of potential decoder complementarity. Practical routing experiments can subsequently test whether observable image characteristics are sufficient to recover any of this potential without access to ground truth.

## 2.3 Choice of Methods

CellViT is selected as the experimental framework because its architecture allows pretrained visual encoders and decoder designs for nuclei segmentation to be examined within a common segmentation model family [8]. This is better aligned with the dissertation’s objective of comparing backbone and decoder choices than comparing several unrelated complete segmentation systems.

ViT-256 and SAM-H are selected as the two backbones because they represent settings that differ substantially in pretraining and model scale. ViT-256 is the ViT-S/16 encoder pretrained on histopathology images and originating from the first stage of HIPT [2], whereas SAM-H is the larger pretrained ViT-H image encoder released as part of SAM [10]. The comparison therefore examines whether performance patterns associated with backbone choice remain consistent across alternative nuclei decoders without assuming in advance that either representation is superior.

HoVer-Net and CPP-Net are selected as the two decoder designs because they use different geometric approaches to nuclei instance reconstruction. HoVer-Net uses horizontal and vertical distance maps to support separation of neighbouring nuclei [6], whereas CPP-Net uses StarDist-style radial distance prediction together with contextual polygon refinement [15, 3]. Evaluating each decoder with both backbones produces the controlled  $2 \times 2$  architecture design used throughout the dissertation.

PanNuke is used for benchmark reproduction because it is the principal dataset used in the original CellViT evaluation [8]. MoNuSeg is then used as a different external histopathology dataset, with its test set of 14 images used for evaluation [11]. The same four configurations trained on PanNuke are evaluated on these images without additional training or fine-tuning. The two settings therefore allow the consistency of backbone and decoder effects to be examined both under the original benchmark and after direct inference on MoNuSeg.

Performance is evaluated using PQ and its decomposition into DQ and SQ, together with bPQ, mPQ where class labels are available, and detection precision, recall, and  $F_1$ . Paired statistical analysis is used as supporting evidence for the consistency of model differences, while qualitative disagreement visualisations and oracle diagnostics are used to investigate decoder-level complementarity.

Together, these methods provide a coherent basis for evaluating reproduction fidelity, backbone effects, decoder effects, behaviour across datasets, and decoder complementarity.

# Chapter 3

## Datasets and Experimental Design

### 3.1 Datasets

This dissertation uses two publicly available histopathology datasets, PanNuke and MoNuSeg. The datasets serve different roles in the empirical investigation. PanNuke provides the primary benchmark for reproducing and analysing the CellViT configurations, whereas MoNuSeg is used as a different external histopathology dataset for inference-only evaluation.

Using the two datasets in this way allows architectural comparisons to be examined first under the original CellViT benchmark setting and subsequently under direct transfer across datasets without training or fine-tuning on the target dataset.

#### 3.1.1 PanNuke

PanNuke is a multi-tissue H&E histopathology dataset developed for nuclei instance segmentation and classification [5]. The dataset used by CellViT contains 7,904  $256 \times 256$  image patches, approximately 189,744 annotated nuclei, 19 tissue types, and five nucleus categories [5, 8].

The five categories are neoplastic, non-neoplastic epithelial, inflammatory, connective, and dead nuclei. Annotation was carried out through a semi-automated process, with the resulting labels reviewed by clinical pathologists [5].

PanNuke provides three predefined data folds. The dataset authors constructed these as randomised training, validation, and testing folds while preserving representation across tissue and nucleus categories [5]. The CellViT implementation subsequently defines three fixed experimental assignments based on these folds. Those assignments are reproduced in this dissertation and are described in Section 3.2.3.

PanNuke contains both instance masks and nucleus-type annotations. It therefore supports class-agnostic instance segmentation evaluation as well as classification-aware multi-class evaluation. In this dissertation, it provides the reference setting for assessing reproduction fidelity and for examining backbone and decoder effects.

#### 3.1.2 MoNuSeg

The MoNuSeg benchmark provides H&E histopathology images with annotated nuclear boundaries for multi-organ nuclei instance segmentation [12, 11]. The 30-image dataset introduced by Kumar et al. was subsequently used as the training set for the MoNuSeg

2018 challenge, which introduced a separate 14-image test set [12, 11].

The challenge test set contains 14  $1000 \times 1000$  images from seven organ types: kidney, lung, colon, breast, bladder, prostate, and brain. Two of these organ types were not represented in the challenge training data [11].

Only these 14 test images are used in this dissertation. No MoNuSeg training images are used for model training, fine-tuning, checkpoint selection, or optimisation on the target dataset. MoNuSeg therefore acts only as an inference and evaluation dataset for the models trained on PanNuke.

Following the CellViT preprocessing procedure, the  $1000 \times 1000$  test images are resized to  $1024 \times 1024$  before inference [8]. The accompanying nuclear boundary annotations are converted into instance maps for evaluation. The same prepared test images and ground-truth instance maps are used for all four model configurations.

MoNuSeg is not treated as a second reproduction benchmark for all four configurations. Instead, it provides a different external histopathology setting in which the persistence of the backbone and decoder patterns observed on PanNuke can be examined.

## 3.2 Experimental Design

The empirical investigation follows a controlled two-stage design. Four CellViT configurations are first trained and evaluated under the predefined PanNuke experimental settings. The resulting models are then applied to the MoNuSeg test set under a fixed inference-only protocol.

The same  $2 \times 2$  experimental design is retained across both datasets. This makes it possible to compare backbone effects while holding the decoder fixed and decoder effects while holding the backbone fixed.

### 3.2.1 Research Questions and Experimental Overview

Based on the research gap identified in Chapter 2, the experiments address the following research questions:

1. **Can the reported CellViT behaviour be reproduced under the predefined PanNuke experimental settings?**

This question assesses whether the reproduced pipeline yields results that are consistent with the corresponding values and performance patterns reported in the original CellViT study.

2. **How does the SAM-H backbone compare with ViT-256 under different decoder architectures and datasets?**

This question examines whether performance differences between backbones remain consistent when the decoder and evaluation dataset change.

**3. How do HoVer-Net and CPP-Net compare when evaluated under the same backbone?**

This question examines whether performance differences between decoders remain consistent across backbone settings, metrics, and datasets.

**4. Do the backbone and decoder patterns observed on PanNuke remain visible on MoNuSeg under inference-only transfer?**

This question evaluates consistency across datasets without training, fine-tuning, or checkpoint selection on MoNuSeg.

**5. What decoder complementarity is revealed by oracle analysis, and can simple observable signals recover any of that potential?**

This question treats oracle and routing analyses as exploratory diagnostics rather than as a proposed segmentation method.

The four configurations evaluated throughout the dissertation are shown in Table 3.1.

Table 3.1: CellViT configurations evaluated in this dissertation.

| Configuration | Backbone | Decoder   |
|---------------|----------|-----------|
| HoVer-ViT-256 | ViT-256  | HoVer-Net |
| CPP-ViT-256   | ViT-256  | CPP-Net   |
| HoVer-SAM-H   | SAM-H    | HoVer-Net |
| CPP-SAM-H     | SAM-H    | CPP-Net   |

### 3.2.2 Implementation and Model Configuration

All four configurations are based on the official CellViT implementation [8].

Experiments were conducted on a Linux server equipped with an NVIDIA L40S GPU.

The software environment used Python 3.9.7 and PyTorch 2.4.1 with CUDA 12.4.

The ViT-256 backbone was initialised using the released weights from the first HIPT stage, which had been pretrained on histopathology data [2]. This encoder is a ViT-S/16 trained on  $256 \times 256$  histology patches and was self-supervised using DINO on approximately 104 million TCGA histology patches. CellViT uses this encoder rather than the complete hierarchical HIPT architecture [2, 8].

The SAM-H configurations were initialised using the pretrained ViT-H image encoder released with the Segment Anything Model [10, 8]. Only the SAM-H image encoder is used as the CellViT backbone; SAM’s prompt encoder and original promptable mask decoder are not used in the evaluated CellViT configurations.

The pretrained encoders initialise the corresponding backbones, while the nuclei segmentation decoder and task prediction branches are trained during the PanNuke

experiments. The complete segmentation checkpoints evaluated in this dissertation were therefore produced by the reproduction experiments rather than downloaded as fully trained CellViT segmentation checkpoints.

Training followed the corresponding official CellViT configuration files. The prediction heads, losses, and post-processing used by HoVer-Net and CPP-Net were retained rather than being artificially homogenised across the two decoders. The shared settings used in the experiments are summarised in Table 3.2.

Table 3.2: Shared training settings used in the PanNuke reproduction experiments.

| Parameter              | Value                           |
|------------------------|---------------------------------|
| Epochs                 | 130                             |
| Batch size             | 16                              |
| Optimiser              | AdamW                           |
| Initial learning rate  | $3 \times 10^{-4}$              |
| Weight decay           | $1 \times 10^{-4}$              |
| Scheduler              | Exponential ( $\gamma = 0.85$ ) |
| Encoder unfreeze epoch | 25                              |
| Random seed            | 19                              |
| Mixed precision        | Enabled                         |
| Evaluation checkpoint  | latest_checkpoint.pth           |

Only environment-specific values, including dataset paths, output directories, and logging locations, were changed where required. The architecture and training configuration relevant to model behaviour followed the corresponding CellViT configuration files.

### 3.2.3 PanNuke Reproduction Protocol

The first empirical stage reproduces the CellViT PanNuke experimental setting. Each of the four CellViT configurations is trained under the three predefined CellViT assignments, producing twelve PanNuke training runs in total.

The assignments used in this dissertation are shown in Table 3.3. They are retained from the official CellViT configuration files to maximise comparability with the original experiments.

