SUPERPATCHING FOR IMAGE ANALYSIS USING TRANSFORMERS AND SUPERPIXELS

By

Christopher Brannon McCutcheon

July, 2025

Director of Thesis: David Hart, PhD

Major Department: Computer Science

ABSTRACT

Transformers have revolutionized Computer Vision, offering robust performance across diverse tasks. However, their reliance on uniform pixel patching presents limitations, including computational inefficiency for larger images, suboptimal handling of local features, and an inability to process non-uniform patches. Addressing these constraints allows for new opportunities to expand their utility in demanding fields, such as medical imaging.

This work proposes a novel architecture combining Convolutional Neural Networks (CNNs) and Transformers to leverage superpixels, clusters of pixels with shared characteristics that capture local feature boundaries effectively. We propose an architecture that segments images into a collection of superpixels, vectorizes these superpixels using a CNN, and passes the resulting tokenized vector representations to a standard Transformer. By removing the uniformity constraint in patching, our approach aims to enhance Transformer performance on tasks requiring large-scale image analysis and fine-grained local feature understanding, potentially opening a way for broader Transformer applications in Computer Vision.

SUPERPATCHING FOR IMAGE ANALYSIS USING TRANSFORMERS AND SUPERPIXELS

A Thesis

Presented to The Faculty of the Department of Computer Science

East Carolina University

In Partial Fulfillment of the Requirements for the Degree

Master of Science in Data Science

By

Christopher Brannon McCutcheon

July, 2025

Director of Thesis: David Hart, PhD

Thesis Committee Members:

Nic Herndon, PhD

Qin Ding, PhD

Table of Contents

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

1.1 Background and Motivation

Transformers, initially designed for natural language processing, have gained traction in computer vision through the Vision Transformer (ViT) architecture. ViTs have demonstrated strong performance on image classification tasks due to their ability to model long-range dependencies through self-attention mechanisms [6]. Vision Transformers deconstruct images into fixed-size patches and process each patch as an input token within their attention mechanism, which yields favorable results to convolutional neural network (CNN) counterparts. These fixed-size patches, however, enforces a restriction on any inference image to be the same size as the training inputs. Additionally, these square patches are a training convenience and do not correlate with semantically meaningful regions of the image.

In recent years, the field of medical image analysis has also witnessed a growing interest in deep learning-based and ViT approaches, particularly for tasks involving whole slide images (WSIs) [10, 15]. WSIs are extremely high-resolution histopathological images used for diagnosis, disease grading, and research in oncology and pathology. These images can reach gigapixel scales, which poses significant computational and memory challenges for traditional CNNs and transformers. When applied directly to WSIs, these models struggle with spatial efficiency and scalability.

Superpixels offer a promising alternative to dense pixel-wise patch processing, opening avenues for solving both the fixed image size restriction and large-scale image memory issues.

A superpixel groups neighboring pixels into perceptually meaningful regions, effectively reducing the spatial resolution of the input without losing important semantic information. This can be particularly advantageous for WSIs, where large homogeneous regions (e.g., white background or uniform tissue) dominate the image.

This thesis proposes a novel architecture that combines the spatial efficiency of superpixels with the global modeling power of transformers. It begins with preprocessing images with a superpixel algorithm after which the image is fed as-is into a CNN which creates a feature map. That feature map is scattered to the superpixel map so that an aggregrate value of pixel representation is generated for each superpixel. These aggregated values are then fed into a transformer as input tokens and processed by their self-attention mechanism to produce classification results. A basic overview of this process is shown in Figure 1.1. By preprocessing images into superpixels and passing them through a CNN-Transformer pipeline, we aim to improve performance, interpretability, and computational efficiency.

1.2 Problem Statements

This thesis simultaneously addresses two main problem statements. First, traditionally trained ViT networks cannot handle variable-sized input images after training without first resizing the input images. Second, processing WSIs with transformer-based models requires breaking the image into thousands of fixed-size patches or tiles. This process is inefficient and loses spatial context when dealing with high resolution imagery. This additionally leads to memory usage issues and a reduction in value for positional encodings for large inputs.

We hypothesize that replacing fixed-size patches with superpixels allows for semantically meaningful, adaptive grouping of pixels that better capture object structure and increase model capability. Additionally, using CNNs to extract features from these superpixel regions can retain high-level representations, which the transformer can leverage in addition to its inherit attention mechanism.

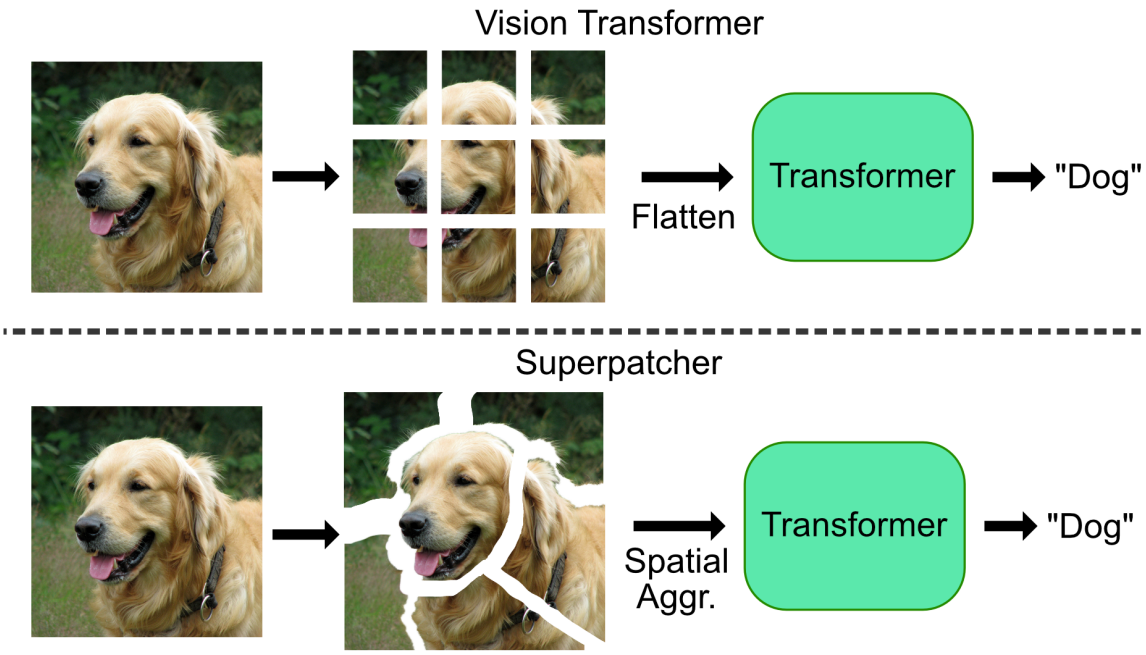

Figure 1.1: An overview of our approach. In a standard vision transformer (top), images are broken into square patches of a predetermined size. The patches are unlikely to correlate with semantically meaningful regions of the image. In our approach (bottom), we break patches into nonuniform regions using superpixels. The patch information is summarized through a spatial aggregation function before being fed into the transformer. This allows for transformer inputs that correspond to semantically meaningful regions in the input.

1.3 Research Objectives

The main objectives of this research are:

- To develop a hybrid model that combines superpixel segmentation, CNN-based feature extraction, and a transformer's attention architecture for image analysis.
- To integrate SLIC-based superpixel segmentation as a preprocessing step for adaptive spatial reduction.
- To evaluate the performance of the proposed Superpixel Transformer on commonplace datasets and histopathological datasets in terms of classification accuracy and efficiency.

1.4 Contributions

This thesis makes the following contributions:

- Proposes a novel architecture that leverages superpixel segmentation to enable efficient transformer-based image analysis.
- Demonstrates a pipeline that extracts and aggregates CNN features from isolated superpixel regions.
- Integrates positional encoding based on superpixel centroids for effective attention modeling.
- Validates the method on multiple datasets, including the Camelyon16 dataset for tumor identification and the ImageNet and OxfordPets datasets for generalization testing.

1.5 Thesis Structure

This thesis is structured as follows:

- **Chapter 2** provides an overview of related work in CNNs, transformers, superpixels, and WSI analysis.
- **Chapter 3** details the proposed methodology, including preprocessing, feature extraction, and model architecture.
- **Chapter 4** presents the experiments, results, and evaluation metrics.
- **Chapter 5** discusses findings, limitations, and future work.

Chapter 2

Related Works

2.1 Vision Transformers in Computer Vision

Vision Transformers (ViTs), introduced by Dosovitskiy et al. [6], marked a significant shift in computer vision by removing convolutional inductive biases and using self-attention to model image-wide dependencies. ViTs divide an image into a sequence of non-overlapping patches, flatten them, and encode them positionally before feeding them into a transformer encoder. While effective on moderate-size images such as those in ImageNet, ViTs scale poorly with extremely large images like WSIs due to computational and memory constraints. This has motivated research into improving the spatial efficiency and scalability of ViTs.

Several adaptations have emerged to improve ViT scalability: Swin Transformers [12] introduce hierarchical tokenization through shifted windows; PiT [7] and CvT [17] reintroduce convolutional priors to reduce redundancy. However, these approaches still depend on dense tiling and fixed-size patches, which are not optimal for high-resolution or irregular semantic structures like those found in WSIs.

2.2 Superpixels in Deep Learning

Superpixels are oversegmented regions that group pixels with similar appearance and spatial proximity. The Simple Linear Iterative Clustering (SLIC) algorithm by Achanta et al. [1] is among the most widely used methods for generating superpixels due to its speed and effectiveness in adhering to image boundaries.

Unlike grid-based patches, superpixels adapt to image content, making them ideal for heterogeneous data like histology slides. In medical imaging, superpixels have been employed for tissue segmentation, cell counting, and region-of-interest identification. Recent work has shown that superpixel-based representations can improve the robustness and interpretability of models by filtering out redundant background and isolating meaningful tissue structures.

Several recent works have explored the integration of superpixels into transformer-based architectures. SPFormer [12] enhances Vision Transformers by replacing fixed-size patch tokens with superpixels to enable more semantically coherent attention. Similarly, Superpixel Transformers [20] align attention mechanisms with superpixel boundaries to improve segmentation efficiency, while SuperFormer [14] guides transformer attention for salient object detection using superpixels. Superpixel Tokenization for ViTs [9] leverages perceptually meaningful superpixels to improve token coherence and model robustness. Superpixel Positional Encoding [2] introduces a lightweight encoding scheme to enhance semantic segmentation by capturing spatial superpixel relationships. In the context of medical and remote sensing domains, SuperCL [18] employs contrastive learning with superpixel-guided pseudo-masks for medical image segmentation, and Self-Supervision with Superpixels [13] generates pseudo-labels using superpixels in few-shot learning settings. Superpixel Transformers for Remote Sensing [5] adapts this concept for high-resolution satellite image classification. While these methods emphasize superpixels in segmentation and attention modeling, they adhere to the patching mechanism of transformers. Our approach uniquely focuses on replacing the patching mechanism by transforming superpixel-level CNN features into structured tokens for classification, emphasizing feature aggregation, position-aware encoding, and model adaptability across both standard and whole-slide image domains.

2.3 CNN-Transformer Hybrids

CNNs are well known for their ability to extract local texture and edge features, while transformers excel in modeling global relationships. Hybrid models, such as CoAtNet [4]

and BoTNet [16], leverage the strengths of both by using CNN layers for low-level feature extraction and transformers for high-level context modeling. These models perform well across vision benchmarks, particularly when spatial locality and global dependency need to be balanced.

In histopathology, similar combinations have been explored [11]. Patch-based approaches typically extract CNN embeddings for each tile and use transformers to model spatial relations. However, these methods treat all patches equally, regardless of semantic importance, leading to inefficient attention and limited interpretability.

2.4 Whole Slide Image Analysis

WSIs pose unique challenges due to their size and variability. Traditional methods use sliding windows or tile the WSI into fixed-size patches, which are then processed independently. Multiple instance learning (MIL) frameworks aggregate patch-level predictions to infer slide-level labels, as seen in models like CLAM [11] and ABMIL [8]. While effective, these methods rely on dense sampling and often fail to leverage structural priors present in the image.

Recent work has attempted to address this by introducing spatial context through graph-based models [19] or attention mechanisms [3], enabling better modeling of inter-patch relationships in whole slide image analysis. However, these still operate on arbitrary patch grids. By contrast, superpixel-guided approaches offer a more structured and semantically meaningful partitioning of tissue, potentially improving both efficiency and interpretability.

