

Title

Abstract

Table of Contents

1	Introduction and background	1
1.1	Background	1
1.2	Introduction	1
1.3	Structure of the Report	2
2	Literature review	3
2.1	Microtome and Microscope Selection	3
2.2	Deep Learning in Tissue Sectioning	3
3	Methodology and theory	4
3.1	Computer Vision - Image Segmentation	4
3.1.1	Edge Detection	4
3.1.2	Theresold Segmentation	5
3.2	Deep Learning	6
3.2.1	Convolutional Neural Networks (CNN)	6
3.2.2	Transfer Learning	7
4	Experimental work/ analytical investigation/ design	8
4.1	Data collection	8
4.2	Data labeling	9
4.3	模型 1: 原始图像 + 简单的 cnn 网络	11
4.4	改进: 图片预处理	13
4.4.1	边缘检测	13
4.4.2	阈值分割	16
4.4.3	另一种阈值分割方法-指纹算法	16
4.4.4	小结	17
4.5	模型 2: 预处理图像 + 简单的 cnn 网络	17

4.5.1 小结	17
4.6 模型 3: 原始图像 + 迁移学习	18
4.6.1 小结	19
4.7 模型选择总结	19
5 Presentation of experimental or analytical results/descriptions of final constructed product . .	20
5.1 带入测试集验证准确度	20
5.2 模型的进一步提高（改变输入分辨率）	20
5.3 探究机器的最佳切削角	21
5.4 模型通用性	22
6 Discussion and conclusions	24
6.1 Discussion of results	24
6.2 Future work	25
6.3 Conclusions	26
7 Project management, consideration of sustainability and health and safety	27
7.1 Subsection 5.1	27
7.2 Subsection 5.2	27
References	27

1 Introduction and background

1.1 *Background*

As the fundamental units of life, human research into cells and tissues has never ceased. Biological tissue sections, serving as crucial means for the direct observation of cellular morphology and structure, are essential for biomedical research and clinical diagnosis. A complete and usable tissue section is of great importance to researchers and physicians, as it provides vital information about cell structure, tissue morphology, and pathological changes. Within this, the quality of the section is of paramount importance.

Traditional manual sectioning methods are time-consuming and prone to variability, hence the emergence of automatic microtomes has provided a solution to these issues. For different biological tissues, varying cutting parameters can yield differing results, both positive and negative. Therefore, to enhance the utilization rate of biological sections and increase the production of high-quality specimens, determining optimal cutting parameters for specific tissues remains a goal.

Machine learning and deep learning have achieved significant success in the fields of computer vision and image processing. Machine learning is defined as a series of methods that can automatically detect patterns in data, which are then used to predict future outcomes or make decisions [1]. In this paper, we integrate advanced image analysis and machine learning techniques to identify section quality and then evaluate the quality of tissue samples under different sectioning parameters.

1.2 *Introduction*

Project Overview

This project aims to optimize the cutting parameters of biological tissue microtomes, which are crucial devices in biomedical research and clinical diagnostics. The objective is to enhance the precision and efficiency of tissue sample preparation by determining the optimal slicing conditions. By collecting tissue samples under various cutting parameters and conducting subsequent manual image classification, this study employs deep learning techniques to analyze and predict the most effective cutting parameters. This work not only promises to improve the quality of tissue samples for microscopic examination but also helps to simplify laboratory workflows, thereby advancing biological and medical sciences.

Objectives:

1. Collect a comprehensive dataset of tissue samples sliced under different parameters.
2. Employ artificial image classification to categorize the quality and characteristics of these samples.
3. Develop and train a deep learning model capable of assessing tissue sample quality.
4. Use the model's insights to determine the optimal cutting parameters for the tissue slicer.
5. Validate the model's predictions through empirical testing and refinement.

1.3 *Structure of the Report*

The project is structured into several chapters, each aimed at systematically exploring research background, methodologies, experimental work, results presentation, discussions and conclusions, as well as considerations for project management, sustainability, and health safety:

Introduction and Background - This chapter outlines the project's goals, objectives, and structural arrangements. It briefly introduces the motivation and necessity for the research, as well as the adopted technical protocols and standards.

Literature Review - Provides an in-depth discussion on biological tissue sections, image classification, and the application of deep learning in the preparation of biological samples. This section positions the current study within the context of existing research.

Methods and Theory - Detailed description of experimental methods, theoretical frameworks, and specific plans for data collection and processing.

Experimental Work/Analysis Investigation/Design - Describes in detail the experimental design, implementation, and analytical survey. It elaborates on the strategies and methods adopted to achieve the project's objectives.

Presentation of Experimental or Analysis Results/Final Constructed Product Description - This chapter presents experimental data, analysis results, or final design products, detailing the outcomes of experiments or designs.

Discussion and Conclusion - Analyzes the results, discussing their scientific significance and practical value. This chapter also provides conclusions from the research and suggests potential directions for future studies.

Project Management, Sustainability, and Health Safety Considerations - Discusses project management strategies, sustainability issues, and health safety measures to ensure efficient and safe conduct of the research work.

References - Lists all the referenced literature, supporting the research and providing a foundation for the study.