Table 3.3: PanNuke experimental assignments reproduced from the official CellViT configuration.

| Setting | Training Fold | Validation Fold | Test Fold |
|---------|---------------|-----------------|-----------|
| Fold-1  | 0             | 1               | 2         |
| Fold-2  | 1             | 0               | 2         |
| Fold-3  | 2             | 1               | 0         |

These three settings should not be interpreted as three independent test folds in a conventional cross-validation scheme. In particular, Fold-1 and Fold-2 share the same PanNuke test fold while using different training and validation assignments. The settings are nevertheless reproduced unchanged because the objective of this stage is comparability with the predefined CellViT experiments rather than redesigning the original evaluation protocol.

Each run is evaluated using its corresponding `latest_checkpoint.pth`. Results from the three predefined settings are then averaged to obtain the aggregate PanNuke reproduction results. Where the original CellViT study provides directly corresponding values, these are used to assess reproduction fidelity [8].

The same twelve experiments also support controlled comparisons of backbone and decoder choice. ViT-256 and SAM-H are compared while holding the decoder fixed, while HoVer-Net and CPP-Net are compared while holding the backbone fixed.

### 3.2.4 MoNuSeg External Evaluation Protocol

The second empirical stage examines whether the backbone and decoder patterns observed on PanNuke remain visible under direct transfer to MoNuSeg.

For each of the four configurations, the `latest_checkpoint.pth` from the corresponding Fold-1 PanNuke run is used. Fold-1 is fixed as the evaluation checkpoint for all four configurations and is not selected according to MoNuSeg performance. No MoNuSeg training, fine-tuning, checkpoint selection, or hyperparameter optimisation is performed using the MoNuSeg data.

This protocol differs from the published CellViT MoNuSeg experiment. The original study evaluated its selected CellViT architectures with HoVer-Net decoders using the final models obtained from all three PanNuke training folds [8]. In contrast, the present dissertation uses one fixed Fold-1 checkpoint per CellViT configuration so that the complete  $2 \times 2$  design, including the CPP-Net variants, can be evaluated under one consistent checkpoint selection rule. The resulting MoNuSeg experiment is therefore an extension and controlled external evaluation rather than a direct reproduction of the published CellViT MoNuSeg result.

For inference, the prepared  $1024 \times 1024$  MoNuSeg images are decomposed into

$256 \times 256$  patches with a 64-pixel overlap. Overlapping predictions are subsequently merged before image-level evaluation. The same tiling and merging procedure is applied across all four configurations.

The CellViT MoNuSeg inference pathway used in this study assumes the output structure of CellViT models with HoVer-Net decoders. Additional compatibility logic was therefore required to evaluate CPP-Net checkpoints. The adaptations include loading the dedicated CPP model classes, processing outputs according to the decoder, passing the CPP distance and refined polygon outputs through the appropriate instance reconstruction interface, and adapting overlap merging to avoid dependence on HoVer-specific output keys.

A sigmoid activation is applied to the CPP `dist_map` logits before CPP instance reconstruction. Additional guards are used when processing cells at patch edges and merging outputs from the different decoders. These adaptations are documented in Appendix A.1.

To verify that the changes affected only CPP-Net processing, the original and adapted MoNuSeg inference pathways were compared using HoVer-Net configurations. The resulting bPQ and detection  $F_1$  values agreed to four decimal places. This check supports the interpretation that the adaptations enable CPP compatibility without altering the existing HoVer-Net inference path.

### 3.2.5 Evaluation Protocol

The metric definitions are introduced in Chapter 2. This section specifies how they are applied in the experiments.

Panoptic Quality (PQ) metrics match predicted and ground-truth instance masks using the standard  $IoU > 0.5$  criterion. Binary Panoptic Quality (bPQ) treats all nuclei as one class. For PanNuke, multi-class Panoptic Quality (mPQ) is additionally calculated using the five annotated nucleus categories [5, 8].

Detection precision, recall, and detection  $F_1$  use a different matching procedure. Following the PanNuke and CellViT detection evaluation, predicted and ground-truth nucleus centroids are matched using a 12-pixel distance criterion at  $0.25 \mu\text{m}/\text{px}$  [5, 8]. Unmatched predictions contribute false positives, while unmatched ground-truth nuclei contribute false negatives.

This distinction is important because detection  $F_1$  and Detection Quality (DQ) are not produced by the same matching rule in the evaluation pipeline. DQ is the detection component of PQ and therefore follows PQ’s instance overlap matching, whereas the separately reported detection  $F_1$  uses matching based on centroid distance.

For PanNuke, evaluation is conducted independently for each of the three predefined experimental settings and the resulting metrics are averaged for the aggregate reproduction analysis.

For MoNuSeg, each of the 14 test images is evaluated separately under all four configurations. Image-level bPQ, detection  $F_1$ , bDQ, and bSQ are retained so that corresponding predictions on the same images can be compared using paired statistical analysis.

### 3.2.6 Statistical, Qualitative, and Diagnostic Analysis

The statistical treatment reflects the different experimental structures of PanNuke and MoNuSeg.

For PanNuke, corresponding mPQ results from the three predefined experimental settings are compared using paired  $t$ -tests as descriptive supporting analyses. Each comparison therefore contains only three paired observations. Furthermore, because Fold-1 and Fold-2 share the same test fold, these observations do not constitute three independent test sets. The resulting  $p$ -values are therefore interpreted descriptively rather than as strong confirmatory inferential evidence.

For MoNuSeg, all four configurations are evaluated on the same 14 test images. Paired Wilcoxon signed-rank tests are used at image level to compare the two backbones under a fixed decoder and the two decoders under a fixed backbone. The analysed metrics are bPQ, detection  $F_1$ , bDQ, and bSQ. The reported  $p$ -values are not adjusted for multiple comparisons;  $p$ -values below 0.05 are therefore treated as nominally significant and interpreted together with the direction and magnitude of the observed differences. Quantitative analysis is complemented by qualitative comparison of HoVer-Net and CPP-Net predictions on MoNuSeg. Contour overlays, enlarged regions, and maps of decoder disagreement are used to inspect local differences in detections and boundaries across images with different nuclear densities. These examples are used for interpretation rather than as an independent quantitative test.

Exploratory oracle analysis is then used to estimate the performance attainable if the better decoder could be selected using ground-truth performance. The oracle is therefore non-deployable and is interpreted only as a diagnostic measure of potential decoder complementarity.

Simple image-level routing rules based on observable signals related to nuclear density, including predicted nucleus count and predicted nuclear area, are subsequently examined. Threshold selection is performed in a leave-one-image-out manner so that the image being evaluated is not simultaneously used to determine its routing rule.

Because decoder differences may also occur locally within an image, a patch-level diagnostic using non-overlapping  $256 \times 256$  regions is used to examine complementarity at a finer scale. These routing analyses are exploratory: their purpose is to determine whether decoder disagreement contains useful and predictable structure, not to introduce or claim a new segmentation architecture.



# Chapter 4

## Results of the Empirical Investigation

### 4.1 PanNuke Reproduction and Configuration Results

The first empirical stage evaluated four CellViT configurations on PanNuke: ViT-256 with the HoVer-Net decoder, SAM-H with the HoVer-Net decoder, ViT-256 with the CPP-Net decoder, and SAM-H with the CPP-Net decoder. The experiment had two purposes. First, it assessed whether the CellViT experimental pipeline could reproduce the performance reported in the original study. Second, it provided a controlled comparison of backbone and decoder choice under the PanNuke benchmark setting. All PanNuke results in this section are averaged across the three predefined PanNuke experimental settings used by CellViT.

#### 4.1.1 Reproduction against Published CellViT Results

Reproduction fidelity was assessed by comparing the reproduced results with the corresponding values reported in the original CellViT study [8]. The comparison covers overall nuclei detection, overall instance segmentation, and panoptic quality for individual nucleus categories. For nuclei detection, Table 4.1 compares precision, recall, and detection  $F_1$  across the four configurations, with differences calculated as reproduced minus published values.

Table 4.1: PanNuke nuclei detection reproduction results.

| Configuration | Precision |         |         | Recall  |         |         | Detection $F_1$ |         |         |
|---------------|-----------|---------|---------|---------|---------|---------|-----------------|---------|---------|
|               | Reprod.   | CellViT | Diff.   | Reprod. | CellViT | Diff.   | Reprod.         | CellViT | Diff.   |
| HoVer-ViT-256 | 0.8223    | 0.8300  | -0.0077 | 0.8259  | 0.8200  | +0.0059 | 0.8240          | 0.8200  | +0.0040 |
| HoVer-SAM-H   | 0.8421    | 0.8400  | +0.0021 | 0.8153  | 0.8100  | +0.0053 | 0.8285          | 0.8300  | -0.0015 |
| CPP-ViT-256   | 0.8528    | 0.8700  | -0.0172 | 0.7875  | 0.7600  | +0.0275 | 0.8189          | 0.8100  | +0.0089 |
| CPP-SAM-H     | 0.8572    | 0.8700  | -0.0128 | 0.7970  | 0.7800  | +0.0170 | 0.8260          | 0.8200  | +0.0060 |

The reproduced detection results remain close to the published CellViT values across all four configurations. Detection  $F_1$  differs from the published result by less than 0.009 in every case. The two HoVer-Net configurations also reproduce the reported precision and recall values closely. Somewhat larger deviations are observed for the CPP-Net configurations, particularly in recall. Nevertheless, the reproduced results retain the same overall pattern in precision and recall reported by CellViT: CPP-Net produces higher precision but lower recall than HoVer-Net.

For overall instance segmentation, published aggregate bPQ and mPQ values are available for the two HoVer-Net configurations but not for the corresponding CPP-Net

configurations. Table 4.2 therefore compares the reproduced and published HoVer-Net results, again with differences calculated as reproduced minus published values.