2.5 Summary

The existing literature demonstrates a growing interest in combining CNNs and transformers for complex vision tasks. While Vision Transformers offer powerful modeling capabilities, their generality and application to WSIs remains limited by spatial restrictions and scale. Superpixels have shown promise in capturing semantic structure and improving model focus.

Chapter 3

Methodology

3.1 Overview

This chapter outlines the proposed methodology for superpixel-based transformer modeling in standard image analysis (Superpatcher) and WSI analysis (WSI-SuperPatcher). The core idea in both architectures is to improve image analysis by leveraging superpixel boundaries and features in conjunction with the self-attention mechanism of transformers. We enhance semantic coherence by segmenting images into superpixels and encoding them into meaningful representations using a CNN-Transformer pipeline. Although the Superpatcher and the WSI-SuperPatcher share the same general concept, some modifications were necessary to accommodate the unique challenges of WSI data. The first main section in our methodology will outline the architecture of the Superpatcher (Oxford Pets, ImageNet, High Res). The second section will detail the changes used in the WSI-Superpatcher without re-explaining the architectural components used in the default model. This chapter details the architecture, embeddings, training strategy, and evaluation protocol for the Superpatcher variant, which also serve as a foundation for later adaptations to WSI-based tasks.

3.2 Superpatcher

3.2.1 Dataset Preparation

The datasets used for the Superpatcher model include Oxford Pets, the 2012 ILSVRC subset of ImageNet, and a custom High-Resolution Cat-Dog-Bird (BCD) dataset. Each dataset was selected to introduce distinct challenges and test the generalizability of the proposed model across varying conditions.

Oxford Pets is a relatively small-scale dataset comprising 37 fine-grained classes, which encourages the model to capture subtle intra-class and inter-class visual differences. The ImageNet subset provides a standard benchmark for image classification and serves as a baseline for comparison with conventional convolutional and transformer-based vision models. The High-Resolution BCD dataset presents a more computationally demanding task due to the larger size and fine detail of its images. Although it was originally intended to be processed at full resolution, practical constraints require the images to be resized prior to model input.

All datasets underwent preprocessing that included resizing to fixed resolutions (224×224, 512×512, or 1024×1024), depending on experimental configuration. This resizing step is necessary to ensure alignment between CNN feature maps and superpixel segmentation masks, which is essential for effective feature aggregation. Image annotations were obtained either directly from provided metadata (e.g., directory structure) or inferred from image filenames, depending on the dataset.

3.2.2 Superpixel Segmentation

All models employed the SLIC (Simple Linear Iterative Clustering) algorithm to segment input images into approximately N superpixels. SLIC operates in a five-dimensional space—comprising the CIELAB color components and pixel spatial coordinates—and clusters pixels based on both color similarity and spatial proximity [1]. This ensures that resulting superpixels are

both visually coherent and spatially compact.

It is important to note that the SLIC parameter _n_segments specifies an approximate rather than an exact number of output superpixels. The final number may vary slightly depending on image structure and boundary complexity. Each pixel in the image is assigned a superpixel label ranging from 0 to $N - 1$, forming a segmentation map that serves as the basis for downstream feature aggregation. This is shown in Figure 3.1.

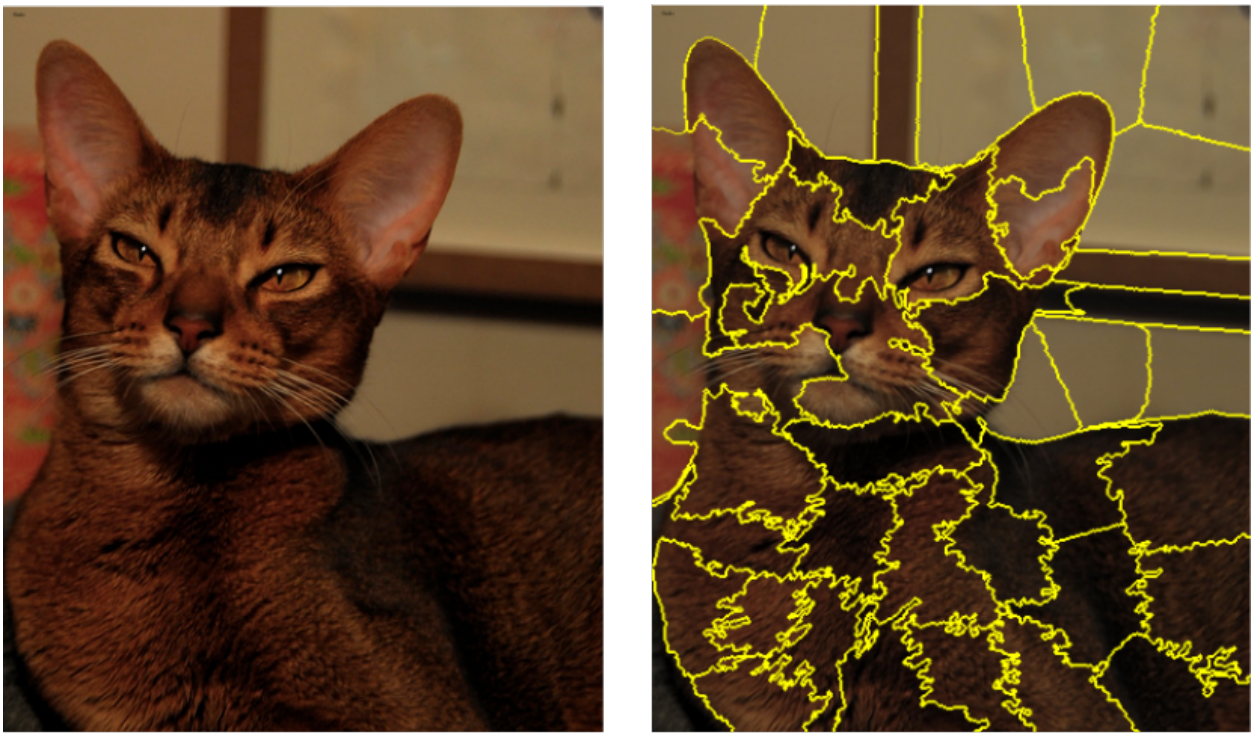

Figure 3.1: Example Image from the Oxford Pets Dataset and its accompanying SLIC superpixel map boundaries with approximately 50 segments.

3.2.3 Feature Extraction with CNN

A convolutional neural network (CNN) based on the ResNet architecture (ResNet18, ResNet50, or ResNet101) is employed to extract high-level feature representations from each input image. The ResNet's standard pretrained weights from ImageNet are maintained. After segmentation with the SLIC algorithm, the original image is passed through the CNN to

generate a deep feature map. Because the spatial resolution of the CNN output is reduced during pooling and strided convolutions, the resulting feature map is subsequently upsampled using bilinear interpolation to match the resolution of the superpixel map. This step ensures that each pixel-level feature vector can be accurately associated with its corresponding superpixel label in the subsequent aggregation stage.

3.2.4 Superpixel Embedding and Aggregation

Once the upsampled feature map is obtained, it is then mapped to the superpixel map using a scatter function. This operation aggregates pixel-level CNN features into superpixel-level embeddings by assigning each pixel's feature vector to its corresponding superpixel region. Specifically, the `torch.scatter` function from the PyTorch library is used to index into a tensor representing the superpixel embeddings. The embedding features are averaged over all pixels belonging to the same superpixel.

Formally, let $F \in R^{C \times H \times W}$ denote the CNN feature map and $S \in N^{H \times W}$ denote the superpixel map, where each pixel location (i, j) has an associated superpixel label $S_{i,j} \in \{0, 1, \ldots, N-1\}$. We flatten both tensors and use `scatter` to accumulate features from F into a tensor $E \in R^{N \times C}$, where each row corresponds to the aggregated embedding of a superpixel. This results in a compact and semantically meaningful representation of the image that can be passed to the transformer for further processing.

To define the scatter operation more precisely, let $\texttt{scatter}(F, S) \rightarrow E$ be an aggregation function that groups feature vectors in F based on their corresponding superpixel labels in S. Specifically, for each superpixel label $n \in \{0, 1, \ldots, N-1\}$, the embedding $E_n \in R^C$ is computed as:

$$E_n = \frac{1}{|P_n|} \sum_{(i,j) \in P_n} F_{:,i,j}, \tag{3.1}$$

where $P_n = \{(i, j) \mid S_{i,j} = n\}$ is the set of pixel positions belonging to superpixel n, and $F_{:,i,j} \in R^C$ is the feature vector at spatial location (i, j).

3.2.5 Additional Embeddings

To illustrate the unique contribution of superpixels to model performance, we experimented with the inclusion of auxiliary embeddings that could enhance the semantic or geometric understanding of each superpixel. Specifically, we introduced *Positional Encodings* and *Covariance Encodings* as additional input features to the transformer. These were tested independently and in combination to evaluate their effect on downstream performance.

Positional Encodings

Positional encodings are computed from the centroid coordinates of each superpixel. For each superpixel, we calculate the average row and column of its constituent pixels to obtain a 2D centroid coordinate. If a superpixel contains pixels at locations $(r_1, c_1), (r_2, c_2), \ldots, (r_N, c_N)$, the centroid $(\bar{r}, \bar{c})$ is:

$$\bar{r} = \frac{1}{N} \sum_{i=1}^{N} r_i, \quad \bar{c} = \frac{1}{N} \sum_{i=1}^{N} c_i$$

These centroid coordinates are then normalized by the image height H and width W:

$$\left(\frac{\bar{r}}{H}, \frac{\bar{c}}{W} \right)$$

This normalized 2D position is projected into the transformer embedding space using a linear layer and added to the corresponding superpixel token. These embeddings were then added to the corresponding superpixel feature vectors prior to transformer input, enabling the model to incorporate spatial awareness.

Covariance Encodings

Covariance encodings were introduced to provide shape-based information for each superpixel region. For every superpixel, we calculated the unbiased covariance matrix of the pixel coordinates relative to the superpixel centroid. Specifically, we measure how far the pixels

in each superpixel are spread out along rows and columns:

$$\sigma_r = \text{std}(r_1, r_2, \ldots, r_N), \quad \sigma_c = \text{std}(c_1, c_2, \ldots, c_N)$$

These two values form a compact geometric descriptor of the superpixel. Like the positional encoding, they are passed through a learnable projection and used to augment the transformer input features. A gated tanh activation was applied to regulate the magnitude of these embeddings before combining them with the rest of the feature representation.

3.2.6 Transformer Encoding

The transformer encoder receives as input a sequence of superpixel-level feature vectors, which are always included and serve as the primary representation of image content. Depending on the model configuration, two optional auxiliary encodings may be added to enrich these embeddings:

- **Positional encodings**, derived from the centroid coordinates of each superpixel and normalized relative to the image dimensions.
- **Covariance encodings**, capturing the spatial dispersion and orientation of pixels within each superpixel.

When used, each auxiliary embedding is projected into the same dimensional space as the superpixel features and added element-wise to the corresponding token. This design enables the transformer to integrate spatial location and shape information alongside the learned visual representations.

The transformer encoder consists of L layers, each comprising a multi-head self-attention mechanism, layer normalization, and a position-wise feed-forward network. The self-attention module allows each superpixel token to attend to all others in the sequence, facilitating contextual reasoning across the image. This is particularly valuable for capturing long-range dependencies and interactions between spatially distant but semantically-related regions.

3.2.7 Output and Prediction Head

Since the task is classification (e.g., cat vs. dog), the transformer architecture uses a `[CLS]` token prepended to the input sequence. After processing through the transformer, the output corresponding to this `[CLS]` token is passed through a fully connected layer to produce the final class logits.

3.2.8 Loss, Optimization, and Evaluation

The model is trained end-to-end using the Cross-Entropy Loss function, which is standard for multi-class classification problems. Optimization is performed using Adam or AdamW, optionally with learning rate warmup and cosine decay scheduling to stabilize training. Model performance is evaluated using standard classification accuracy.

3.2.9 Pipeline Summary

Combining all the steps outlined in this section, a standard pipeline can be created that takes images as input and predicts classification scores as the output. The pipeline pseudocode is presented in Algorithm 1 and visualized in Figure 3.2.