Assumptions and Technical Specifications

The project is based on several key assumptions and technical protocols, which include:

1. Consistency in the properties of tissue samples across different batches.
2. The reliability and accuracy of the biological tissue microtome and imaging equipment.
3. The adequacy of deep learning models in interpreting complex biological image data.

Technical specifications regarding tissue microtome settings, image classification standards, and deep learning architecture are detailed in the Methods and Theory section.

2 Literature review

This literature review explores the integration of technologies in biological tissue sectioning, with a particular focus on the application of image classification and deep learning in optimizing slicing parameters. It aims to highlight significant advancements, identify gaps in current methodologies, and lay the groundwork for the proposed project.

2.1 Microtome and Microscope Selection

In recent years, the advent of automatic microtomes has significantly simplified the sectioning process and improved the quality of sections.

Zimmermann, in the article "Improved reproducibility in preparing precision-cut liver tissue slices," advocates for the use of the new Leica vibratome to enhance the accuracy and reproducibility of tissue sections from rats, mice, and human tissues [2].

In this experiment, the HM355S microtome provided by Epredia is used for sectioning. This machine is a popular device for biological tissue sectioning research, and many experiments and papers have utilized this equipment for sectioning.

Elzbieta Klimuszko has used the HM355S microtome for sectioning teeth to investigate the calcium and magnesium content in dental enamel [3].

Andelko Hrzenjak also used the HM355S microtome for sectioning pathological endometrial tissues to study the mechanisms of endometrial carcinoma development [4].

Similarly, the choice of microscope is crucial. In this experiment, the VHX7000 microscope from Keyence is used for image acquisition. It is capable of capturing images of biological tissue sections (e.g., mouse prostate cells [5]), as well as inorganic materials (such as ceramics [6], glass [7]).

The experiments will employ the HM355s microtome and VHX7000 microscope for sectioning and image acquisition. This setup ensures that both equipment selection and technological application are optimally aligned to enhance the precision and efficiency of the tissue sectioning process, supporting the overall goals of the research project.

2.2 Deep Learning in Tissue Sectioning

The application of deep learning technologies in the biomedical field has achieved significant advancements. Deep learning models excel in tasks such as image classification, object detection, and segmentation, providing powerful tools for research and diagnostics in biomedical laboratories.

Lorena Guachi-Guachi proposed a method utilizing CNN networks to identify and refine tissue sections. This approach represents an innovative application of deep learning that can enhance the precision of tissue preparation and analysis [8].

In the book *Biomedical Texture Analysis*, Vincent Andrearzyk introduced a CNN architecture specifically designed for texture analysis, which significantly improves the accuracy of classifying biological

tissues compared to traditional architectures [9]. This development demonstrates the potential of deep learning to enhance the detailed analysis of tissue characteristics, which is crucial for accurate diagnostics and research.

Yan Xu suggested that features extracted from CNNs trained on the large natural image database, ImageNet, can be transferred to histopathological images of tissues. This provides a viable approach for implementing transfer learning, which can greatly enhance the efficiency of tissue image classification and analysis [10].

Based on the literature, deep learning technology holds broad prospects for application in image classification and analysis of tissue sections. By leveraging deep learning models, efficient identification and classification of tissue samples can be achieved, providing strong support for optimizing sectioning parameters.

This section underscores the transformative impact of deep learning on the field of tissue sectioning, promising significant improvements in the accuracy and utility of histological analyses.

3 Methodology and theory

3.1 Computer Vision - Image Segmentation

For the acquired image data, appropriate image preprocessing can be applied. Under the premise of maintaining the integrity and quality of images, certain processing can be implemented to highlight the features intended for computer recognition and, to some extent, remove irrelevant features and noise. This enhances the accuracy of subsequent deep learning models.

Image segmentation is a critical step in image processing, aiming to divide the image into several meaningful regions for further analysis and processing. In models focusing on the yield rate of biological tissues, it is necessary to segment the biological sections into biological tissue and paraffin areas, emphasizing the biological tissue parts.

Common image segmentation algorithms include edge detection and threshold segmentation.

3.1.1 Edge Detection

For biological tissue sections, a crucial indicator of quality is the clarity of the section's edges. The integrity and continuity of the slice edges can reflect whether there are quality issues with the sample.

There are numerous algorithms for edge detection, such as Sobel, Laplacian, and Canny operators [11].

The **Sobel operator** is a first-order differential operator that can be used to detect image edges [12]. Suppose there is a one-dimensional image $f(x)$, the relationship between its intensity and the pixel coordinate x can be represented as shown in Figure 1. It can be observed in Figure 3.1 that the slope is the largest around $x=2.2$, indicating that there is a sudden change in image intensity (an edge exists) near this point. Taking its derivative gives the first-order derivative $f'(x)$, as shown in Figure 3.2, where

the absolute value of the derivative is the largest. The Sobel operator uses this characteristic to detect edges.