Table 4.2: PanNuke instance segmentation reproduction results.

| Configuration | bPQ     |         |         | mPQ     |         |         |
|---------------|---------|---------|---------|---------|---------|---------|
|               | Reprod. | CellViT | Diff.   | Reprod. | CellViT | Diff.   |
| HoVer-ViT-256 | 0.6531  | 0.6696  | -0.0165 | 0.4825  | 0.4846  | -0.0021 |
| HoVer-SAM-H   | 0.6647  | 0.6793  | -0.0146 | 0.4964  | 0.4980  | -0.0016 |

The reproduced mPQ values are particularly close to the published results. For ViT-256 + HoVer, the reproduced mPQ is 0.4825 compared with 0.4846 reported by CellViT, corresponding to an absolute difference of 0.0021. For SAM-H + HoVer, the reproduced value is 0.4964 compared with 0.4980, an absolute difference of 0.0016.

The bPQ differences are larger than the corresponding mPQ differences. ViT-256 + HoVer obtains a reproduced bPQ of 0.6531 compared with 0.6696 in the original study, while SAM-H + HoVer obtains 0.6647 compared with 0.6793. The absolute differences are therefore 0.0165 and 0.0146, respectively. Despite these deviations, the relative ordering of the two backbones is preserved for both bPQ and mPQ, with SAM-H outperforming ViT-256 in both the reproduced and published results.

The reproduction comparison was further examined for individual nucleus categories using the panoptic quality values reported by CellViT [8]. Table 4.3 compares the five nucleus categories; Neo., Epi., Inf., and Con. denote neoplastic, non-neoplastic epithelial, inflammatory, and connective nuclei, respectively.

Table 4.3: PanNuke class-wise PQ reproduction results.

| Configuration | Our Reproduction |        |        |        |        | Reported in CellViT |       |       |       |       |
|---------------|------------------|--------|--------|--------|--------|---------------------|-------|-------|-------|-------|
|               | Neo.             | Epi.   | Inf.   | Con.   | Dead   | Neo.                | Epi.  | Inf.  | Con.  | Dead  |
| HoVer-ViT-256 | 0.5656           | 0.5562 | 0.4234 | 0.4025 | 0.1387 | 0.567               | 0.559 | 0.405 | 0.405 | 0.144 |
| HoVer-SAM-H   | 0.5790           | 0.5765 | 0.4239 | 0.4213 | 0.1586 | 0.581               | 0.583 | 0.417 | 0.423 | 0.149 |
| CPP-ViT-256   | 0.5422           | 0.5332 | 0.4065 | 0.3727 | 0.1303 | 0.540               | 0.524 | 0.414 | 0.369 | 0.133 |
| CPP-SAM-H     | 0.5717           | 0.5634 | 0.4111 | 0.3931 | 0.1385 | 0.571               | 0.565 | 0.405 | 0.395 | 0.131 |

The PQ results for individual nucleus categories also show close agreement with the published CellViT results. The largest absolute difference is 0.0184 for the inflammatory class under ViT-256 + HoVer, for which the reproduced PQ is 0.4234 compared with the published value of 0.405. The differences for the remaining categories are smaller. In addition to the numerical agreement, the relative performance patterns across the four configurations are broadly preserved.

Taken together, the overall detection and segmentation comparisons, along with the results for individual nucleus categories, show that the reproduced PanNuke results closely match the published CellViT benchmarks where direct comparison is available.

The reproduction therefore provides a sufficiently consistent empirical baseline for examining backbone and decoder effects in the following subsections.

### 4.1.2 Backbone Results

The PanNuke results show a consistent advantage for the SAM-H backbone over ViT-256 under both decoder architectures.

Under HoVer-Net, replacing ViT-256 with SAM-H increases bPQ from 0.6531 to 0.6647, mPQ from 0.4825 to 0.4964, and detection  $F_1$  from 0.8240 to 0.8285. The corresponding absolute mPQ improvement is 0.0139.

Under CPP-Net, replacing ViT-256 with SAM-H increases bPQ from 0.6249 to 0.6450, mPQ from 0.4564 to 0.4775, and detection  $F_1$  from 0.8189 to 0.8260. The corresponding mPQ improvement is approximately 0.021. As shown in Figure 4.1, SAM-H achieves higher mPQ than ViT-256 under both decoder settings, although the magnitude of the improvement differs between the two decoders.

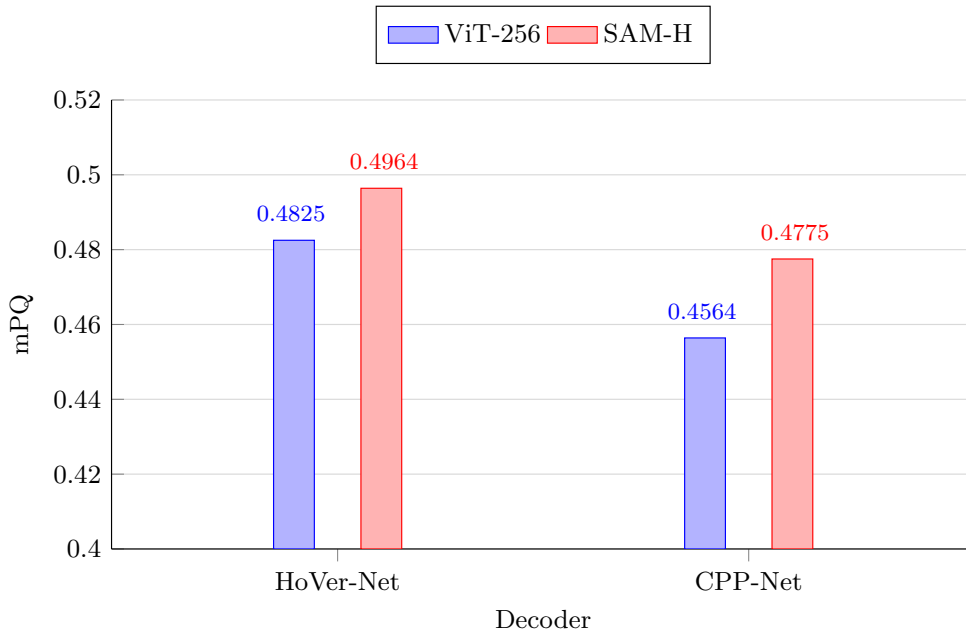


Figure 4.1: Backbone comparison of PanNuke mPQ.

### 4.1.3 Decoder Results

The decoder comparison shows a different pattern from the backbone comparison. HoVer-Net achieves higher aggregate bPQ and mPQ than CPP-Net under both backbone settings.

Under ViT-256, HoVer-Net achieves an mPQ of 0.4825 compared with 0.4564 for CPP-Net, corresponding to an absolute difference of 0.0261. Under SAM-H, HoVer-Net achieves an mPQ of 0.4964 compared with 0.4775 for CPP-Net, corresponding to an

absolute difference of 0.0189. As shown in Figure 4.2, HoVer-Net achieves higher mPQ under both backbones, and the same overall direction is observed for bPQ.

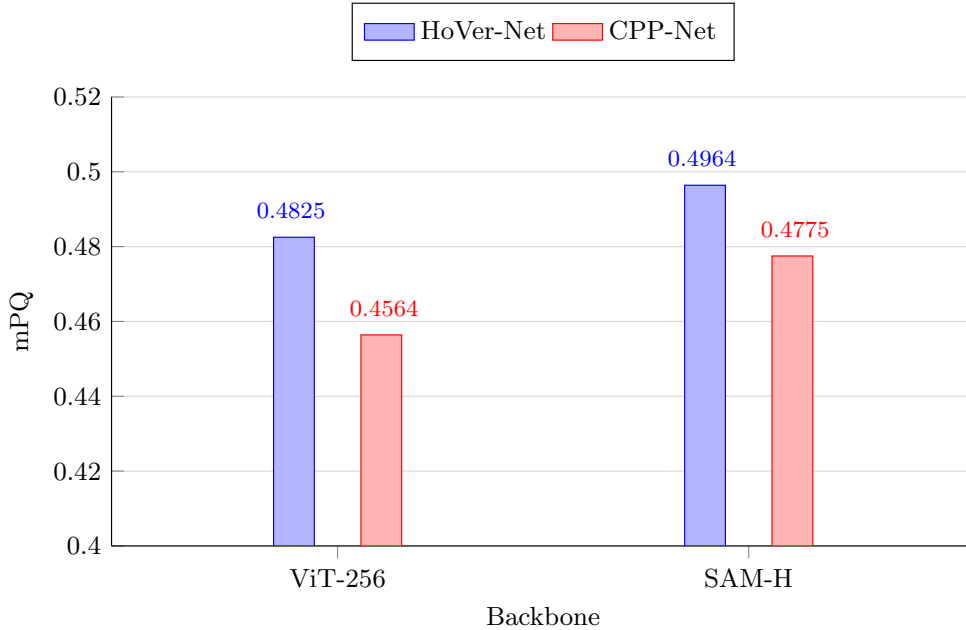


Figure 4.2: Decoder comparison of PanNuke mPQ.

The detection results show that this decoder comparison is not a simple dominance relationship. CPP-Net achieves higher detection precision than HoVer-Net under both backbones, whereas HoVer-Net achieves higher recall.

Under ViT-256, CPP-Net obtains a precision of 0.8528 compared with 0.8223 for HoVer-Net. In contrast, HoVer-Net obtains a recall of 0.8259 compared with 0.7875 for CPP-Net. Under SAM-H, CPP-Net obtains a precision of 0.8572 compared with 0.8421 for HoVer-Net, whereas HoVer-Net obtains a recall of 0.8153 compared with 0.7970 for CPP-Net.

The reproduced PanNuke results therefore show a consistent precision-recall trade-off between the two decoders. As shown in Figure 4.3, CPP-Net favours higher detection precision, while HoVer-Net favours higher recall and achieves higher aggregate PQ.