Algorithm 1: Superpatcher Forward Pass

Input: Image I

Output: Prediction $\hat{y}$

Transform and Resize I;

Pass I through SLIC to obtain superpixel map S;

Pass I through CNN to obtain feature map F;

Use `scatter_reduce` to aggregate pixel features into superpixel-level vectors $\{z_i\}$ based on S;

For each superpixel S_i:

    Compute centroid position p_i;

    Normalize p_i by image dimensions;

    Compute covariance of pixel coordinates relative to p_i;

Construct embedding matrix $Z = [z_1, \ldots, z_n]$;

Project $\{p_i\}$ to positional encodings and add to Z;

Project covariance encodings and add to Z with gated modulation;

Pass Z through Transformer Encoder to get output H;

Compute prediction $\hat{y}$ from H via classification head;

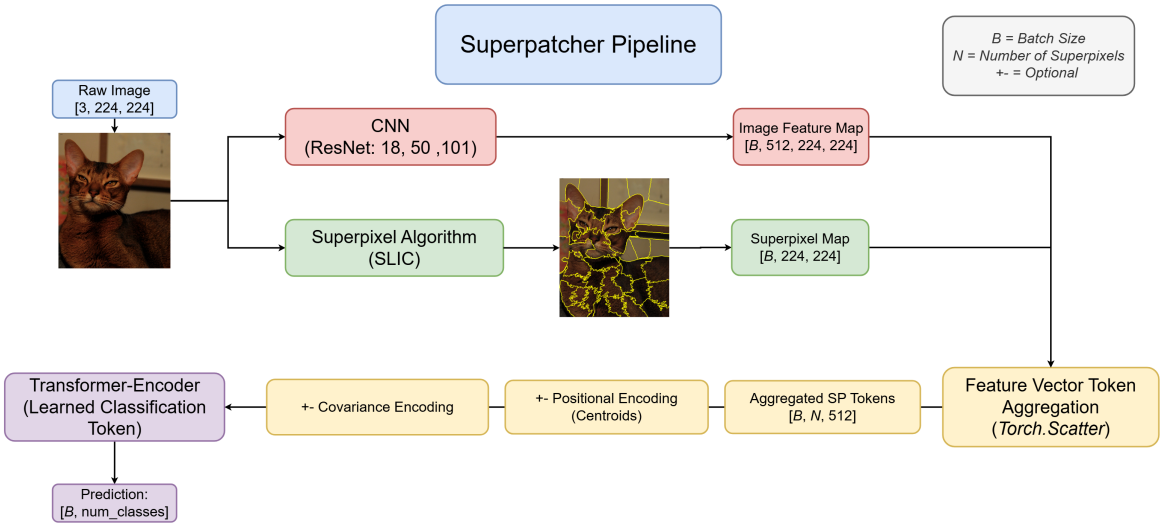

Figure 3.2: Overview of the Superpatcher pipeline. Image are segmented into superpixels and fed through a CNN to extract features, which are then aggregated into feature vectors and fed into a transformer encoder.

3.3 WSI-Superpatching

3.3.1 Dataset Preparation

The WSI-Superpatcher variant is designed to process gigapixel-scale whole slide images (WSIs), which pose unique computational challenges due to their size and sparsity. To manage this, we utilize the OpenSlide library, which provides efficient tools for reading .tif format WSIs and for accessing image content at multiple resolution levels. Specifically, OpenSlide allows us to generate thumbnails and extract downsampled versions of each slide across predefined levels (0, ..., N), reducing memory load during preprocessing.

Due to hardware constraints—particularly limited VRAM on the GPU—WSIs could not be processed at their full native resolution. Instead, they were downscaled to a resolution level that retained sufficient tissue detail for analysis while remaining computationally tractable.

3.3.2 Cropping and Background Removal

Most pixels in a WSI correspond to non-informative background, typically appearing as nearly pure white. These regions not only lack diagnostic value but also introduce noise and inflate memory requirements. To mitigate this, we implemented a tissue detection procedure based on Otsu thresholding.

First, the downsampled WSI is converted to grayscale. Otsu's method is then applied to identify an optimal intensity threshold that separates background from tissue. Pixels below this threshold are classified as foreground. Non-informative tiles (i.e., patches consisting predominantly of background) are excluded from further processing. A visualization of an example superpixel map is presented in Figure 3.3. This filtering step ensures that the computational budget is focused on biologically relevant regions.

3.3.3 Superpixel Segmentation

Superpixel segmentation in the WSI pipeline is performed identically to the Superpatcher using the SLIC algorithm. Each foreground patch is segmented into superpixels that capture locally coherent structures based on color and spatial proximity.

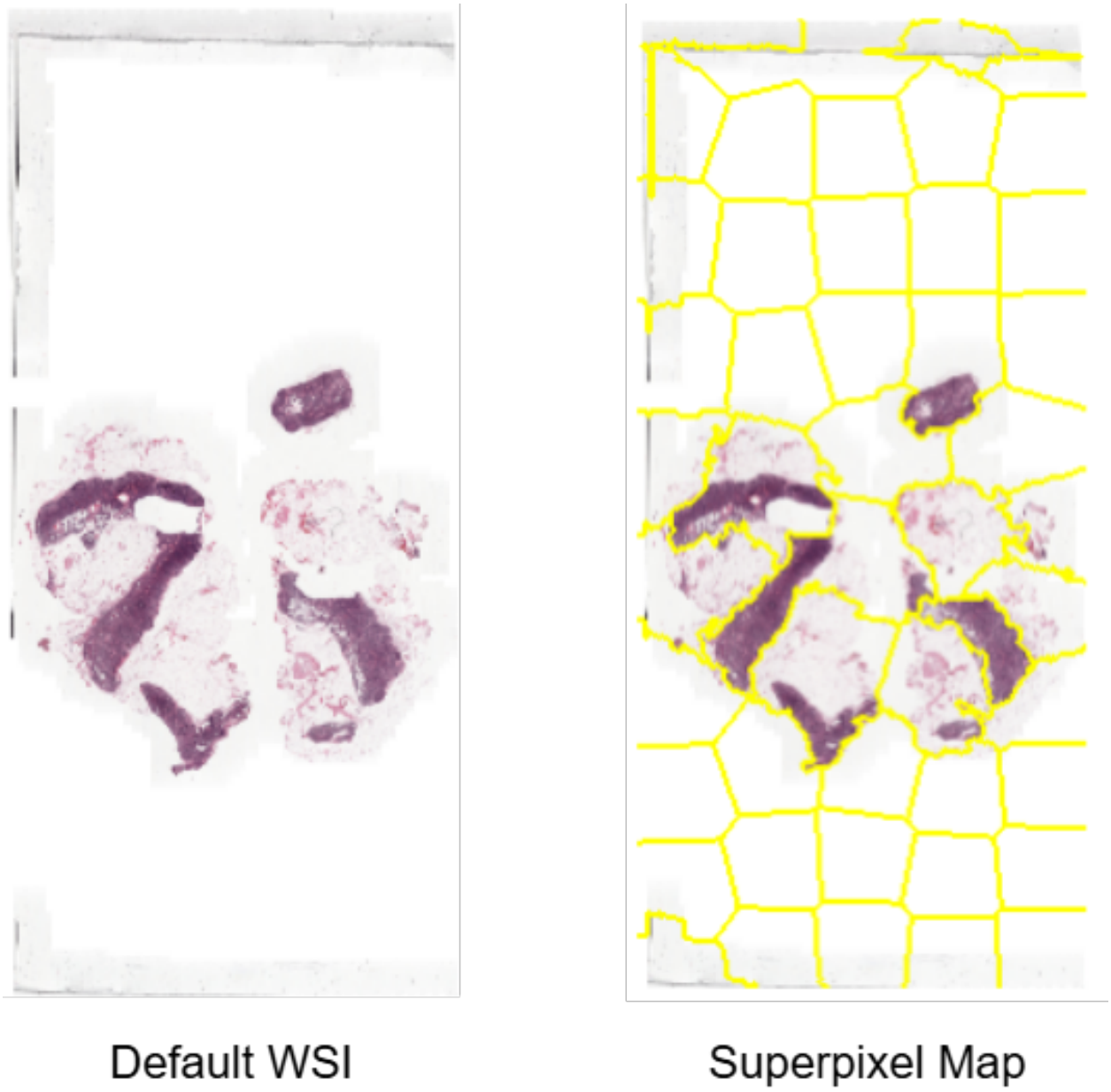

Figure 3.3: Example Image from the Camelyon-16 Dataset and its accompanying SLIC superpixel map boundaries with approximatley 50 segments.

3.3.4 Patch Extraction

To localize processing to tissue-relevant superpixel regions, we utilize the regionprops function from the skimage.measure module. For each superpixel, regionprops extracts its bounding box, defined by the minimum and maximum row and column indices of the pixel coor-

dinates belonging to that superpixel. The rectangular region specified by this bounding box is treated as a patch.

Within each bounding box, a binary mask is applied to isolate the true shape of the superpixel from surrounding tissue. This masked patch retains the superpixel's boundary fidelity while excluding adjacent content, thereby preserving structural coherence. This cropped and masked superpixel region is then passed individually through the CNN.

3.3.5 Feature Extraction with CNN

Each masked patch is forwarded through a shared CNN, consistent with the Superpatcher architecture. The CNN processes the patch and outputs a feature map, which is subsequently reduced to a fixed-size token using adaptive average pooling (e.g., to 8×8). These pooled tokens are stored in a list corresponding to their superpixel index and ultimately constitute the input sequence for the transformer.

3.3.6 Superpixel Embedding and Aggregation

Superpixel embeddings for the WSI pipeline follow the same structure as the Superpatcher. Additional positional and covariance embeddings—when used—are computed per superpixel region and integrated as described previously.

3.3.7 Transformer Encoding, Output Head, Loss, and Evaluation

The transformer encoder, output and prediction head, loss function, and evaluation metrics are unchanged from the Superpatcher configuration. However, comparative analysis to baseline CNN and ViT models was not performed for the WSI pipeline, as these models cannot feasibly process WSIs at high resolution under our hardware constraints.

3.3.8 Pipeline Summary

Combining all the steps outlined in this section, a WSI pipeline can be created that takes images as input and predicts classification scores as the output. This pipeline pseudocode is presented in Algorithm 2 and visualized in Figure 3.4.

Algorithm 2: WSI-Superpatcher Transformer Forward Pass

Input: Whole Slide Image W

Output: Prediction $\hat{y}$

Downsample W to manageable resolution I using OpenSlide;

Convert I to grayscale and apply Otsu thresholding;

Crop I to foreground tissue regions based on threshold mask;

Segment I into superpixels $S_1, \ldots, S_n$ using SLIC;

foreach *superpixel* S_i **do**

- Extract bounding box B_i using `regionprops`;
- Crop region P_i from I using B_i;
- Generate binary mask M_i for superpixel S_i within B_i;
- Apply M_i to P_i to isolate true superpixel shape;
- Pass masked patch $P_i \odot M_i$ through CNN to get feature map;
- Apply adaptive pooling to obtain fixed-size vector z_i;
- Compute and store centroid position p_i of S_i;

Construct embedding matrix $Z = [z_1, \ldots, z_n]$;

Add positional encodings from $\{p_i\}$;

Pass Z through Transformer Encoder;

Use output of [CLS] token or pooling to get final representation h;

Predict $\hat{y} = f(h)$ via classification head;

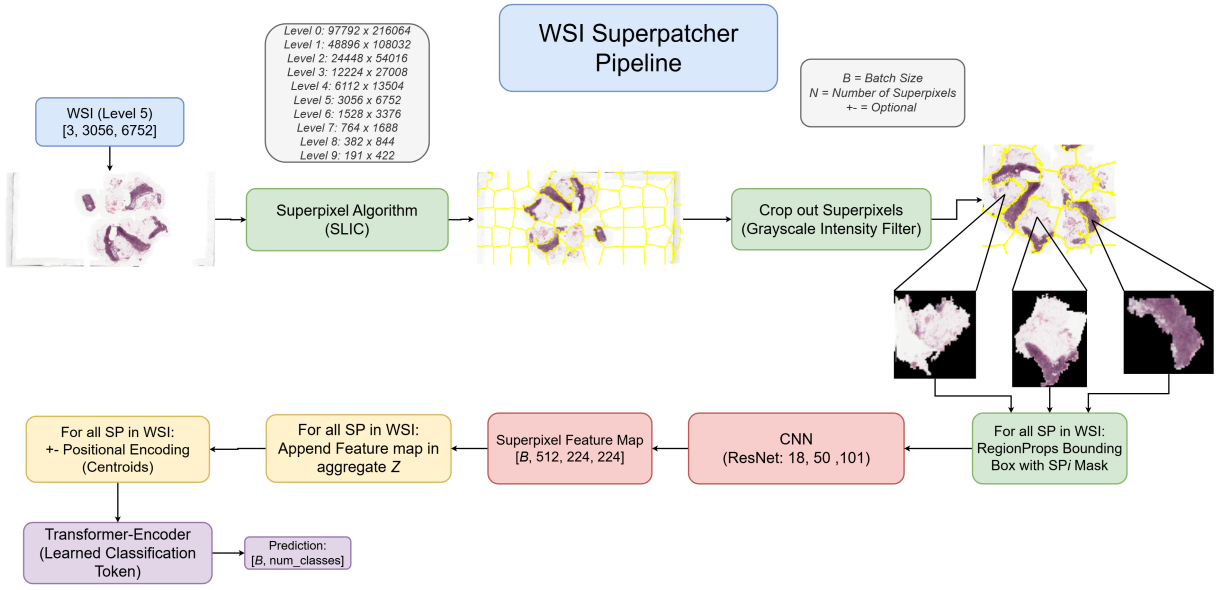

Figure 3.4: Overview of the Superpixel Transformer Pipeline. WSIs are segmented into superpixels, masked regions are passed through a CNN to extract features, which are then fed into a transformer encoder.