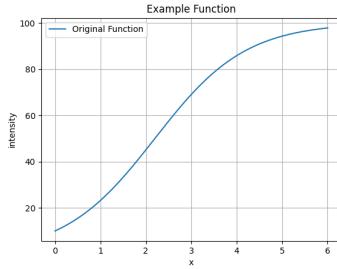


Figure 3.1 $f(x)$

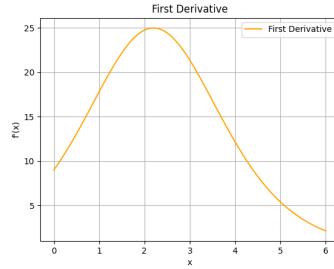


Figure 3.2 $f'(x)$

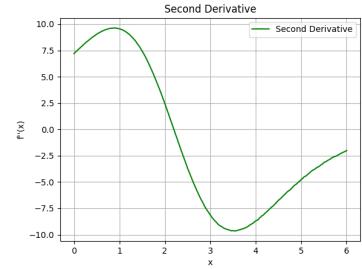


Figure 3.3 $f''(x)$

The **Laplacian operator** is a second-order differential operator that performs well in edge detection of images. It is derived by taking the derivative of the Sobel operator once more. In 2D images, the Laplacian operator is defined as follows:

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \quad (3.1)$$

As shown in the figure above, taking the derivative of the first-order derivative results in the second-order derivative $f''(x)$, as shown in Figure 3.3. It can be seen that around $x=2.2$, the second-order derivative is 0, which indicates that when the value of the Laplacian operator $\nabla^2 f$ is 0, there is a sudden change in image intensity, indicating the presence of an edge.

Canny Operator is a multi-stage differential operator that enhances the edge detection process by incorporating noise suppression, building on the initial computations similar to those used by the Sobel operator. Introduced by John F. Canny in 1986 [13], the Canny operator refines the results obtained from Sobel operator calculations through additional steps such as non-maximum suppression and hysteresis thresholding. These steps set thresholds to eliminate false edges from the image, resulting in more accurate edge detection.

In the chapter on *Experimental Work/Analytical Investigation/Design*, experiments will be conducted on the collected image data using these three edge detection algorithms—Sobel, Laplacian, and Canny—to compare their effectiveness. This comparative analysis will help in identifying the most suitable method for edge detection in the context of tissue sectioning, where the clarity and precision of edges are vital for quality assessment. The results will guide the selection of the optimal algorithm to be integrated into the image processing pipeline, enhancing the capability of the system to accurately segment and analyze biological tissue sections.

3.1.2 *Threshold Segmentation*

Apart from edge detection, another method employed is threshold segmentation. This technique divides the image pixels into two categories: those above a certain threshold and those below it. It is

particularly useful in situations where there is a significant grayscale difference between the target and the background in the image.

For specimens, a straightforward approach is to contrast the colors of the paraffin area and the biological tissue area (which is stained during preparation), and then separate them using threshold segmentation. Assuming the biological tissue is yellow and the paraffin is white, setting a threshold could isolate the white parts of the image, leaving behind the biological tissue.

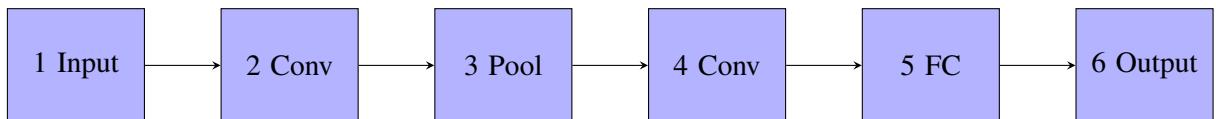
Additionally, there are more sophisticated methods of threshold segmentation, such as the Otsu method used for fingerprint extraction. Implementing this method can significantly enhance the segmentation of biological tissues. Yue Yaru and Zhu Jialin in "Algorithm of fingerprint extraction and implementation based on OpenCV" have proposed an improved Otsu-based fingerprint extraction algorithm using OpenCV. This algorithm excels particularly under conditions of uneven illumination and blurred images, providing accurate, simple, and fast fingerprint extraction [14].

Comparisons and experiments related to these segmentation techniques will be conducted in the *Experimental Work/Analytical Investigation/Design* chapter.

3.2 Deep Learning

3.2.1 Convolutional Neural Networks (CNN)

Convolutional Neural Networks (CNNs) are a type of deep learning model that are particularly effective at processing image data. They automatically learn spatial hierarchies of features through a series of convolutional layers without the need for manual feature extraction. A typical CNN model includes layers such as convolutional layers, pooling layers, and fully connected layers [15]. The architecture of a CNN is illustrated below:



In this model:

Convolutional layers (Conv): These are the core layers of a CNN, responsible for feature extraction from images.

Pooling layers (Pool): These serve to reduce the dimensionality of the feature maps, thereby decreasing the computational load.

Fully connected layers (FC): These integrate the features extracted by the convolutional and pooling layers for classification or regression analysis, eventually leading to the output.

The typical method for training a CNN involves several key processes:

1. **Forward Propagation:** Input data passes through each layer of the network until it reaches the output layer.

2. **Loss Computation:** The network's output is compared to the actual labels using a loss function, such as cross-entropy loss, to calculate the difference.
3. **Backpropagation:** The gradient of the loss function with respect to the network weights is computed.
4. **Weight Update:** The network weights are updated using an optimization algorithm such as gradient descent or its variants like Adam or RMSprop, with the aim of minimizing the loss function.