Overall, the PanNuke experiment establishes two main empirical patterns. First, SAM-H produces higher aggregate performance than ViT-256 under both decoder architectures. Second, HoVer-Net produces higher aggregate bPQ and mPQ than CPP-Net, although the two decoders exhibit different precision and recall behaviour.

## 4.2 MoNuSeg External Evaluation Results

The second empirical stage evaluated the same four CellViT configurations on MoNuSeg. As described in Chapter 3, the corresponding Fold-1 checkpoint trained on PanNuke was used for each configuration without additional training or fine-tuning and

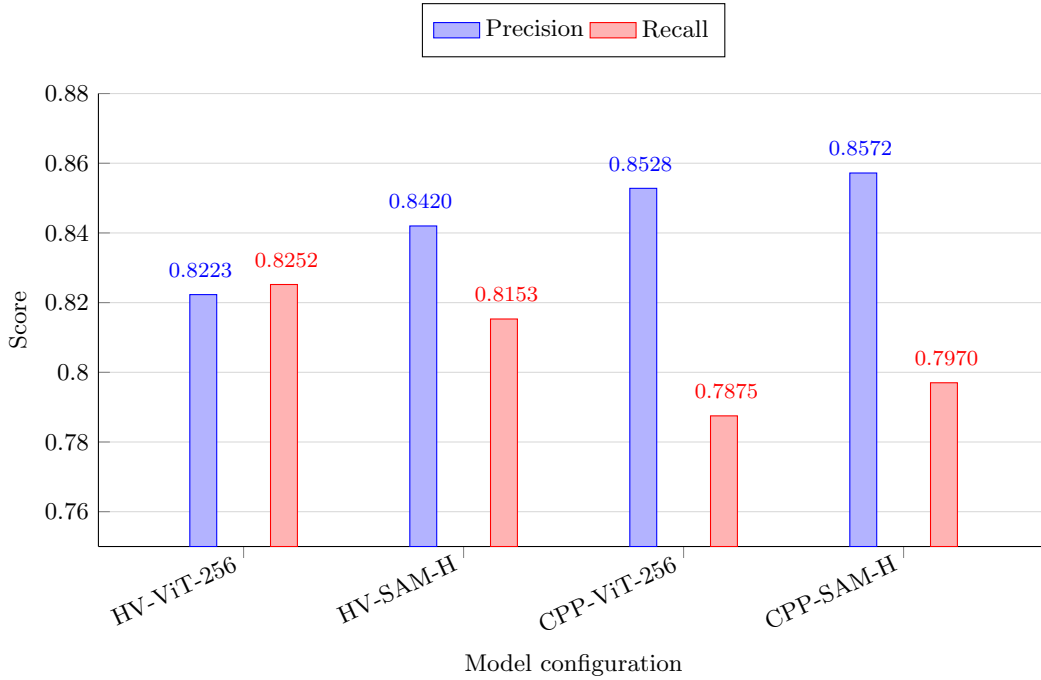


Figure 4.3: PanNuke nuclei detection precision and recall.

without using MoNuSeg for checkpoint selection or hyperparameter optimisation. MoNuSeg is evaluated as a binary nuclei instance segmentation dataset. The analysis therefore focuses on bPQ, detection  $F_1$ , bDQ, and bSQ. Unlike the PanNuke experiment, the purpose of this stage is not to establish a second direct reproduction benchmark for all four configurations, but to examine whether the backbone and decoder patterns observed on PanNuke remain visible on a different external histopathology dataset.

#### 4.2.1 Overall Performance

Across the 14 MoNuSeg test images, the four configurations achieve relatively similar performance across bPQ, detection  $F_1$ , bDQ, and bSQ. These results are summarised in Table 4.4, although the ranking depends on the metric considered.

Table 4.4: MoNuSeg performance across the four configurations.

| Model configuration | bPQ    | Detection $F_1$ | bDQ    | bSQ    |
|---------------------|--------|-----------------|--------|--------|
| HoVer-ViT-256       | 0.6593 | 0.8511          | 0.8580 | 0.7679 |
| CPP-ViT-256         | 0.6577 | 0.8489          | 0.8646 | 0.7602 |
| HoVer-SAM-H         | 0.6744 | 0.8638          | 0.8717 | 0.7733 |
| CPP-SAM-H           | 0.6760 | 0.8557          | 0.8766 | 0.7707 |

CPP-SAM-H achieves the highest bPQ at 0.6760 and the highest bDQ at 0.8766. In contrast, HoVer-SAM-H achieves the highest detection  $F_1$  at 0.8638 and the highest bSQ

at 0.7733.

The overall MoNuSeg results therefore retain a clear backbone pattern but do not identify a single decoder that achieves the best result across all four metrics.

### 4.2.2 Backbone Results

The backbone comparison shows a consistent advantage for SAM-H over ViT-256 under both decoder architectures.

Under HoVer-Net, replacing ViT-256 with SAM-H increases bPQ from 0.6593 to 0.6744 and detection  $F_1$  from 0.8511 to 0.8638. The corresponding bDQ increases from 0.8580 to 0.8717, while bSQ increases from 0.7679 to 0.7733.

A similar pattern is observed under CPP-Net. Replacing ViT-256 with SAM-H increases bPQ from 0.6577 to 0.6760 and detection  $F_1$  from 0.8489 to 0.8557. bDQ increases from 0.8646 to 0.8766, while bSQ increases from 0.7602 to 0.7707. As shown in Figure 4.4, SAM-H achieves higher bPQ than ViT-256 with both HoVer-Net and CPP-Net, consistent with the pattern observed across the other evaluation metrics.

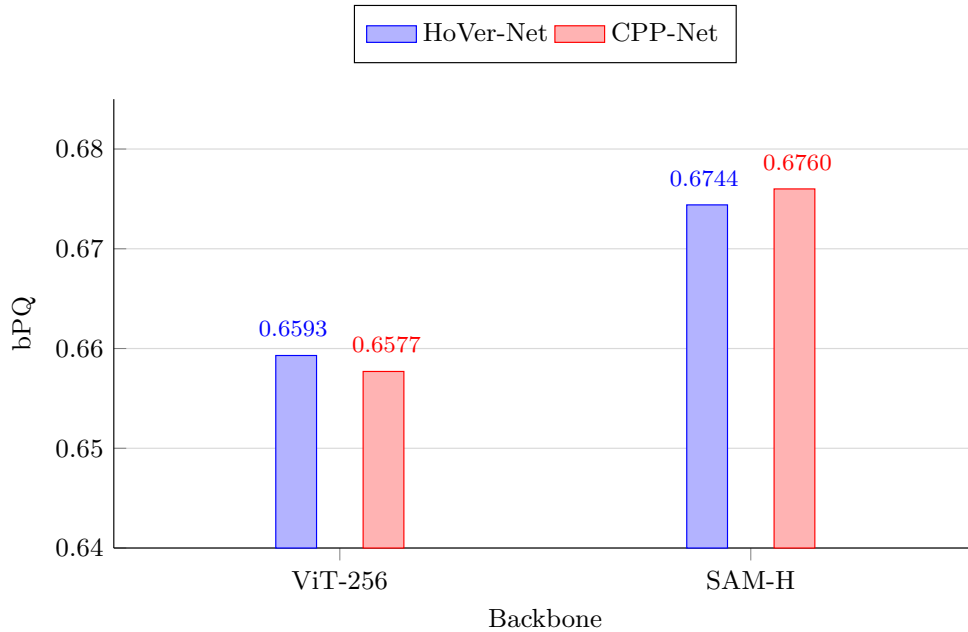


Figure 4.4: Backbone comparison of MoNuSeg bPQ.

Overall, SAM-H achieves higher bPQ, detection  $F_1$ , bDQ, and bSQ than ViT-256 under both decoder settings. The backbone advantage observed on PanNuke therefore remains visible under MoNuSeg external evaluation.

### 4.2.3 Decoder Results

The decoder comparison on MoNuSeg is less consistent than the backbone comparison, with the relative ranking depending on both the backbone and the evaluation metric.

Under ViT-256, HoVer-Net achieves slightly higher bPQ and detection  $F_1$  than CPP-Net. The bPQ values are 0.6593 for HoVer-Net and 0.6577 for CPP-Net. CPP-Net, however, achieves a higher bDQ of 0.8646 compared with 0.8580 for HoVer-Net, while HoVer-Net retains the higher bSQ.

Under SAM-H, the bPQ ranking is reversed. CPP-Net achieves a slightly higher bPQ of 0.6760 compared with 0.6744 for HoVer-Net and also achieves the higher bDQ. In contrast, HoVer-Net produces the higher detection  $F_1$  and bSQ. As shown in Figure 4.5, HoVer-Net is marginally higher under ViT-256, whereas CPP-Net is marginally higher under SAM-H.

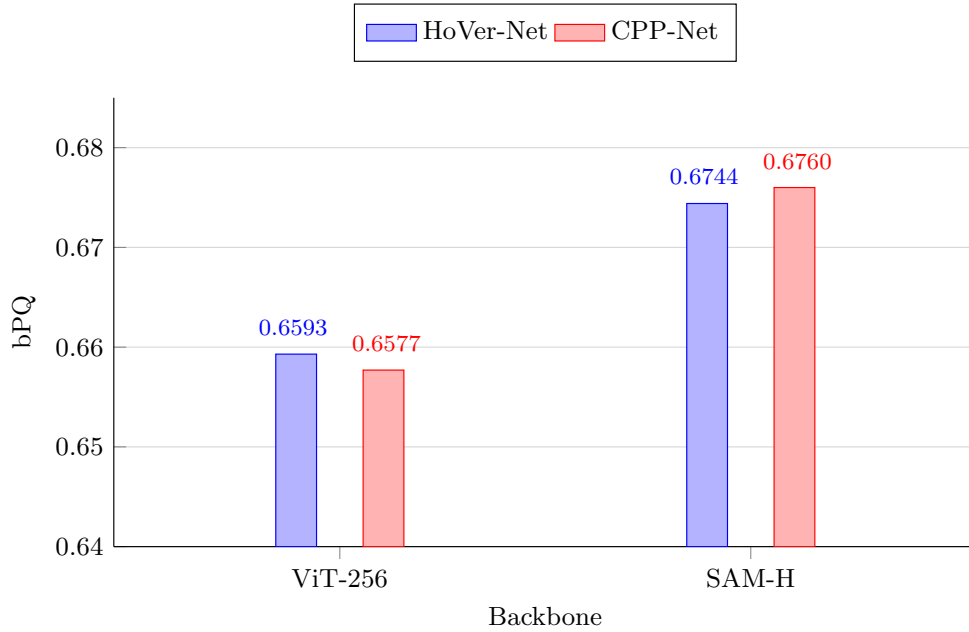


Figure 4.5: Decoder comparison of MoNuSeg bPQ.