3.4 Hyperparameter Settings

The following values are the hyperparameter options used for both the Superpatcher and the WSI Super-patcher during training.

Parameter	Value
CNN backbone	ResNet18 / ResNet50 / ResNet101
Transformer layers	3
Embedding dim	512
Superpixels/tile	50 / 100 / 150
Batch size	16 / 8 / 4 / 2 / 1
Learning rate	1e-4 with cosine decay / 1e-3 with cosine decay
Optimizer	Adam / AdamW
Loss function	CrossEntropy
Epochs	50/ 30 / 20 / 10
Scheduler	Cosine annealing with warmup

3.5 Summary of Both Methods

This work proposes a superpixel-based transformer architecture tailored for both conventional image classification and whole slide image (WSI) analysis. The central idea is to leverage the structural coherence of superpixels to guide feature aggregation and improve transformer-based modeling. Two variants are introduced: the **Superpatcher**, which processes datasets such as Oxford Pets, ImageNet, and a custom high-resolution BCD dataset; and the **WSI-Superpatcher**, designed for high-resolution histopathology slides.

In the Superpatcher pipeline, input images are segmented using the SLIC algorithm to generate superpixels. A ResNet-based CNN extracts feature maps, which are then upsampled and aggregated into superpixel-level embeddings using a scatter-reduce operation. Each superpixel token may be enriched with auxiliary embeddings:

- **Positional Encodings**, computed from the normalized centroid coordinates of each superpixel.
- **Covariance Encodings**, computed from the standard deviation of pixel positions within the superpixel.

These enriched tokens are passed through a transformer encoder consisting of three layers of multi-head self-attention, layer normalization, and feed-forward networks. A learned `[CLS]` token is used for classification, with predictions obtained via a fully connected layer.

For WSIs, the WSI-Superpatcher introduces adaptations to handle large image sizes and sparse content. OpenSlide is used to downsample the images and extract relevant tissue regions. Background removal is performed via Otsu thresholding. Each foreground patch is segmented, masked, and passed through a shared CNN. The resulting tokens are then pooled and passed to the same transformer encoder used in the standard pipeline.

Both variants are trained using cross-entropy loss, optimized with Adam or AdamW, and evaluated using accuracy and loss metrics. This unified pipeline yields promising results for image classification.

Chapter 4

Results and Discussion

This chapter presents the classification performance of the proposed Superpatcher models, benchmarked against standard baseline architectures. All models employed CNN backbones pre-trained on ImageNet at a resolution of 224×224. Experiments were conducted on four datasets: Oxford Pets, ImageNet subsets, a custom HighRes dataset, and the Camelyon-16 histopathology dataset. Reported results reflect the highest validation accuracy attained during training for each configuration.

4.1 Baseline Model Performance

The baseline models include ResNet-18, ResNet-50, ResNet-101, and a standard Vision Transformer (ViT) using 16×16 input patch tokens. All models were initialized with ImageNet-pretrained weights and evaluated across the Oxford Pets and HighRes datasets at multiple input resolutions.

For the Oxford Pets and HighRes datasets, 30 epochs of fine-tuning were applied. Notably, the 16×16 ViT model is constrained to fixed-size inputs of 224×224 due to architectural limitations and is therefore excluded from evaluations at higher resolutions.

The performance of the baseline models is given in Table 4.1 and charts visualizing the performance at each scale are presented in Figures 4.1, 4.2, and 4.3.

Table 4.1: Table outlining the **validation accuracy of the baseline non superpixel-based models on the OxfordPets and the High Res** dataset at various scales.

Scale	Model	OxfordPets	HighRes
224x224	ResNet18	0.79	0.92
	ResNet50	0.67	0.91
	ResNet101	0.60	0.87
	16x16 ViT	0.14	0.92
512x512	ResNet18	0.648	**0.95**
	ResNet50	0.48	0.85
	ResNet101	0.44	0.86
	16x16 ViT	-	-
1024x1024	ResNet18	0.38	0.92
	ResNet50	0.08	0.82
	ResNet101	0.03	0.86
	16x16 ViT	-	-

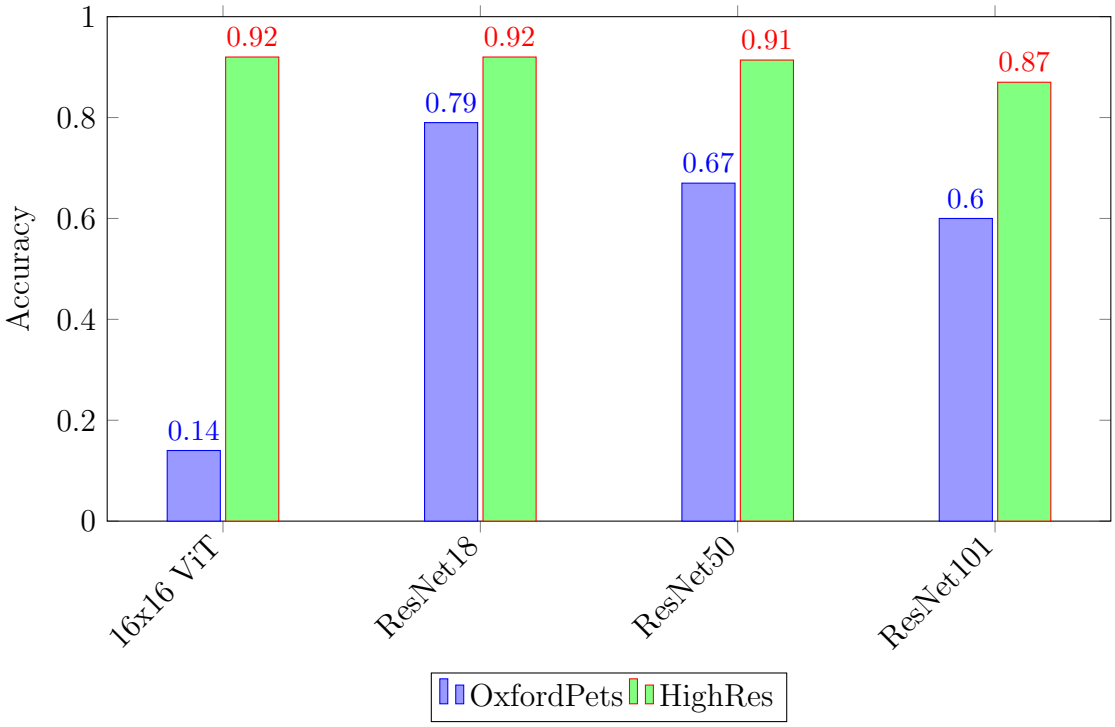

Figure 4.1: Bar char outlining **performance of baseline models on OxfordPets and HighRes at 224x224**.

At 224×224 resolution, the ResNet variants performed in line with expectations, achieving reasonable generalization on the Oxford Pets dataset after fine-tuning. In contrast, the ViT model failed to converge meaningfully, which aligns with prior findings that Vision Transformers typically require large-scale datasets to generalize effectively [6]. CNNs, by comparison, benefit from strong spatial inductive biases that allow them to perform well even on relatively small datasets such as Oxford Pets (7,400 samples).

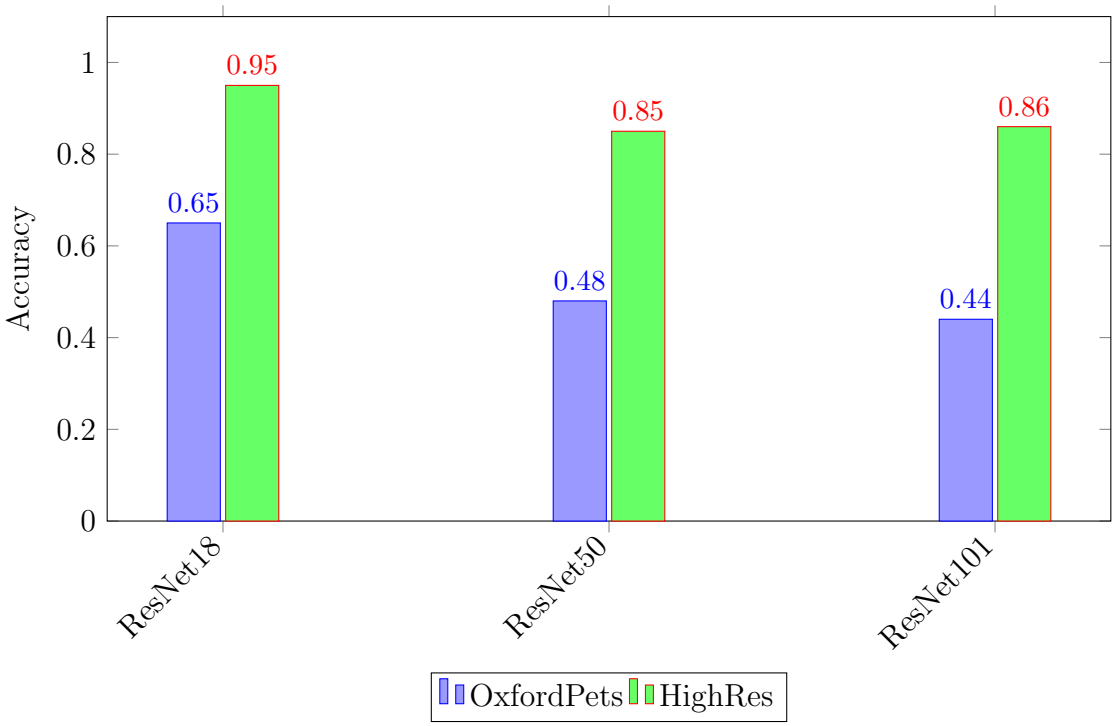

Figure 4.2: Bar char outlining **performance of baseline models on OxfordPets and HighRes at 512x512**.

As the input resolution increases to 512x512, a noticeable decline in classification accuracy is observed across all baseline models, including the ResNet variants. This degradation in performance is likely attributable to the inability of standard convolution kernels to generalize across spatial scales. The higher pixel count may hinder the model's ability to isolate and attend to the most salient regions of the image, thereby diluting its representational focus.