Once trained, the CNN can be employed to predict labels for new, unseen images. The distinctive feature of CNNs is their ability to automatically and efficiently learn features at different levels of abstraction, making them highly effective for tasks involving complex image data, such as medical image analysis, where accuracy and detail are paramount.

3.2.2 Transfer Learning

Indeed, for complex image tasks, constructing a simple CNN network is often insufficient. In such cases, transfer learning becomes essential. Transfer learning is a machine learning method that accelerates the training process by transferring knowledge from a pre-trained model to a new task. The core idea of transfer learning is to leverage knowledge from the source domain to aid learning in the target domain.

For CNN models, there are several approaches to transfer learning, such as fine-tuning and feature extraction:

Fine-tuning involves adjusting the parameters of a pre-trained model to adapt it to a new task. This often includes retraining some of the convolutional layers along with the fully connected layers on the new data, which allows the model to fine-tune the features to the specific characteristics of the new dataset.

Feature extraction involves using a pre-trained model as a fixed feature extractor, where only the fully connected layers are trained on the new data. In this approach, the convolutional layers retain their learned weights and act solely to extract features, which are then used by the newly trained classifier layers to perform tasks specific to the new dataset.

Commonly used pre-trained models include VGG, Inception, and others. These models have been extensively trained on large datasets like ImageNet, where the weights of various layers in the model have been optimized and can be effectively used for transfer learning.

Table 2 shows the number of parameters in models such as VGG16, VGG19 [16], InceptionV3 [17], Xception [18], etc. These models have a large number of parameters, allowing them to accurately extract features from complex images. The capability to leverage these well-trained models enables researchers and practitioners to achieve high performance on specific tasks without the need to train an entire network from scratch, saving both time and resources while maintaining high accuracy.

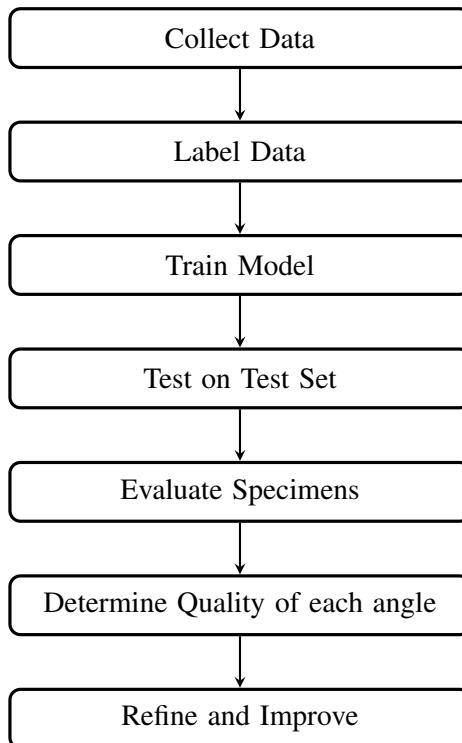
Table 3.1 Comparison of CNN Models

Model	VGG16	VGG19	InceptionV3	Xception
Number of Parameters	138,357,544	143,667,240	23,851,784	22,910,480

4 Experimental work/ analytical investigation/ design

Experimental workflow

The experiment workflow is outlined in the diagram below, detailing the sequential steps from data collection to iterative improvement of the model.



4.1 Data collection

The first essential step for deep learning is data collection. In this experiment, pre-prepared paraffin-embedded tissue sections (fish ovary tissues) were used. These sections were placed on an HM355s automatic microtome, and slicing operations were performed according to different cutting angles as specified in the microtome's manual. The cutting data was recorded meticulously.

The schematic diagram of the microtome (taking a tooth as an example) is shown in Figure 4.2 [19].

In the experiment, the microtome was configured with the following parameters: the mode was set to continuous, the feed rate was 5.0, the trimming value was 25, the speed was set at 32, the water flow rate was 7.5, and the water temperature was approximately 36 degrees Celsius. The cutting angle was adjusted between 8 to 12 degrees.