The aggregate HoVer-Net advantage observed on PanNuke therefore does not transfer directly to MoNuSeg. Instead, the relative decoder ranking becomes dependent on the backbone and evaluation metric, and the differences between the two decoders remain small in magnitude.

Taken together, the PanNuke and MoNuSeg results show a more consistent backbone effect than decoder effect. SAM-H achieves higher performance than ViT-256 across both decoder settings and datasets, whereas the relative behaviour of HoVer-Net and CPP-Net varies between PanNuke and MoNuSeg.



# Chapter 5

## Validation of Results

### 5.1 Statistical Validation

The results in Chapter 4 identified two main patterns. SAM-H achieved higher aggregate performance than ViT-256 under both decoder architectures, whereas the relative performance of HoVer-Net and CPP-Net was less consistent across datasets and metrics. This section examines whether these patterns are also supported by the paired observations available from the PanNuke and MoNuSeg experiments.

#### 5.1.1 PanNuke Statistical Analysis

The PanNuke results showed higher mPQ for SAM-H than ViT-256 under both decoder architectures, and higher mPQ for HoVer-Net than CPP-Net under both backbones. To examine whether these differences were consistent across the three predefined CellViT experimental settings, paired comparisons were performed using the corresponding mPQ values.

Only three paired observations are available for each comparison. In addition, the three settings do not form three independent cross-validation test folds because Fold-1 and Fold-2 share the same test fold. Two-sided paired  $t$ -tests are therefore used only as descriptive supporting analyses rather than as confirmatory statistical tests. Each test has two degrees of freedom, and positive mean differences favour the first model named in the comparison; the resulting statistics are summarised in Table 5.1.

Table 5.1: Descriptive paired  $t$ -test results on PanNuke.

| Comparison                | Metric | Mean difference | Paired $t$ | $p$ -value |
|---------------------------|--------|-----------------|------------|------------|
| SAM-H vs. ViT-256 (HoVer) | mPQ    | 0.0139          | 24.5510    | 0.0017     |
| SAM-H vs. ViT-256 (CPP)   | mPQ    | 0.0210          | 11.3237    | 0.0077     |
| HoVer vs. CPP (ViT-256)   | mPQ    | 0.0261          | 31.0979    | 0.0010     |
| HoVer vs. CPP (SAM-H)     | mPQ    | 0.0189          | 7.6217     | 0.0168     |

The paired comparisons follow the same direction as the aggregate PanNuke results. SAM-H achieves higher mPQ than ViT-256 under both HoVer-Net and CPP-Net, with mean differences of 0.0139 and 0.0210, respectively. HoVer-Net also achieves higher mPQ than CPP-Net under both ViT-256 and SAM-H, with corresponding mean differences of 0.0261 and 0.0189.

These results indicate that the backbone and decoder patterns reported in Chapter 4 are consistent across the three predefined PanNuke settings. However, the small number of observations and the shared test fold substantially limit the statistical interpretation. The PanNuke analysis is therefore treated as descriptive evidence supporting the reproduced trends.

### 5.1.2 MoNuSeg Statistical Analysis

The same 14 MoNuSeg test images were evaluated using all four model configurations, allowing direct paired comparisons between corresponding predictions. Two-sided paired Wilcoxon signed-rank tests were used because no assumption was made that the image-level paired differences followed a normal distribution.

Four metrics were examined: bPQ, detection  $F_1$ , bDQ, and bSQ. Comparisons were made between the two decoders under the same backbone and between the two backbones under the same decoder.

The reported  $p$ -values are not adjusted for multiple comparisons. Values below the conventional  $p < 0.05$  threshold are therefore described as nominally significant and are interpreted together with the direction and magnitude of the observed differences. The complete paired comparisons are summarised in Table 5.2.

Table 5.2: Paired Wilcoxon test results on MoNuSeg.

| Comparison                | Metric          | $W$  | $p$ -value | Interpretation        |
|---------------------------|-----------------|------|------------|-----------------------|
| CPP vs. HoVer (ViT-256)   | bPQ             | 51.0 | 0.9515     | Not significant       |
| CPP vs. HoVer (ViT-256)   | Detection $F_1$ | 51.0 | 0.9515     | Not significant       |
| CPP vs. HoVer (ViT-256)   | bDQ             | 28.0 | 0.1353     | Not significant       |
| CPP vs. HoVer (ViT-256)   | bSQ             | 14.0 | 0.0134     | Nominally significant |
| CPP vs. HoVer (SAM-H)     | bPQ             | 40.0 | 0.4325     | Not significant       |
| CPP vs. HoVer (SAM-H)     | Detection $F_1$ | 41.0 | 0.5016     | Not significant       |
| CPP vs. HoVer (SAM-H)     | bDQ             | 19.0 | 0.0353     | Nominally significant |
| CPP vs. HoVer (SAM-H)     | bSQ             | 23.0 | 0.0640     | Not significant       |
| SAM-H vs. ViT-256 (HoVer) | bPQ             | 3.0  | 0.0006     | Nominally significant |
| SAM-H vs. ViT-256 (HoVer) | Detection $F_1$ | 33.0 | 0.2412     | Not significant       |
| SAM-H vs. ViT-256 (HoVer) | bDQ             | 8.0  | 0.0031     | Nominally significant |
| SAM-H vs. ViT-256 (HoVer) | bSQ             | 2.0  | 0.0015     | Nominally significant |
| SAM-H vs. ViT-256 (CPP)   | bPQ             | 0.0  | 0.0001     | Nominally significant |
| SAM-H vs. ViT-256 (CPP)   | Detection $F_1$ | 50.0 | 0.8753     | Not significant       |
| SAM-H vs. ViT-256 (CPP)   | bDQ             | 9.0  | 0.0040     | Nominally significant |
| SAM-H vs. ViT-256 (CPP)   | bSQ             | 0.0  | 0.0001     | Nominally significant |

The decoder comparisons provide little evidence for a consistent difference in overall panoptic quality. Under ViT-256, the bPQ comparison between CPP-Net and

HoVer-Net gives  $p = 0.9515$ , while under SAM-H the corresponding value is  $p = 0.4325$ . Neither comparison reaches the nominal significance threshold.

Differences between the decoders are instead confined to particular components of PQ. Under ViT-256, the difference in bSQ is nominally significant ( $p = 0.0134$ ), whereas the difference in bDQ is not. Under SAM-H, the opposite pattern is observed: the difference in bDQ is nominally significant ( $p = 0.0353$ ), while the difference in bSQ is not ( $p = 0.0640$ ). This is consistent with the results in Chapter 4, where the relative performance of HoVer-Net and CPP-Net depended on the metric being considered.

The backbone comparisons show a clearer pattern. SAM-H achieves higher bPQ, bDQ, and bSQ than ViT-256 under both decoder architectures, and the unadjusted  $p$ -values for all six corresponding comparisons are below 0.05. For HoVer-Net, the comparisons give  $p = 0.0006$  for bPQ,  $p = 0.0031$  for bDQ, and  $p = 0.0015$  for bSQ. Under CPP-Net, the corresponding values are  $p = 0.0001$ ,  $p = 0.0040$ , and  $p = 0.0001$ .

Detection  $F_1$  does not show the same statistical pattern. Although the aggregate values reported in Chapter 4 are higher for SAM-H under both decoders, the paired comparisons give  $p = 0.2412$  under HoVer-Net and  $p = 0.8753$  under CPP-Net.

Overall, the MoNuSeg analysis provides clearer support for the SAM-H advantage in panoptic quality and in its DQ and SQ components than for an advantage in detection  $F_1$ . In contrast, the comparison between HoVer-Net and CPP-Net remains dependent on the metric being examined.

## 5.2 Qualitative Comparison of Decoder Predictions

The aggregate bPQ values of HoVer-Net and CPP-Net are close on MoNuSeg, and the paired comparisons do not show a nominally significant difference under either backbone. Differences nevertheless appear in individual detection and segmentation metrics.

To select representative cases systematically, the 14 MoNuSeg test images were divided into low-, medium-, and high-density groups using quantiles of the ground-truth nucleus count. Within each density group, the image showing the largest disagreement between the HoVer-Net and CPP-Net binary prediction masks under the SAM-H backbone was selected. Disagreement was measured as one minus the Dice similarity between the two masks. The selected cases are therefore intended to illustrate decoder disagreement at different nuclear densities rather than to represent an unbiased sample of all MoNuSeg images. These cases are shown in Figure 5.1, which compares the original images, predicted contours, and regions of disagreement between the two decoders. In the contour overlays, green denotes ground-truth contours, blue denotes HoVer-Net contours, and red denotes CPP-Net contours. In the disagreement maps, green indicates regions predicted by both decoders, blue indicates HoVer-Net-only regions, and red indicates CPP-Net-only regions.