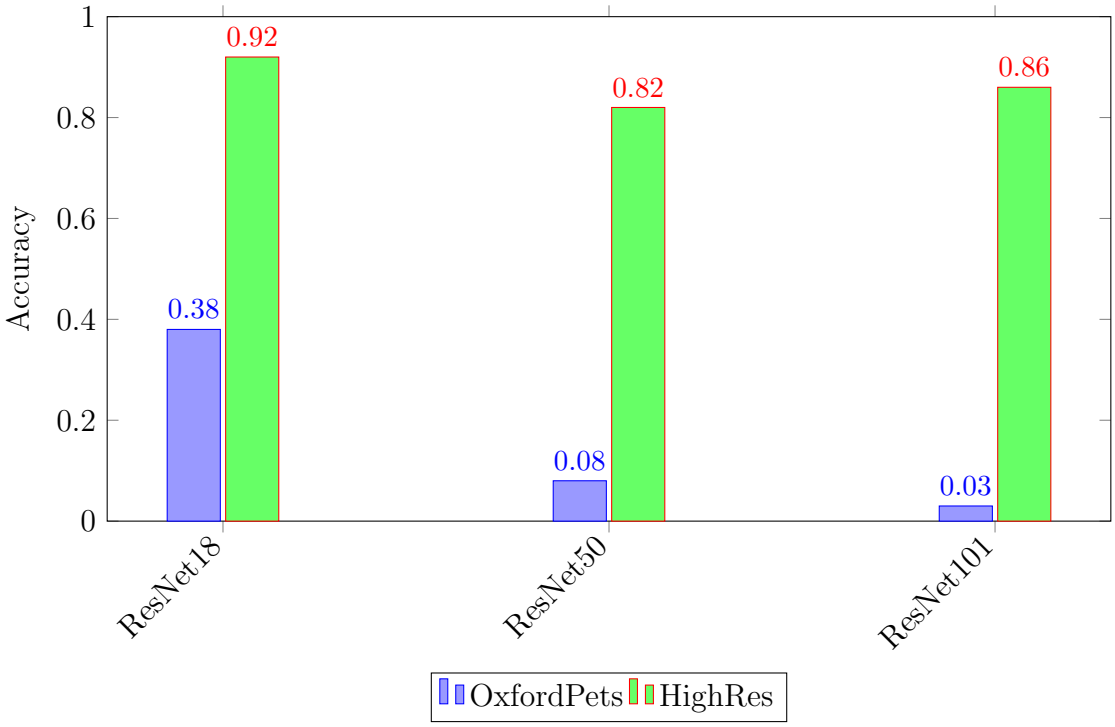

Figure 4.3: Bar char outlining **performance of baseline models on OxfordPets and HighRes at 1024x1024**.

As with increasing the input resolution to 512×512, a decline in classification performance was observed for most CNNs at 1024x1024, but the degradation becomes more pronounced, particularly in Oxford Pets. High Res performance differs less, likely due to the simplicity of the dataset classes. These results suggest that while CNNs are capable of handling larger inputs, the increased pixel count introduces redundant or noisy information, which can hinder learning if not appropriately managed. The models likely struggled to focus on semantically relevant regions within larger images, leading to diminished accuracy.

4.2 Superpatcher Performance

We next evaluated the proposed Superpatcher models using ResNet-18, ResNet-50, and ResNet-101 as CNN backbones, without any positional or covariance encodings. These results are presented in Table 4.2. Across all tested resolutions, the Superpatcher outperformed its baseline counterparts, demonstrating that superpixel-based tokenization can serve as a

highly effective alternative to uniform patch-based approaches.

Table 4.2: Table outlining the **validation accuracy of the superpixel-based models on the OxfordPets and the High Res** dataset at various scales.

Scale	Superpatcher Model	OxfordPets	HighRes
224x224	ResNet18-backbone	0.90	0.97
	ResNet50-backbone	0.93	0.98
	ResNet101-backbone	0.93	0.98
512x512	ResNet18-backbone	0.93	**0.99**
	ResNet50-backbone	0.95	**0.99**
	ResNet101-backbone	0.95	**0.99**
1024x1024	ResNet18-backbone	0.88	0.95
	ResNet50-backbone	0.89	0.98
	ResNet101-backbone	0.89	0.97

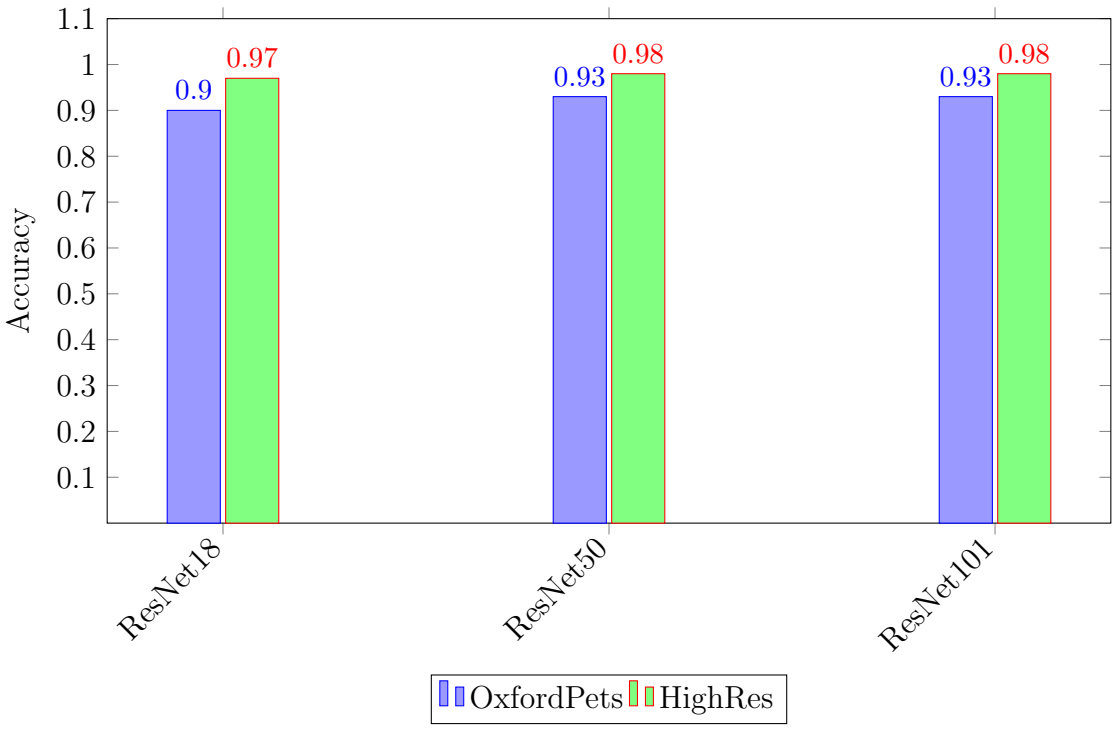

Figure 4.4: Bar chart indicating **performance of Superpatcher models on OxfordPets and HighRes** at 244x244.

At 224×224 resolution, the Superpatcher models achieved significantly higher accuracy, particularly on the Oxford Pets dataset. These results suggest that superpixel-based aggregation provides a strong inductive bias for semantic grouping, enabling the transformer to better capture spatially coherent features. Unlike traditional patching, which requires the model to infer spatial relationships between arbitrary image regions, superpixels introduce meaningful region-level structure from the outset, effectively easing the burden on the self-attention mechanism. Traditional Transformers must infer inter-patch relationships from scratch, but the Superpatcher provides region-level information up front, potentially reducing the learning burden on attention layers.

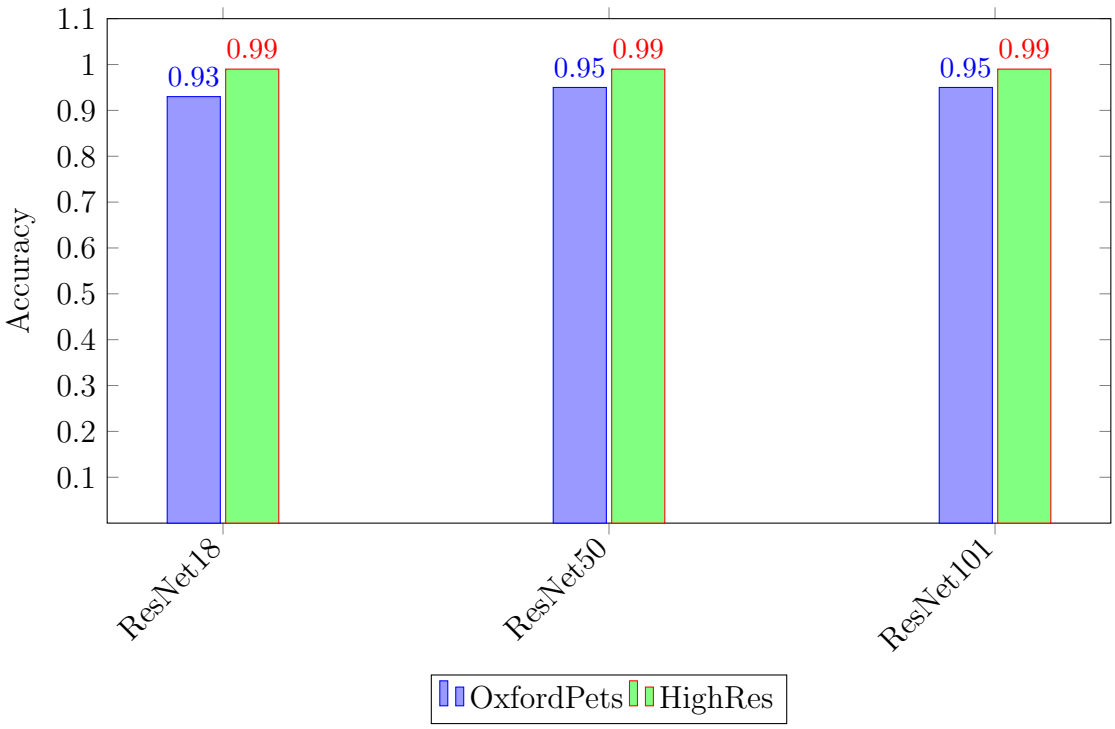

Figure 4.5: Bar chart indicating **performance of Superpatcher models on OxfordPets and HighRes** at 512x512.

Moreover, performance improved further as input resolution increased to 512×512. The added resolution introduces more complex superpixels and, in turn, more fine-grained feature tokens for the transformer to process. This richer representation appears to help the model disambiguate between subtle visual features.

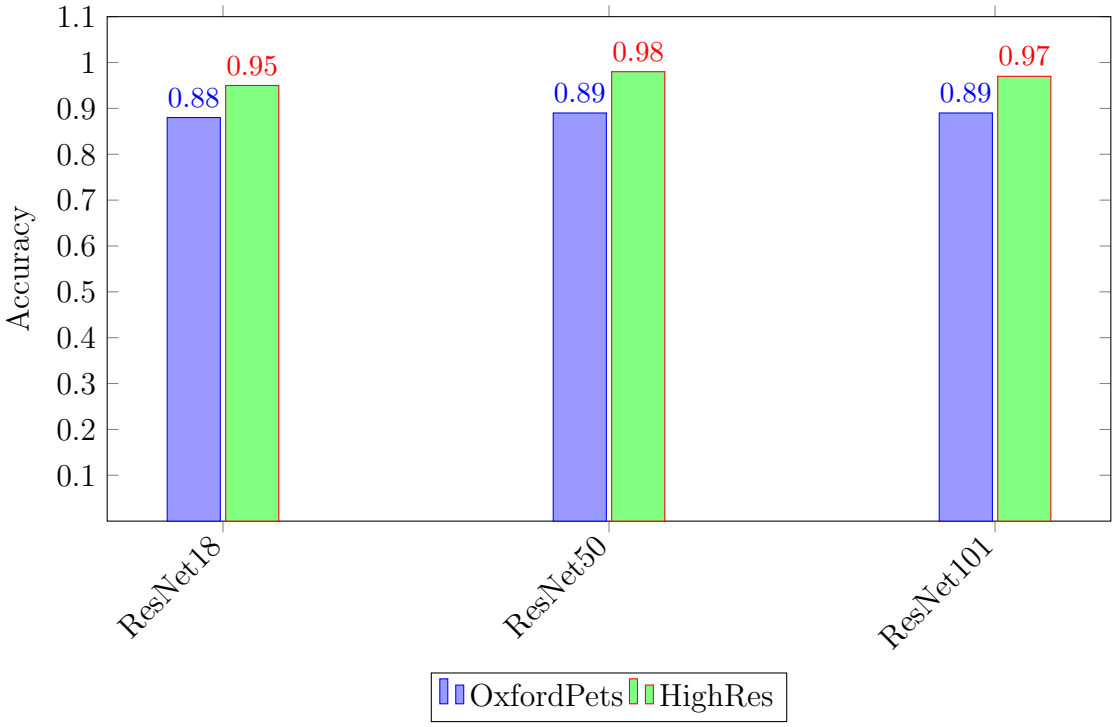

Figure 4.6: Bar chart indicating **performance of Superpatcher models on OxfordPets and HighRes** at 1024x1024.

However, at 1024×1024 resolution, a slight decline was observed across all backbones, indicating a point of diminishing returns, where the added data volume may introduce complexity or redundancy without significant learning benefit.