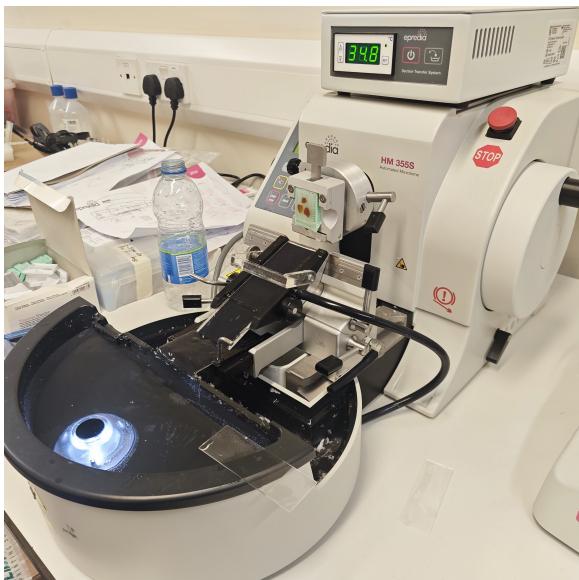


Figure 4.1 Microtome

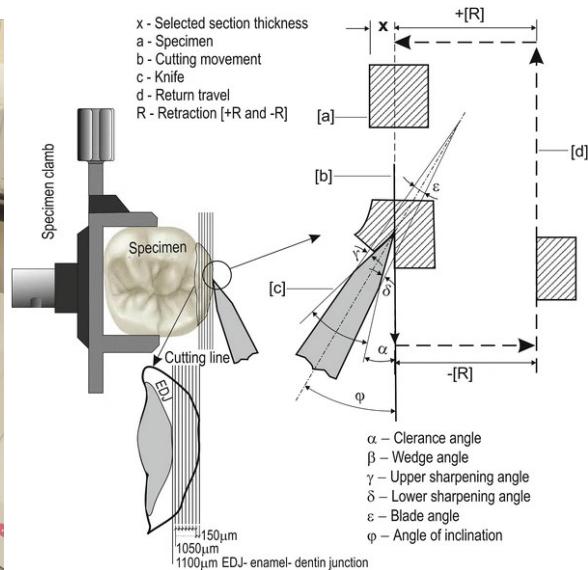


Figure 4.2 Working principle of the microtome

The biological tissues used for sectioning are shown in Figure 4.3. After sectioning, the different types of tissue sections were placed on slides as depicted in Figure 4.4. Once dried, these slides were transferred under a VHX7000 microscope for imaging. Each sample was photographed under the microscope to capture electronic image data, as shown in Figure 4.5.



**Figure 4.3 Biological tis-
sues**



**Figure 4.4 Collecting
samples**

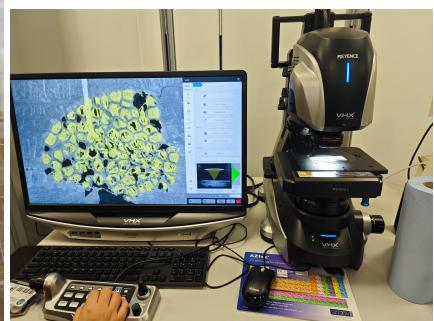


Figure 4.5 Microscope

Based on this, several hundred images were obtained, each with a resolution of 2880*2160. An example of the samples is shown in Figure 4.6.

4.2 Data labeling

For this experiment, the dataset is re-labeled based on the quality of the tissue sections. Overall, the quality of the biological tissues is categorized into two primary classes: normal and bad. Further analysis of the collected data revealed common flaws - the presence of vertical or horizontal white creases on the sections, which clearly indicate unusable slices. Given the distinct nature of these flaws,

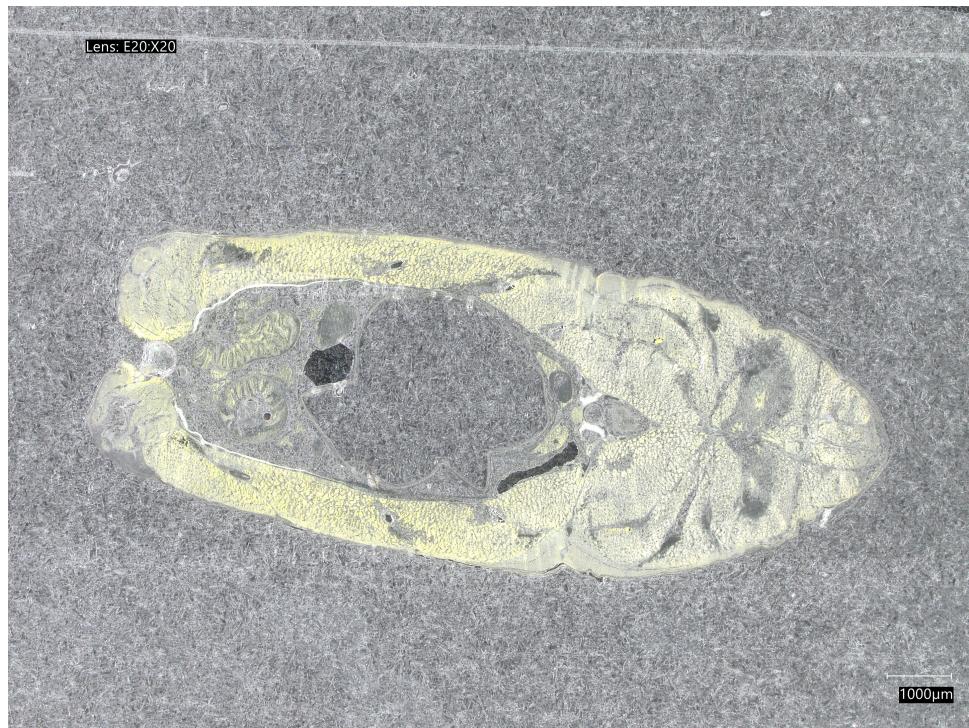


Figure 4.6 Sample with a cutting angle of 9.5 degrees

they are classified into two additional specific categories: **horizontal line**(Figure 4.7) and **vertical line**(Figure 4.8).