A large proportion of nuclei are predicted by both decoders, consistent with their similar

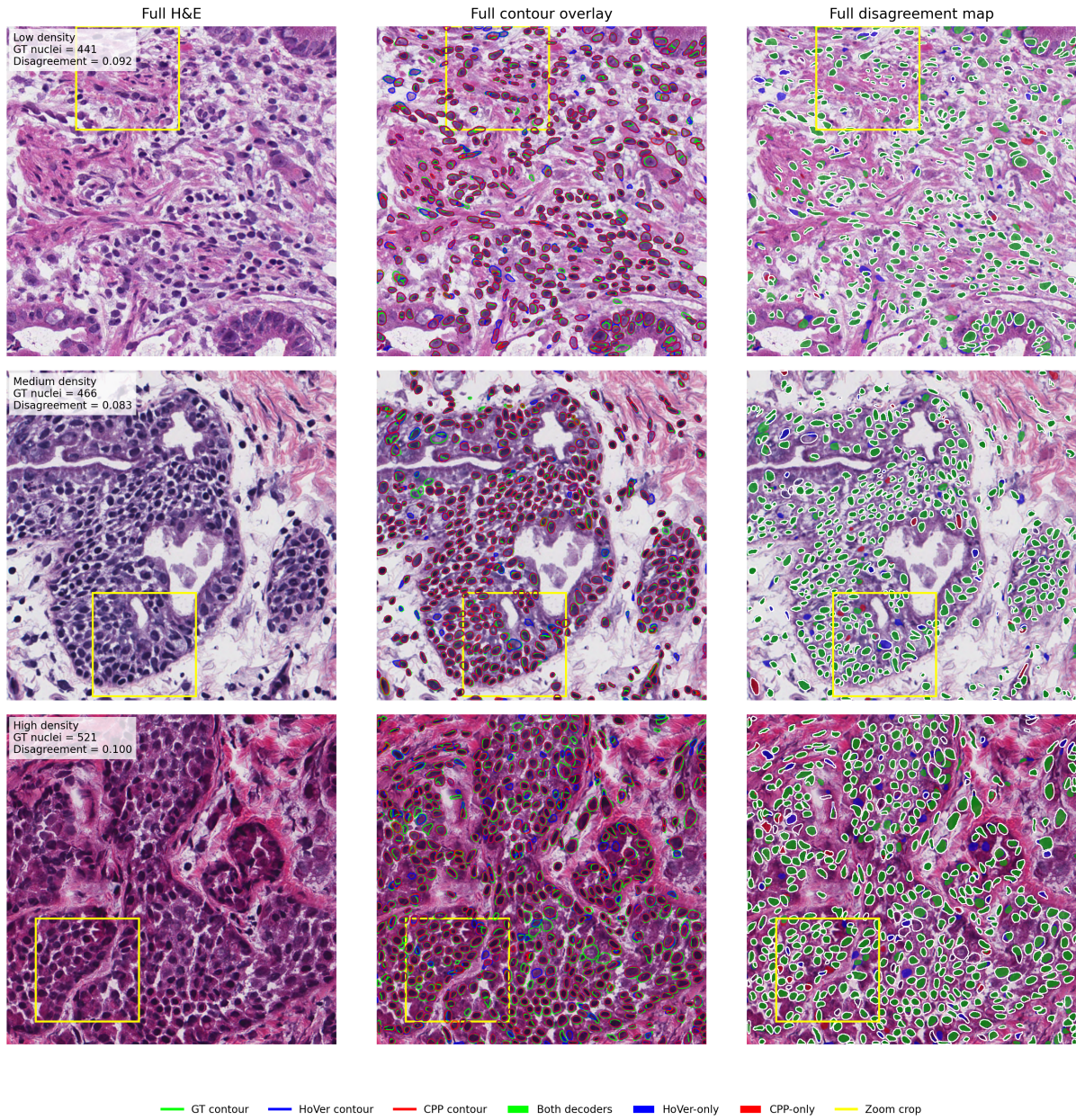


Figure 5.1: Full-context MoNuSeg decoder comparison under SAM-H.

aggregate bPQ values. However, the disagreement maps also contain regions predicted only by HoVer-Net or only by CPP-Net. The predictions are therefore not identical despite the relatively small difference in overall performance.

In the low-density example, disagreement is relatively limited and occurs around several small or weakly stained nuclei. In the medium-density example, differences appear within moderately crowded groups of cells. The high-density example contains more visible disagreement in regions where neighbouring nuclei are closely packed. Enlarged regions from the same three examples are shown in Figure 5.2, using the same colour scheme as Figure 5.1.

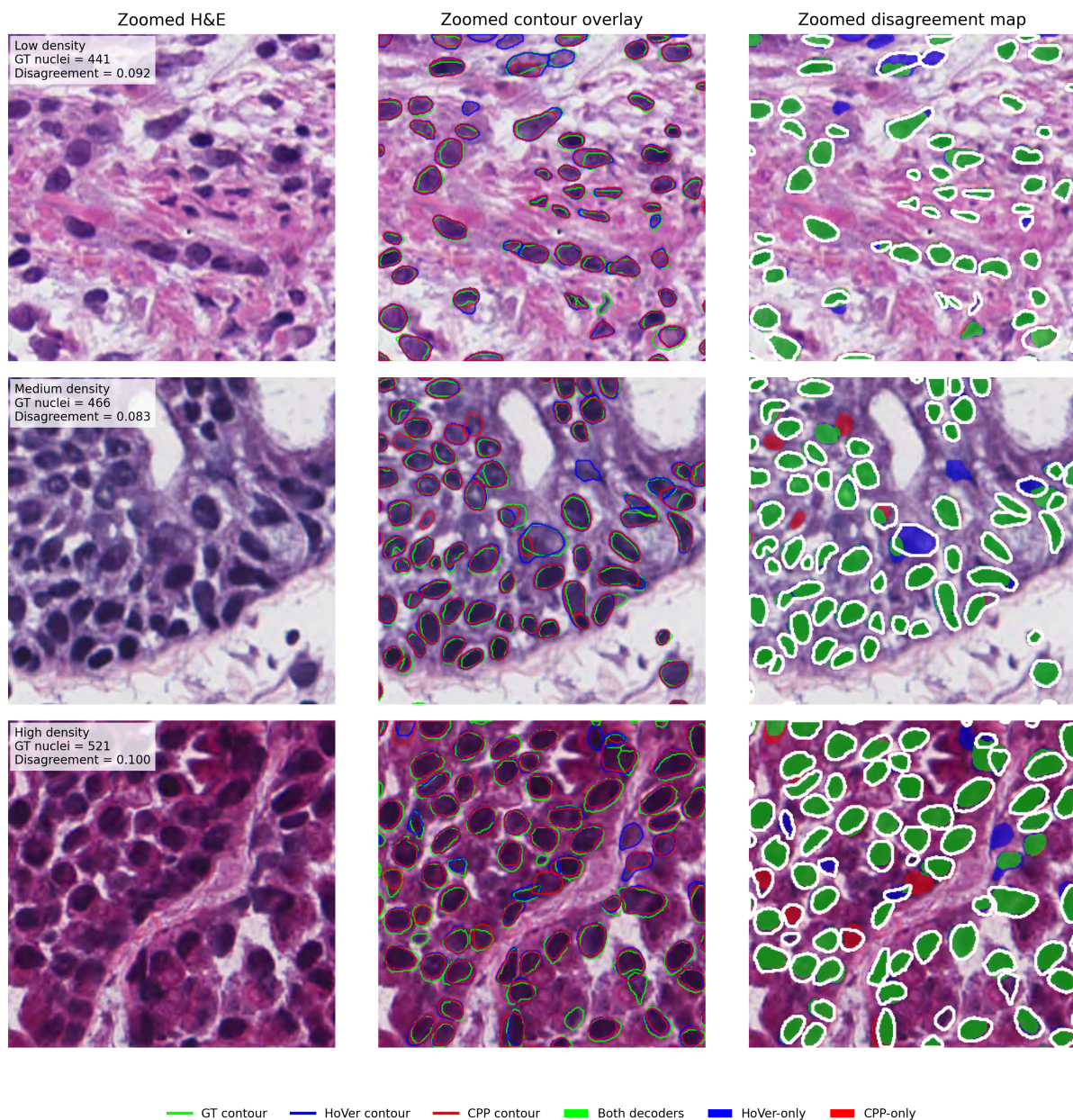


Figure 5.2: Zoomed MoNuSeg decoder comparison under SAM-H.

The enlarged regions show differences around faint nuclei, touching boundaries, and crowded cell groups. These examples show that the two decoders can produce different

predictions for particular nuclei and boundaries. However, the disagreement maps alone do not determine which prediction is correct. A nucleus predicted by only one decoder may represent a correct additional detection, a false positive, a missed detection by the other decoder, or a difference in boundary placement.

The qualitative comparison therefore does not establish that either decoder is generally superior. Instead, it shows that HoVer-Net and CPP-Net can produce different predictions even when their aggregate bPQ values are similar. This observation motivates the following analysis, which examines whether choosing between the two decoders could provide any measurable performance benefit.

### 5.3 Decoder Routing Analysis

The qualitative comparison confirms that HoVer-Net and CPP-Net do not always make the same predictions. The next question is whether these differences contain sufficient complementary information for decoder selection to improve performance.

Three analyses were performed. First, an oracle was used to estimate the maximum improvement available if the better decoder could be selected perfectly for each evaluation unit. Second, simple prediction-derived features describing nuclei abundance and predicted nuclear area were tested as possible selection signals. Finally, the predictions were compared within smaller regions of each image to determine whether decoder differences become more apparent locally.

These experiments are exploratory and are not presented as a new segmentation method.