4.2.1 Superpatcher with Additional Encodings

To evaluate the impact of spatial structural information, we incorporated positional encodings (PE) and covariance-based encodings (COV) into the transformer input. The results of training the Superpatcher model with these encodings is presented in Table 4.3. While these encodings were expected to enhance performance by providing additional spatial context, the observed results were mixed. The modest gains—or lack thereof—suggest that the model may have reached a performance plateau, or that the inherent structure provided by the superpixel representation already encodes sufficient spatial information. Consequently, the addition of explicit positional or shape-based encodings may offer limited benefit or even

introduce redundancy. The following tables summarize the validation results for each model configuration across different datasets and input resolutions.

Table 4.3: Table outlining the **validation accuracy of the superpixel-based models with encodings on the OxfordPets and the High Res** dataset at various scales.

Scale	Superpatcher Model	OxfordPets	HighRes
224x224	ResNet18+Encodings	0.90	0.97
	ResNet50+Encodings	0.93	0.98
	ResNet101+Encodings	0.93	0.98
512x512	ResNet18+Encodings	0.94	**0.99**
	ResNet50+Encodings	0.95	**0.99**
	ResNet101+Encodings	0.92	**0.99**
1024x1024	ResNet18+Encodings	0.85	0.95
	ResNet50+Encodings	0.88	0.94
	ResNet101+Encodings	0.87	0.97

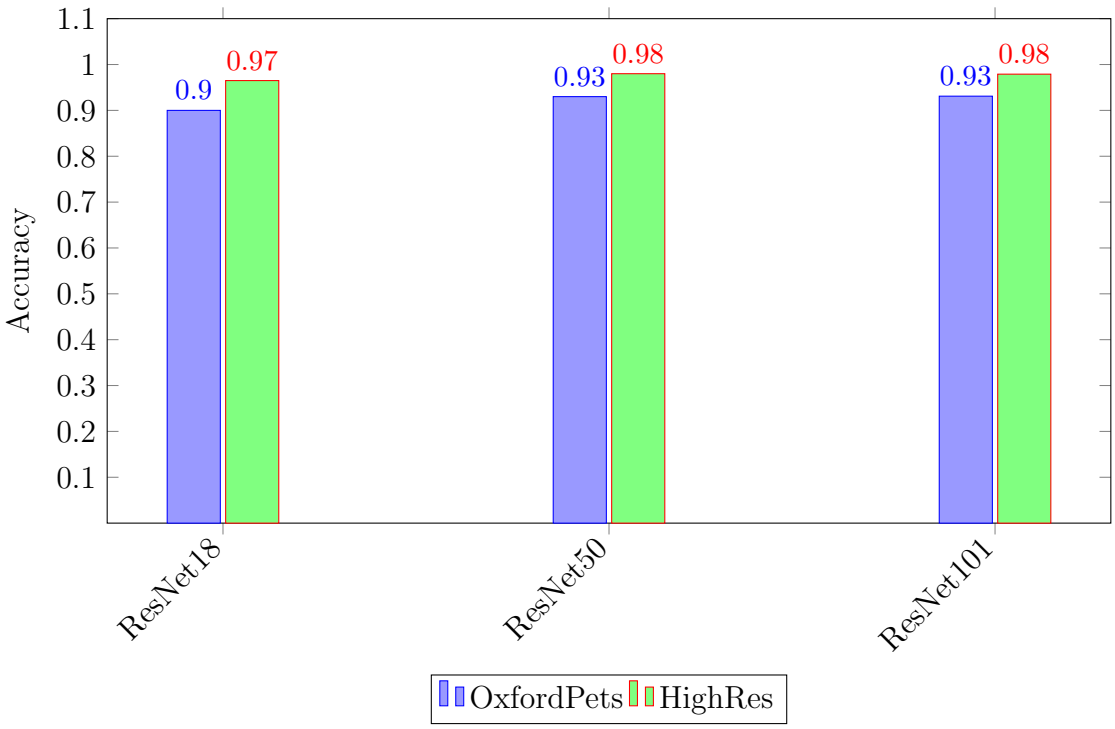

Figure 4.7: Bar chart indicating **performance of Superpatcher models with encodings on OxfordPets and HighRes** at 224x224.

At identical resolutions and with consistent model configurations, the inclusion of positional encodings did not yield a measurable improvement in classification accuracy. In some cases, the added spatial information may have interfered with or diluted the discriminative signals already captured through superpixel-based representations. This suggests that, within this architectural framework, the integration of additional encodings may not always be beneficial and could potentially introduce noise. Further experimentation across varied datasets and model variants may help clarify the conditions under which these encodings contribute meaningfully to performance.

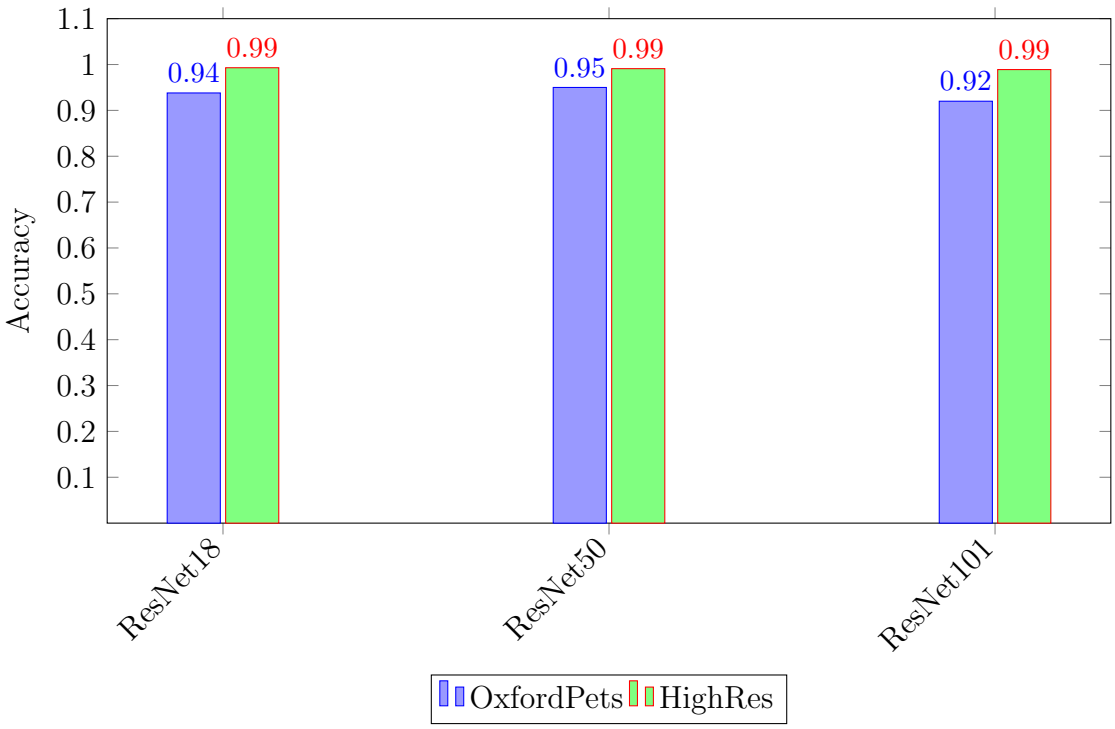

Figure 4.8: Bar chart indicating **performance of Superpatcher models with encodings on OxfordPets and HighRes** at 512x512.

Interestingly, the inclusion of additional encodings appeared to improve performance at higher input resolutions across all model configurations. This trend suggests that, at larger scales, positional and covariance information may help the model better disambiguate fine-grained spatial features. In contrast, at lower resolutions, the element-wise addition of these encodings may obscure salient information already captured by the superpixel features. This resolution-dependent effect highlights the nuanced role that auxiliary spatial encodings play within the model architecture.

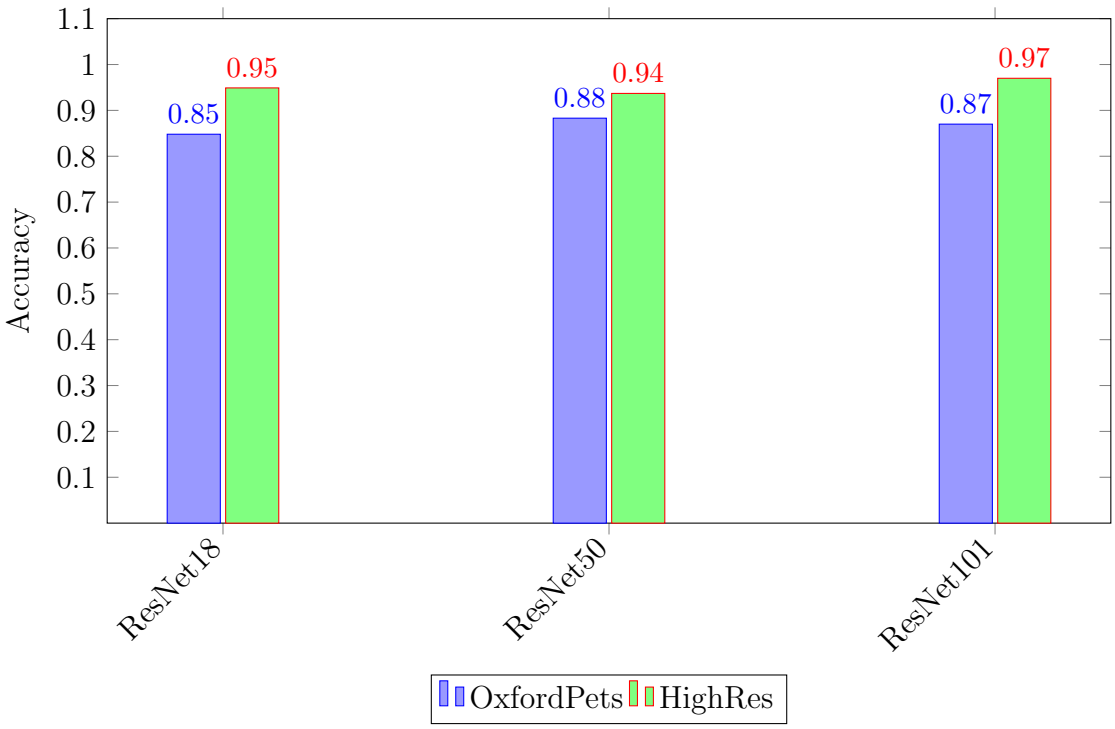

Figure 4.9: Bar chart indicating **performance of Superpatcher models with encodings on OxfordPets and HighRes** at 1024x1024.

This trend persists at the 1024×1024 resolution, where the inclusion of positional encodings continues to offer a marginal benefit over models without them. However, overall performance declines slightly relative to the 512×512 configuration across all model variants. This suggests that although spatial encodings enhance the model's ability to capture fine-grained structural information, the substantial increase in input size may introduce redundancy or noise that diminishes their overall effectiveness. As such, there appears to be a resolution threshold beyond which the added spatial detail no longer yields proportional performance gains. Carefully balancing spatial granularity with model capacity remains essential when scaling to higher resolutions.

4.2.2 Comparing Models Across Scales

Figure 4.10 presents a line plot comparing the validation accuracy of all baseline and Superpatcher models on the Oxford Pets dataset across resolutions. The visualization highlights

both the relative improvements achieved by the superpixel-based architectures and the nuanced effects of increasing resolution on model performance.