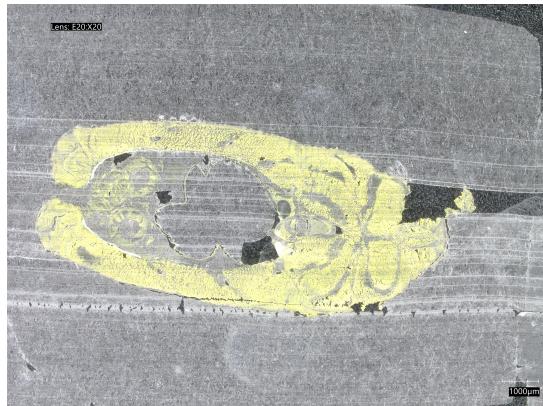


Figure 4.7 horizontal line

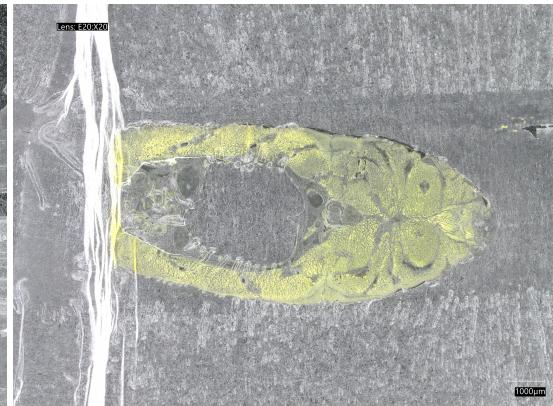


Figure 4.8 vertical line

Additionally, some images were noted to have a significant rotational angle at the time of sampling. These instances are categorized separately as **slope**(Figure 4.9). Finally, any images that do not fit into the aforementioned categories but still show irregularities(Excessive changes in brightness) are labeled as **other**(Figure 4.10).

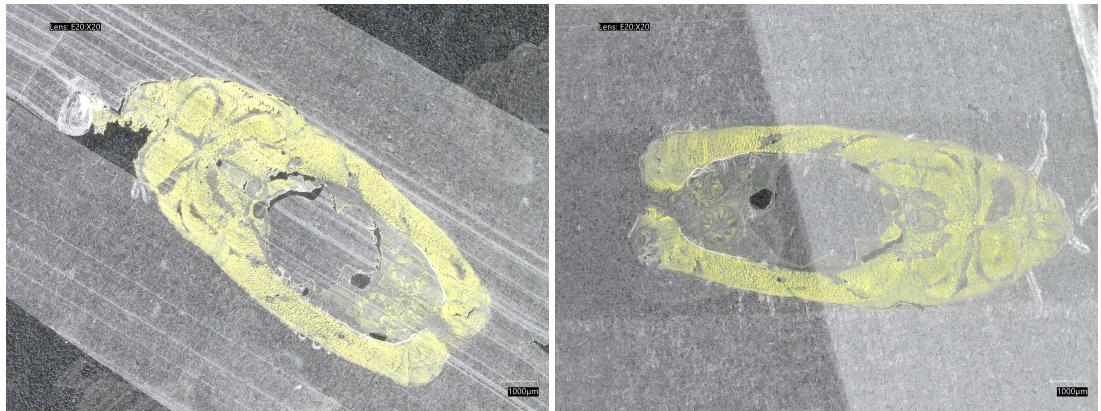


Figure 4.9 slope

Figure 4.10 other

An example of a normal slice that meets observational requirements is shown in Figure 4.11.



Figure 4.11 normal

For each image, we need to label it as one of the above five categories. This will serve as our dataset for training the model.

4.3 模型 1：原始图像 + 简单的 *cnn* 网络

对于一个全新的数据集，在不确定图像复杂度对应的何种模型之前，首先尝试一个简单的典型 *cnn* 网络（架构如下），以了解数据集的特点和图像复杂度。

Table 4.1 Configuration of the simple CNN model

Layer Type	Configuration 1a	Configuration 1b	Configuration 1c
Input Layer	-	-	-
Conv Layer 1	Conv3-32 (relu)	Conv3-16 (relu)	Conv3-32 (relu)
Pooling Layer 1	MaxPooling	MaxPooling	MaxPooling
Conv Layer 2	Conv3-32 (relu)	Conv3-32 (relu)	Conv3-32 (relu)
Pooling Layer 2	MaxPooling	MaxPooling	MaxPooling
Conv Layer 3	Conv3-32 (relu)	Conv3-64 (relu)	Conv3-32 (relu)
Pooling Layer 3	MaxPooling	MaxPooling	MaxPooling
Flattening Layer	Flatten()	Flatten()	Flatten()
FC(Full connect)	Dense(128, relu)	Dense(128, relu)	Dense(256, relu)
Output Layer	-	-	-

Table 4.1显示的三个初始模型，分别为 a, b, c。这三个模型的区别在于卷积层的数量和大小，全连接层的大小。a 和 b 相比修改了卷积层的神经元数量，c 相比 a 修改了全连接层的神经元数量。