#### 5.3.1 Oracle Analysis

The first analysis compares the best fixed decoder with an oracle that selects the better decoder separately for each MoNuSeg image according to the ground-truth metric value. The best fixed decoder is the single decoder with the higher mean performance for the metric being considered. The oracle instead chooses HoVer-Net or CPP-Net separately for each image. Because this selection uses ground-truth performance, it cannot be applied during inference. Its purpose is only to estimate the maximum improvement available from perfect image-level selection between the two evaluated decoders. For comparison, the best tested routing rule based only on prediction-derived signals is also reported, with the SAM-H results summarised in Table 5.3.

Table 5.3: SAM-H image-level oracle and routing results.

| Metric          | Best fixed | Oracle | Best prediction-based routing | Routing gain over best fixed |
|-----------------|------------|--------|-------------------------------|------------------------------|
| bPQ             | 0.6760     | 0.6783 | 0.6742                        | −0.0018                      |
| Detection $F_1$ | 0.8638     | 0.8744 | 0.8704                        | +0.0066                      |
| bDQ             | 0.8766     | 0.8779 | 0.8754                        | −0.0012                      |
| bSQ             | 0.7733     | 0.7744 | 0.7733                        | +0.0000                      |

The oracle improves on the best fixed decoder for all four metrics, showing that the better decoder is not the same for every image. However, the potential improvements are small for bPQ, bDQ, and bSQ. The largest difference occurs for detection  $F_1$ , which increases from 0.8638 for the best fixed decoder to 0.8744 under oracle selection.

The best tested prediction-based routing rule does not reproduce these improvements consistently. Detection  $F_1$  increases to 0.8704, but the routed result is lower than the best fixed decoder for bPQ and bDQ and is unchanged for bSQ. The corresponding ViT-256 comparison is shown in Table 5.4.

Table 5.4: ViT-256 image-level oracle and routing results.

| Metric          | Best fixed | Oracle | Best prediction-based routing | Routing gain over best fixed |
|-----------------|------------|--------|-------------------------------|------------------------------|
| bPQ             | 0.6593     | 0.6636 | 0.6602                        | +0.0009                      |
| Detection $F_1$ | 0.8511     | 0.8642 | 0.8509                        | −0.0002                      |
| bDQ             | 0.8646     | 0.8678 | 0.8625                        | −0.0021                      |
| bSQ             | 0.7679     | 0.7690 | 0.7690                        | +0.0011                      |

The ViT-256 results show a similar pattern. Oracle selection again improves on the best fixed decoder, although the gains remain modest. The best tested prediction-based routing rule produces small improvements for bPQ and bSQ but performs slightly below the best fixed decoder for detection  $F_1$  and bDQ.

The oracle results therefore show that the relative performance of the two decoders varies across the MoNuSeg images. However, the maximum improvements are limited for most metrics, and the simple selection rules do not recover them consistently.

### 5.3.2 Routing Using Global Image Features

The next analysis examined whether simple image-level properties were associated with the relative performance of the two decoders.

For diagnostic purposes, the correlation analysis included ground-truth nucleus count together with prediction-derived signals such as predicted nucleus count and predicted nuclear area fraction. Ground-truth nucleus count was used only to investigate whether true image density was associated with decoder behaviour and was not used as a deployable routing signal. The prediction-based routing rules themselves relied on quantities available from the model predictions. Correlations between these features and the image-level decoder performance differences across the 14 test images are summarised in Table 5.5.

Most of the correlations are weak. For bPQ under SAM-H, the correlations range from approximately  $-0.12$  to  $-0.17$ , indicating little association between the tested density measures and the relative performance of the two decoders. Under ViT-256, the relationship between ground-truth nuclei count and the bPQ difference is also close to zero.

Table 5.5: Correlations between density features and decoder performance differences on MoNuSeg.

| Backbone | Metric          | Feature                      | Correlation $r$ |
|----------|-----------------|------------------------------|-----------------|
| SAM-H    | bPQ             | Ground-truth nuclei count    | −0.1278         |
| SAM-H    | bPQ             | HoVer predicted nuclei count | −0.1583         |
| SAM-H    | bPQ             | CPP predicted nuclei count   | −0.1510         |
| SAM-H    | bPQ             | HoVer area fraction          | −0.1670         |
| SAM-H    | bPQ             | CPP area fraction            | −0.1202         |
| SAM-H    | bPQ             | Mean area fraction           | −0.1441         |
| SAM-H    | Detection $F_1$ | Ground-truth nuclei count    | +0.1139         |
| SAM-H    | Detection $F_1$ | HoVer area fraction          | +0.3618         |
| SAM-H    | Detection $F_1$ | CPP area fraction            | +0.3601         |
| ViT-256  | bPQ             | Ground-truth nuclei count    | −0.0306         |
| ViT-256  | bPQ             | CPP area fraction            | −0.2583         |
| ViT-256  | Detection $F_1$ | Ground-truth nuclei count    | −0.1824         |
| ViT-256  | Detection $F_1$ | Mean area fraction           | +0.0942         |

The strongest observed relationship occurs for detection  $F_1$  under SAM-H. The HoVer-Net and CPP-Net predicted area fractions both have correlations of approximately  $r = 0.36$ . This relationship is stronger than those observed for bPQ, but it remains moderate and is not reproduced consistently under ViT-256.

These results help explain the limited performance of the routing rules. The tested prediction-derived density signals contain some information about decoder differences, particularly for detection  $F_1$ , but they do not provide a reliable basis for choosing the better decoder across images and metrics.

### 5.3.3 Local Decoder Comparison

The preceding analysis treats each MoNuSeg image as a single unit. However, the qualitative examples suggest that HoVer-Net and CPP-Net can differ within separate regions of the same image. To examine whether these local differences provide greater scope for decoder selection, each  $1024 \times 1024$  MoNuSeg image was divided into 16 non-overlapping  $256 \times 256$  patches. Across the 14 test images, this produced 224 local patches for analysis.

These diagnostic patches are distinct from the overlapping patching procedure used during model inference. They are used only to examine local decoder-level behaviour after the full-image predictions have been generated.

An oracle selected the decoder with the higher Dice score for each patch. As in the preceding oracle analysis, this procedure uses ground-truth performance and is used only

to measure the potential benefit of perfect local selection. The resulting patch-level comparison is summarised in Table 5.6.

Table 5.6: Local MoNuSeg oracle results.

| Backbone | Number of patches | HoVer wins | CPP wins | HoVer mean Dice | CPP mean Dice | Oracle mean Dice |
|----------|-------------------|------------|----------|-----------------|---------------|------------------|
| SAM-H    | 224               | 134        | 90       | 0.8309          | 0.8279        | 0.8372           |
| ViT-256  | 224               | 153        | 71       | 0.8292          | 0.8211        | 0.8328           |

Each decoder performs better in some regions. Under SAM-H, HoVer-Net achieves the higher Dice score on 134 patches, while CPP-Net performs better on 90. Under ViT-256, HoVer-Net performs better on 153 patches and CPP-Net on 71. CPP-Net is therefore not uniformly worse even though HoVer-Net has the higher mean Dice under both backbones.

The benefit of perfect selection nevertheless remains modest. For SAM-H, the oracle increases the mean Dice from the best fixed value of 0.8309 to 0.8372, an absolute gain of 0.0063. For ViT-256, the corresponding increase is from 0.8292 to 0.8328, an absolute gain of 0.0036.

Simple routing rules based on features derived from the local predictions were also evaluated, with the best tested rule for each backbone summarised in Table 5.7.

Table 5.7: Local MoNuSeg routing results.

| Backbone | Best feature/rule           | Fixed HoVer Dice | Routed Dice | Gain over fixed HoVer |
|----------|-----------------------------|------------------|-------------|-----------------------|
| SAM-H    | CPP-only fraction threshold | 0.8309           | 0.8318      | +0.0009               |
| ViT-256  | No improving rule found     | 0.8292           | 0.8292      | +0.0000               |

Under SAM-H, the best tested prediction-based routing rule improves mean Dice by only 0.0009 over the fixed HoVer-Net result. Under ViT-256, none of the tested rules improves on the fixed HoVer-Net baseline.

The analyses based on complete images and local patches therefore lead to the same general conclusion. Oracle selection shows that HoVer-Net and CPP-Net provide some complementary predictions, but the available gains are limited and the tested handcrafted features do not reliably identify which decoder should be used.

Several observations contribute to this result. First, the aggregate differences between HoVer-Net and CPP-Net on MoNuSeg are already small, so a routing rule must make sufficiently reliable selections to improve on the best fixed decoder. Second, the tested density-related features show weak relationships with decoder performance differences. Third, the qualitative examples and patch comparisons show that differences between the two decoders can occur within particular regions rather than following a simple global property of the image.

The routing experiments therefore do not provide evidence that the tested selection rules offer a reliable improvement over using a fixed decoder. The experiments show

that differences between HoVer-Net and CPP-Net are present, but the simple features examined in this study are not sufficient to exploit them consistently.

## 5.4 Summary of Validation Findings

The validation analyses provide different degrees of support for the backbone and decoder findings reported in Chapter 4.

For the backbone comparison, the three PanNuke experimental settings consistently favour SAM-H over ViT-256, although the limited number of observations means that the corresponding statistical analysis is treated descriptively. The MoNuSeg results provide stronger evidence from paired comparisons. SAM-H achieves higher bPQ, bDQ, and bSQ than ViT-256 under both decoder architectures, and the unadjusted  $p$ -values for all six corresponding Wilcoxon comparisons are below 0.05. Detection  $F_1$ , however, does not show the same statistical pattern despite being higher for SAM-H in the aggregate results.