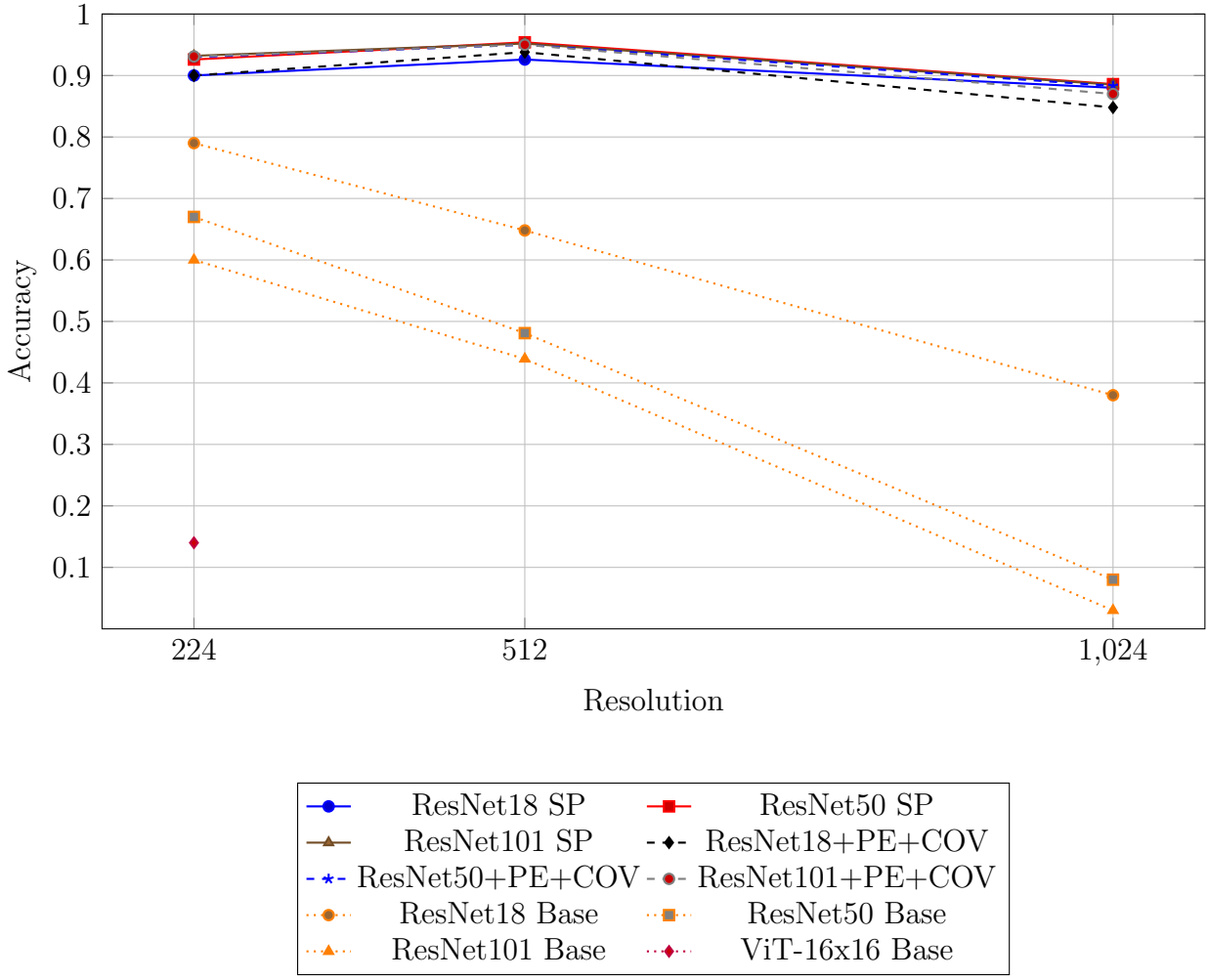

Figure 4.10: **OxfordPets validation accuracy across resolutions for all model configurations.** Solid lines indicate superpixel models; dashed lines indicate superpixel+encodings; dotted lines represent baseline (non-superpixel) models.

Figure 4.11 presents a line plot comparing the validation accuracy of all baseline and Superpatcher models on the custom High-Resolution Bird-Cat-Dog dataset. The plot illustrates the comparative performance trends across resolutions, emphasizing the advantages of the superpixel-based approach in handling high-resolution natural images.

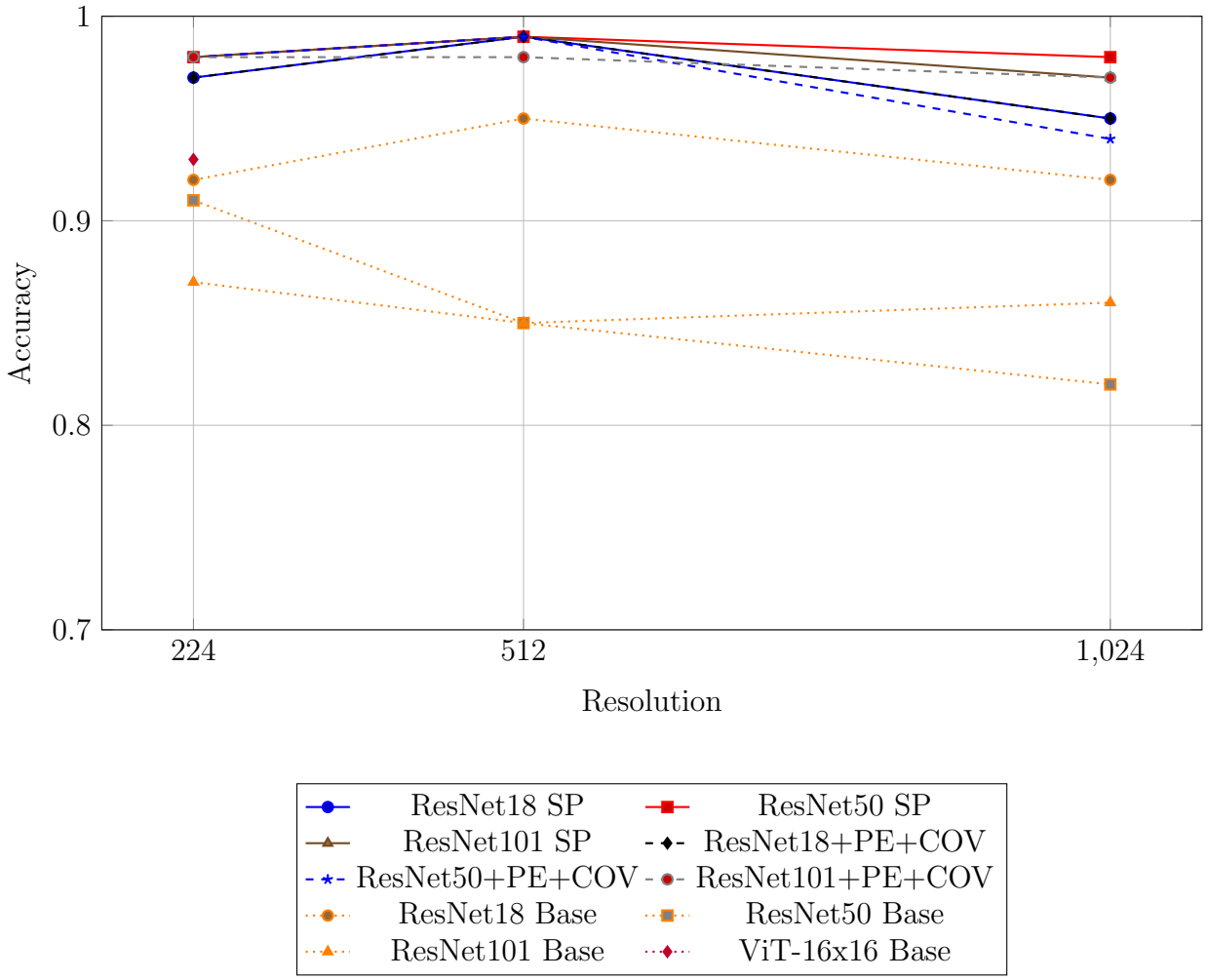

Figure 4.11: **High Resolution Bird-Cat-Dog validation accuracy across resolutions for all model configurations.** Solid lines indicate superpixel models; dashed lines indicate superpixel+encodings; dotted lines represent baseline (non-superpixel) models.

4.2.3 Comparing to ImageNet

The Oxford Pets and HighRes datasets were selected specifically because it showcases that our model is well suited to learn on small datasets at various input resolutions. However, it is also worth considering whether the Superpatcher model can improve performance on ImageNet itself which the backbone ResNet networks were originally trained on. This would indicate that the superpixel aggregation and attention mechanisms improve the generalization of models over a standard CNN.

The results of this experiment are shown in Table 4.4. Because of memory constraints,

only the ResNet18 Superpatcher model could be trained on ImageNet in a reasonable amount of time.

Table 4.4: Table outlining **performance of Superpatcher models with encodings on ImageNet** at 224x224.

Model	ImageNet Accuracy
ResNet18	0.67
ResNet50	0.74
ResNet101	0.76
16x16 ViT	0.88
Superpatcher-ResNet18	**0.66**

Our current results in the ImageNet under perform in regards to what we anticipated based on the other two natural image dataset results. This may be due to the restriction of training with lower batch sizes due to hardware constraints when working with ImageNet and superpixels together. This may also indicate that the superpatcher is better equipped to handle small scale nuanced datasets like Oxford Pets as opposed to large scale broad spectrum datasets.

4.3 WSI Superpatcher

The highest validation accuracy achieved by the WSI-Superpatcher model on the binary classification task (Normal vs. Tumor) using the Camelyon-16 dataset was 63.2%. Figures 4.10 and 4.11 illustrates the range of peak accuracies and loss observed across one of our best training runs.

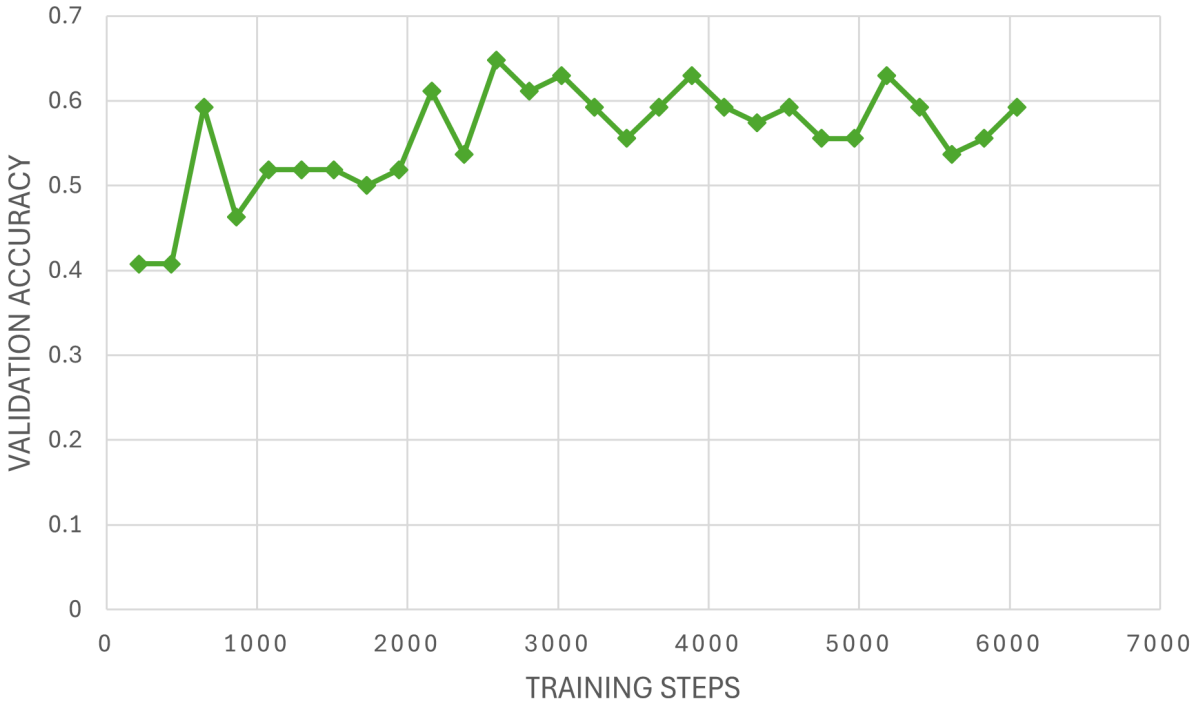

Figure 4.12: **Validation Accuracy of the WSI Superpatcher** across training steps with a learning rate 1e-4 on the Camelyon-16 Dataset.