在数据的预处理部分，先将数据集分为训练集和测试集，其中训练集占 80%，测试集占 20%。

在输入层，这里将图像的长宽缩小一倍（即输入大小从 2880*2160 变为 1440*1080）并归一化数据。

在训练过程中，我们使用了 Adam 优化器，交叉熵损失函数，使用早停。

下面图组展示了模型 1a, 1b, 1c 的准确度和损失随着训练次数的变化。

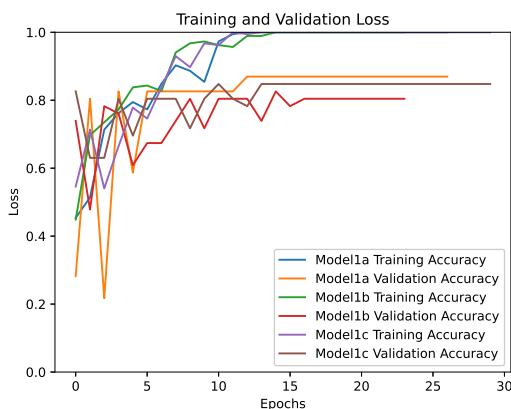


Figure 4.12 Model 1 accuracy

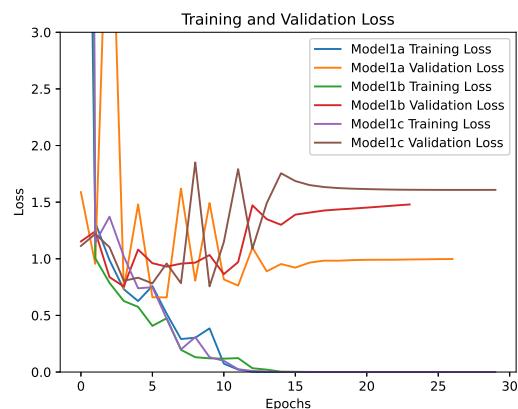


Figure 4.13 Model 1 loss

在图中，观察到模型 1a、1b 和 1c 在不同训练周期（Epochs）的准确度与损失的变化情况。模型 1a、1b 和 1c 的训练准确度随着时间逐步提高，趋向于稳定，而训练损失则呈下降趋势，接近于零，这表明模型在学习训练数据方面表现得相对良好。然而，对于验证集，三个模型的准确度似乎在约 80% 到 85% 的区间内稳定，而验证损失在某些情况下较高，特别是模型 1a 的验

证损失在后期趋近于 2.5，表现出较大波动。这表明存在一定程度的过拟合，即模型在未见过的数据上的表现不如在训练集上。特别值得注意的是，模型 1c 相对于其他模型而言，在验证损失方面表现最佳，这可能意味着其结构或参数调整对于泛化能力的提升更为有效。

在这里过拟合的原因推测可能是模型的复杂度不够，数据集的复杂度过高，模型无法很好的提取特征。这些结果指出虽然模型在训练集上能够实现高准确度和低损失，但在验证集上的泛化能力还有待提高。

因此，我们考虑通过对图像的预处理，人为辅助计算机进行特征提取，以提高模型的准确性。

4.4 改进：图片预处理

在模型表现能力欠佳的情况下，我们考虑是否是图像过于复杂导致模型难以提取出显著特征。因此我们考虑对图像进行预处理，以突出图像中我们希望让计算机识别的特征，并且在一定程度上去除图像的无关特征和噪声，以提高后续的深度学习模型的准确性。

在这里采用边缘检测，阈值分割两种方法对图像进行预处理。

4.4.1 边缘检测

正如在 3.1.1 中所提到的，边缘检测的原理是通过检测像素点的灰度值的变化（梯度）来确定图像中的边缘。假定原始图像是Figure 4.6.

在进行边缘检测之前，还需要进行一步前处理-高斯模糊。这么做的原因是，高斯模糊可以减少图像中的噪声，平滑图像的梯度，减小识别假边缘的几率，使得边缘检测更加准确。[20] 在高斯模糊核的选择上，选择高斯核分别为 21, 41, 61, 81（图像宽度的 1%, 2%, 3%, 4%）。高斯模糊后的图像如下所示。为了方便更直观的展示高斯模糊核对边缘检测的影响，这里采用 sobel 算子计算经过高斯模糊后的边缘并增加 50 的亮度。

从Figure 4.14到Figure 4.20可以看到，随着高斯模糊核的增大，图像的细节逐渐模糊，边缘也逐渐变得模糊。从Figure 4.15到Figure 4.21可以看到，随着高斯模糊核的增大，边缘检测的效果逐渐减弱，边缘变得不明显。考虑到图像边缘的清晰度和底噪的对比，我们选择高斯模糊核为 61。