The decoder comparison is less consistent. HoVer-Net achieves higher aggregate bPQ and mPQ than CPP-Net on PanNuke under both backbones, and the same direction is observed across the three predefined PanNuke settings. On MoNuSeg, however, the bPQ differences between HoVer-Net and CPP-Net do not reach the nominal significance threshold under either backbone. Differences instead appear in particular PQ components, with the bSQ difference being nominally significant under ViT-256 and the bDQ difference being nominally significant under SAM-H. The relative performance of the two decoders therefore depends more strongly on the dataset and metric being considered.

The qualitative comparison supports this interpretation. HoVer-Net and CPP-Net often produce overlapping predictions, but visible differences occur around faint nuclei, neighbouring cells, and crowded regions. The oracle analyses show that selecting between the decoders can provide some improvement, but the potential gains are generally small. Simple routing rules based on predicted nuclei abundance, predicted nuclear area, and local prediction features do not reproduce the oracle advantage consistently.

Taken together, these results provide stronger evidence for a consistent backbone effect than for a consistent decoder effect. SAM-H shows a more consistent advantage over ViT-256 across the evaluated settings, particularly for panoptic quality and its DQ and SQ components. In contrast, the relative performance of HoVer-Net and CPP-Net varies between PanNuke and MoNuSeg and between the metrics used for evaluation. The routing experiments further show that although the two decoders can make different and occasionally complementary predictions, the simple selection strategies evaluated in this study do not convert these differences into reliable performance improvements.

# Chapter 6

## Conclusions and Future Work

### 6.1 Conclusions

This dissertation investigated the effects of backbone and decoder choice in CellViT for nuclei instance segmentation in histology images. Four configurations combining ViT-256 or SAM-H with HoVer-Net or CPP-Net were evaluated. The study first reproduced the CellViT experiments on PanNuke and then applied the corresponding trained configurations to MoNuSeg without additional training or fine-tuning.

The PanNuke reproduction closely matched the published CellViT results where direct comparison was possible. This provided a reliable basis for comparing the different backbone and decoder combinations.

The backbone comparison showed a clear overall pattern. SAM-H performed better than ViT-256 under both decoder architectures on PanNuke, and the same general pattern was also observed on MoNuSeg. This suggests that backbone choice had a relatively consistent effect across the two datasets.

The decoder comparison was less consistent. HoVer-Net performed better on PanNuke for the main aggregate segmentation metrics, while the MoNuSeg results depended more on the metric being considered. Neither HoVer-Net nor CPP-Net was consistently better across all settings.

The qualitative and routing analyses also showed differences between the two decoders. HoVer-Net and CPP-Net produced different predictions in some images and local regions, especially around faint nuclei, neighbouring cells, and crowded areas. Oracle analysis showed that selecting between the two decoders could provide some improvement, but the simple routing rules tested in this study did not consistently outperform the best fixed decoder.

Overall, backbone choice showed a more consistent effect than decoder choice. SAM-H generally performed better than ViT-256, while the relative performance of HoVer-Net and CPP-Net changed between datasets and metrics.

The study also has some limitations. The MoNuSeg evaluation used only 14 test images and one fixed checkpoint per configuration, and the statistical analyses were based on limited samples. The routing analysis also used only simple prediction-derived features.

## 6.2 Future Work

Future work could evaluate the same backbone and decoder combinations on more histopathology datasets. This would help determine whether the backbone and decoder patterns observed on PanNuke and MoNuSeg also appear in other datasets.

The density-based routing experiments in this dissertation used simple prediction-derived features and threshold rules. Future work could extend this approach using a learned patch-level routing model in which local nuclear density or crowding is used to help select between HoVer-Net and CPP-Net.

Another direction is adaptive decoder fusion. Instead of selecting only one decoder, a lightweight gating module could combine the outputs of HoVer-Net and CPP-Net according to local image conditions. This may make better use of the differences between the two decoders.

# References

- [1] J. Chen, J. Mei, X. Li, Y. Lu, Q. Yu, Q. Wei, X. Luo, Y. Xie, E. Adeli, Y. Wang, M. P. Lungren, S. Zhang, L. Xing, L. Lu, A. Yuille, and Y. Zhou. Transunet: Rethinking the u-net architecture design for medical image segmentation through the lens of transformers. *Medical Image Analysis*, 97:103280, 2024.
- [2] R. J. Chen, C. Chen, Y. Li, T. Y. Chen, A. D. Trister, R. G. Krishnan, and F. Mahmood. Scaling vision transformers to gigapixel images via hierarchical self-supervised learning. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16123–16134, 2022.
- [3] S. Chen, C. Ding, M. Liu, J. Cheng, and D. Tao. Cpp-net: Context-aware polygon proposal network for nucleus segmentation. *IEEE Transactions on Image Processing*, 32:980–994, 2023.
- [4] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2021.
- [5] J. Gamper, N. A. Koohbanani, K. Benes, S. Graham, M. Jahanifar, S. A. Khurram, A. Azam, K. Hewitt, and N. Rajpoot. Pannuke dataset extension, insights and baselines, 2020.
- [6] S. Graham, Q. D. Vu, S. E. A. Raza, A. Azam, Y. W. Tsang, J. T. Kwak, and N. Rajpoot. Hover-net: Simultaneous segmentation and classification of nuclei in multi-tissue histology images. *Medical Image Analysis*, 58:101563, 2019.
- [7] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2980–2988, 2017.
- [8] F. Hörst, M. Rempe, L. Heine, C. Seibold, J. Keyl, G. Baldini, S. Ugurel, J. Siveke, B. Grünwald, J. Egger, and J. Kleesiek. Cellvit: Vision transformers for precise cell segmentation and classification. *Medical Image Analysis*, 94:103143, 2024.
- [9] A. Kirillov, K. He, R. Girshick, C. Rother, and P. Dollár. Panoptic segmentation. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9396–9405, 2019.
- [10] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollár, and R. Girshick. Segment anything.

- In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3992–4003, 2023.
- [11] N. Kumar, R. Verma, D. Anand, Y. Zhou, O. F. Onder, E. Tsougenis, H. Chen, P.-A. Heng, J. Li, Z. Hu, Y. Wang, N. A. Koohbanani, M. Jahanifar, N. Z. Tajeddin, A. Gooya, N. Rajpoot, X. Ren, S. Zhou, Q. Wang, D. Shen, C.-K. Yang, C.-H. Weng, W.-H. Yu, C.-Y. Yeh, S. Yang, S. Xu, P. H. Yeung, P. Sun, A. Mahbod, G. Schaefer, I. Ellinger, R. Ecker, O. Smedby, C. Wang, B. Chidester, T.-V. Ton, M.-T. Tran, J. Ma, M. N. Do, S. Graham, Q. D. Vu, J. T. Kwak, A. Gunda, R. Chunduri, C. Hu, X. Zhou, D. Lotfi, R. Safdari, A. Kascenas, A. O’Neil, D. Eschweiler, J. Stegmaier, Y. Cui, B. Yin, K. Chen, X. Tian, P. Gruening, E. Barth, E. Arbel, I. Remer, A. Ben-Dor, E. Sirazitdinova, M. Kohl, S. Braunewell, Y. Li, X. Xie, L. Shen, J. Ma, K. D. Bakshi, M. A. Khan, J. Choo, A. Colomer, V. Naranjo, L. Pei, K. M. Iftekharuddin, K. Roy, D. Bhattacharjee, A. Pedraza, M. G. Bueno, S. Devanathan, S. Radhakrishnan, P. Koduganty, Z. Wu, G. Cai, X. Liu, Y. Wang, and A. Sethi. A multi-organ nucleus segmentation challenge. *IEEE Transactions on Medical Imaging*, 39(5):1380–1391, 2020.
  - [12] N. Kumar, R. Verma, S. Sharma, S. Bhargava, A. Vahadane, and A. Sethi. A dataset and a technique for generalized nuclear segmentation for computational pathology. *IEEE Transactions on Medical Imaging*, 36(7):1550–1560, 2017.
  - [13] J. Ma, Y. He, F. Li, L. Han, C. You, and B. Wang. Segment anything in medical images. *Nature Communications*, 15(1):654, Jan 2024.
  - [14] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, Cham, 2015. Springer International Publishing.
  - [15] U. Schmidt, M. Weigert, C. Broaddus, and G. Myers. Cell detection with star-convex polygons. In A. F. Frangi, J. A. Schnabel, C. Davatzikos, C. Alberola-López, and G. Fichtinger, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2018*, pages 265–273, Cham, 2018. Springer International Publishing.
  - [16] Student. The probable error of a mean. *Biometrika*, 6(1):1–25, 1908.
  - [17] F. Wilcoxon. Individual comparisons by ranking methods. *Biometrics Bulletin*, 1(6):80–83, 1945.

# Appendices



# Appendix A

## External Materials

### A.1 CPP-Net Compatibility for MoNuSeg Inference

The original CellViT MoNuSeg inference pathway used in this study was structured around the output format of CellViT models with HoVer-Net decoders. To evaluate the CPP-Net configurations under the same MoNuSeg inference protocol, a small number of compatibility adaptations were introduced.

Support was first added for the CellViT256CPP and CellViTSAMCPP model classes so that CPP-Net checkpoints could be loaded through the existing inference pipeline.

Post-processing was then handled separately for the two decoders. For HoVer-Net models, the original instance reconstruction pathway was retained. For CPP-Net models, a sigmoid activation was applied to the predicted distance map logits before the distance map, refined radial predictions, and nucleus type predictions were passed to the CPP-Net instance reconstruction interface.

The same decoder distinction was maintained when processing overlapping patches. CPP-Net predictions required adaptations to the merging procedure so that it did not depend on HoVer-Net-specific output keys, while the existing HoVer-Net pathway remained unchanged.

As a compatibility check, the original and adapted inference pathways were compared using HoVer-Net configurations. The resulting bPQ and detection  $F_1$  values agreed to four decimal places, providing evidence that the adaptations affected only CPP-Net processing rather than altering the existing HoVer-Net evaluation pathway.