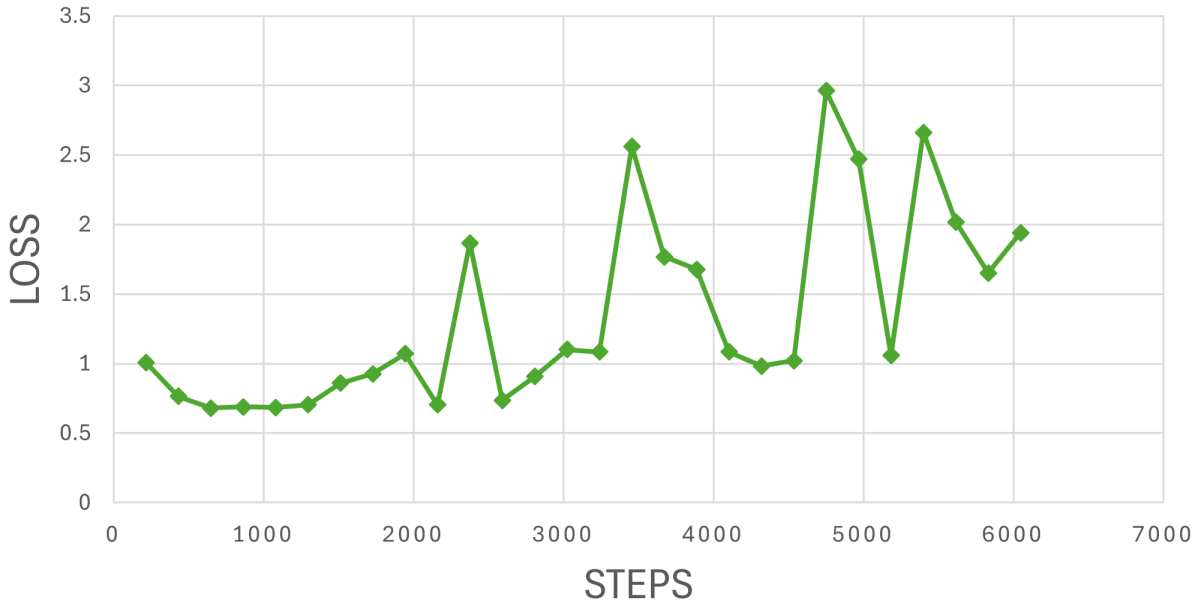

Figure 4.13: **Loss of the WSI Superpatcher** across training steps with a learning rate 1e-4 on the Camelyon-16 Dataset.

Due to the inherently variable nature of whole slide images (WSIs), it is challenging to report a consistent or meaningful average pixel resolution across samples. WSIs vary significantly in size and scale, with pixel counts often reaching into the millions. For this study, all slides were processed at scale level 5, which offered a practical trade-off between resolution and computational feasibility.

During training, the model exhibited consistent improvement in classification accuracy, reaching a peak of 63.2% before often settling around 59% in later epochs. To better understand the learning dynamics, training loss was monitored per epoch and followed a roughly linear decline during the initial stages of training. This trend indicates that the model was indeed learning meaningful features and leveraging them to improve classification performance, even in the presence of high-resolution and highly heterogeneous input data.

Given the extreme size of whole slide image (WSI) data, samples had to be processed individually rather than in conventional multi-image batches. As a result, we were unable to leverage standard batch-based optimization, which introduced instability in the loss values due to the high variability across individual WSI samples. Despite this, the network demonstrated the ability to learn effectively, with classification accuracy steadily improving and the training process stabilizing over time.

It is worth noting that even correctly classified examples can contribute to higher loss if the model's confidence is low or inconsistent. Since accuracy is a threshold-based metric that only considers whether predictions fall on the correct side of the decision boundary, it is possible for a model to improve in accuracy, by making more correct predictions overall, while simultaneously experiencing increased loss due to fluctuating confidence levels. This phenomenon is particularly evident in binary classification tasks, where small changes in predicted probabilities can disproportionately affect the computed loss without negatively impacting overall accuracy.

Chapter 5

Conclusion and Future Work

5.1 Key Findings

This research set out to explore whether incorporating superpixels into a transformer-based architecture could improve image classification performance across a range of visual domains, focusing on natural images and high-resolution medical slides. The results consistently demonstrate that leveraging superpixels as structured, semantically meaningful alternatives to traditional patching provides clear advantages, particularly when integrated with CNN backbones and transformer encoders.

Across standard datasets such as Oxford Pets and ImageNet subsets, our Superpatcher model outperformed conventional ResNet and Vision Transformer baselines at nearly every resolution tested. This performance gap was especially evident at mid-to-high resolutions (e.g., 512×512), where superpixels facilitated more effective spatial grouping and reduced the token complexity presented to the transformer. These findings suggest that the superpixel segmentation step contributes significantly to improving the model's ability to attend to salient regions of the image. This may indicate that encoded superpixels reduce the inductive burden placed on the self-attention mechanism.

Furthermore, the integration of positional and covariance encodings revealed a resolution-dependent benefit. While these additional embeddings did not improve performance at lower resolutions—and in some cases, degraded it, they contributed positively at higher resolutions, likely by helping the transformer distinguish between similarly shaped or positioned super-

pixel regions. This may indicate that transformers may not always require explicit spatial encodings when working with nuanced input representations like superpixels but benefit from them when the visual complexity increases. The results suggest that superpixels alone provide a strong inductive bias, but the addition of auxiliary encodings becomes more valuable as input size, and consequently information density grows.

In the case of Whole Slide Image (WSI) classification using the Camelyon-16 dataset, the Superpatcher demonstrated the ability to learn from extremely high-resolution medical images, achieving up to 63.2% accuracy in a binary classification task. Although these results were more modest than those obtained on standard datasets, they nonetheless suggest that superpixel-based representations can scale to the demands of medical imaging, where input sizes and spatial variability pose unique challenges. The fact that the model was able to operate at this scale without resorting to thumbnailing the entire slide is a promising sign of viability.

5.2 Implications for Medical Imaging and Beyond

Although the results on Camelyon-16 did not reach state-of-the-art performance levels, they point toward the potential utility of superpixels in medical image analysis. Traditional approaches to WSI classification often rely on heavy-handed downsampling or uniform patch-based methods that risk discarding important spatial information. In contrast, our model processes slides more holistically, using superpixels to intelligently compress the input without losing key spatial cues. This approach allowed for relatively high-resolution analysis directly on WSIs, demonstrating that the Superpatcher can serve as a viable backbone for future medical imaging pipelines.

The main limitation encountered in this domain was not conceptual, but computational. Despite a scalable design, the Superpatcher's resource demands—especially when paired with deeper CNN backbones and high-resolution slides—quickly surpassed what was feasible with the available hardware. This constraint curtailed deeper exploration into large-scale WSI

datasets and more complex model variants that could further validate or refine these early findings. Given the superpatcher's performance on low resolution natural images, it may be worth exploring the superpatcher's performance on low resolution medical imaging as well.

5.3 Hardware Limitations

All experiments for this thesis were conducted using three workstations equipped with NVIDIA RTX 4090 GPUs, each offering 24GB of VRAM. While this setup initially seemed adequate, the memory footprint of the Superpatcher—particularly at 1024×1024 resolutions or when using deeper CNNs like ResNet-101—proved to be a persistent bottleneck. These constraints were most pronounced when processing WSIs, as the slide-level input sizes often reached into the millions of pixels and generated high-density large superpixel patches. Likewise, our attempt to explore a meaningful subset of the ImageNet dataset was hindered by memory overflows and prolonged training times, limiting the depth of experimentation that could be carried out. These limitations restricted our ability to fully scale the model or examine the complete generalization capabilities of the Superpatcher on more demanding datasets.

5.4 Future Work

Several promising avenues for future research emerge from the findings and constraints of this work. First, a more thorough investigation of the ImageNet dataset is warranted. With access to expanded computational resources, it would be feasible to conduct a large-scale evaluation of the Superpatcher's generalization capacity across a more diverse set of categories and image types. This could help to further validate the scalability of the superpixel-transformer paradigm. Optimizing the memory foot print of the Superpatcher network would also lead to increased performance and reduce restrictions.

Second, additional work is needed to optimize the model for WSI classification. Future iterations may incorporate alternate superpixel segmentations methods, meaningful patch

selection and filtering, or different architectural components to either boost computational efficiency or predictive performance. Exploring further options with better hardware could improve both accuracy and efficiency, particularly in the medical imaging domain where detail and precision are critical.

Third, the utility of the Superpatcher architecture should be tested on a broader range of tasks beyond classification. Applications such as semantic segmentation, object detection, or even image captioning may benefit from the interpretable and structured representations that superpixels provide.

Finally, the hardware constraints encountered during this research underscore the need for access to more advanced computing infrastructure. Leveraging distributed training, memory-efficient Superpatcher variants, or more powerful hardware accelerators to unlock higher levels of performance and scalability.

In summary, this thesis presents compelling evidence that superpixel-guided transformers are a promising direction for image analysis tasks. The results suggest a strong foundation for further exploration into both general-purpose computer vision and specialized domains such as digital pathology.

BIBLIOGRAPHY

[1] Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., and Süsstrunk, S. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence 34*, 11 (2012), 2274–2282.

[2] Amoroso, R., Tomei, M., Baraldi, L., and Cucchiara, R. Superpixel positional encoding to improve vit-based semantic segmentation models. In *34th British Machine Vision Conference 2023, BMVC 2023, Aberdeen, UK, November 20-24, 2023* (2023), BMVA.

[3] Chen, R. J., Lu, M. Y., Shaban, M. T., Chen, T. Y., and Mahmood, F. Transmil: Transformer-based correlated multiple instance learning for whole slide image classification. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)* (2021), Springer, pp. 139–149.

[4] Dai, Z., Liu, H., Le, Q. V., and Tan, M. Coatnet: Marrying convolution and attention for all data sizes. In *NeurIPS* (2021).

[5] Dang, G., Mao, Z., Zhang, T., Liu, T., Wang, T., Li, L., Gao, Y., Tian, R., Wang, K., and Han, L. Joint superpixel and transformer for high resolution remote sensing image classification. *Scientific Reports 14*, 1 (2024), 5054.

[6] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations (ICLR)* (2021).

[7] Heo, B., Yun, S. J., Han, D., Kim, S., Choi, H., Yoo, Y., and Yoon, S. Rethinking spatial dimensions of vision transformers. In *Conference on Computer Vision and Pattern Recognition (CVPR)* (2021).

[8] Ilse, M., Tomczak, J. M., and Welling, M. Attention-based deep multiple instance learning. In *International Conference on Machine Learning (ICML)* (2018).

[9] Lew, J., Jang, S., Lee, J., Yoo, S., Kim, E., Lee, S., Mok, J., Kim, S., and Yoon, S. Superpixel tokenization for vision transformers: Preserving semantic integrity in visual tokens, 2025.

[10] Li, J., Chen, Y., Chu, H., Sun, Q., Guan, T., Han, A., and He, Y. Dynamic graph representation with knowledge-aware attention for histopathology whole slide image analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (June 2024), pp. 11323–11332.

[11] Lu, M. Y., Williamson, D. F. K., Chen, T. Y., Chen, R. J., Barbieri, M., and Mahmood, F. Data-efficient and weakly supervised computational pathology on whole-slide images. *Nature Biomedical Engineering 5*, 6 (2021), 555–570.

[12] Mei, J., Chen, L.-C., Yuille, A., and Xie, C. Spformer: Enhancing vision transformer with superpixel representation, 2024.

[13] Ouyang, C., Biffi, C., Chen, C., Kart, T., Qiu, H., and Rueckert, D. Self-supervision with superpixels: Training few-shot medical image segmentation without annotation. In *Computer Vision – ECCV 2020* (Cham, 2020), A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, Eds., Springer International Publishing, pp. 762–780.

[14] Park, J., Gharaee, Z., and Fieguth, P. W. Superformer: Superpixel-based transformers for salient object detection, 2024.

[15] Ronneberger, O., Fischer, P., and Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)* (2015), Springer, pp. 234–241.

[16] Srinivas, A., Lin, T.-Y., Parmar, N., Shlens, J., Abbeel, P., and Vaswani, A. Bottleneck transformers for visual recognition. In *Conference on Computer Vision and Pattern Recognition (CVPR)* (2021).

[17] Wu, H., Xiao, B., Codella, N., Liu, M., Dai, X., Yuan, L., and Zhang, L. Cvt: Introducing convolutions to vision transformers. In *International Conference on Computer Vision (ICCV)* (2021).

[18] Zeng, S., Zhu, L., Zhang, X., He, H., and Lu, Y. Supercl: Superpixel guided contrastive learning for medical image segmentation pre-training, 2025.

[19] Zhao, Y., Fu, Y., Dong, J., Zhang, H., Yang, X., and Wang, L. Graph-based weakly supervised learning for whole slide image classification. In *European Conference on Computer Vision (ECCV)* (2020), pp. 624–641.

[20] Zhu, A. Z., Mei, J., Qiao, S., Yan, H., Zhu, Y., Chen, L.-C., and Kretzschmar, H. Superpixel transformers for efficient semantic segmentation. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (2023), pp. 7651–7658.