以下是在高斯模糊 ($k=61$) 后使用 python 的 opencv 库执行 laplacian 算子的结果。

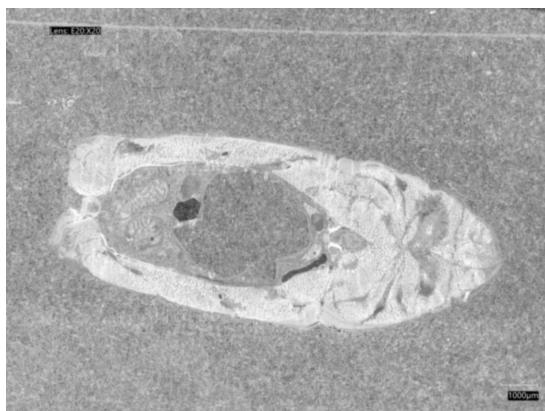


Figure 4.14 blurred k=21



Figure 4.15 sobel k=21

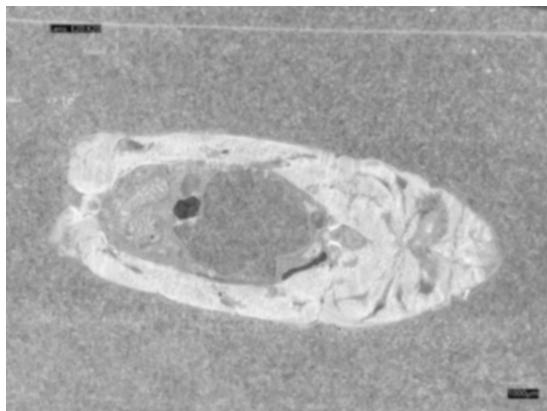


Figure 4.16 blurred k=41

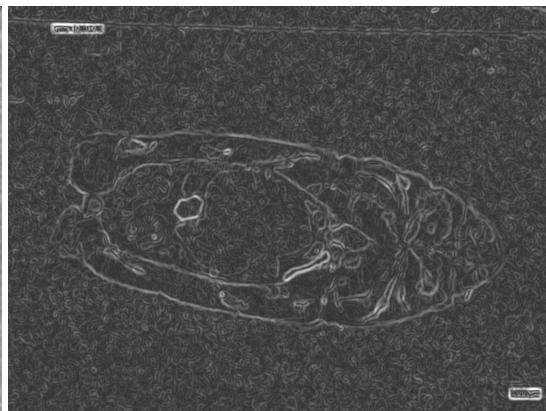


Figure 4.17 sobel k=41

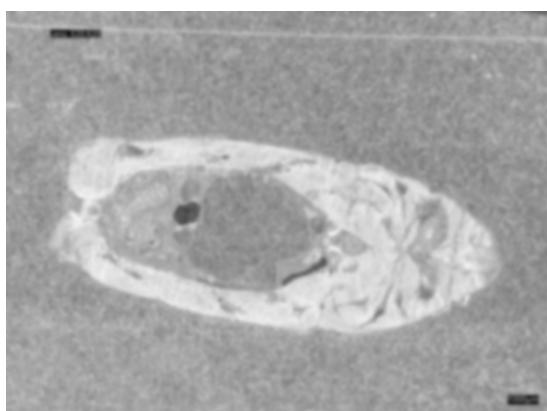


Figure 4.18 blurred k=61



Figure 4.19 sobel k=61



Figure 4.22 laplacian

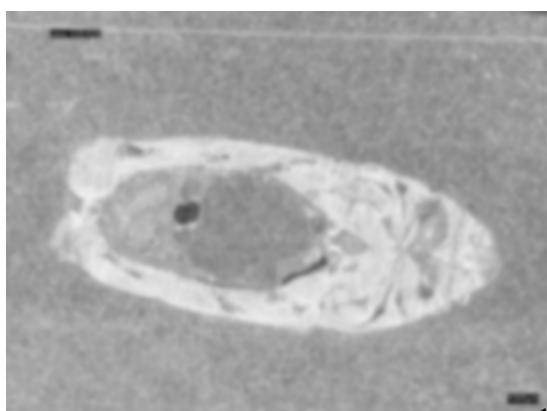


Figure 4.20 blurred k=81

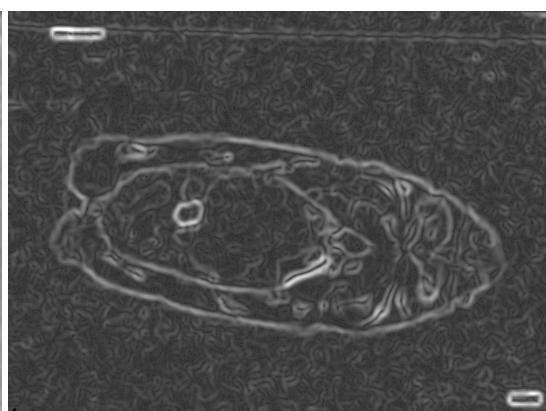


Figure 4.21 sobel k=81

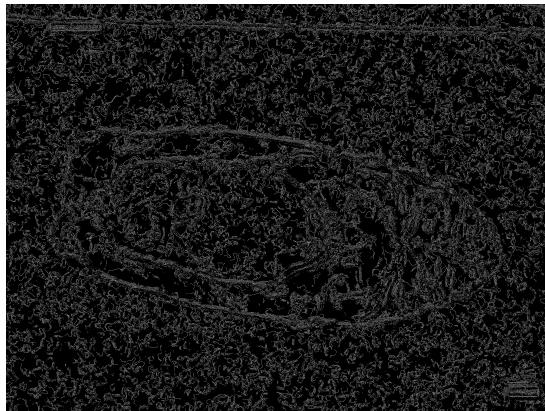


Figure 4.23 canny 2 5

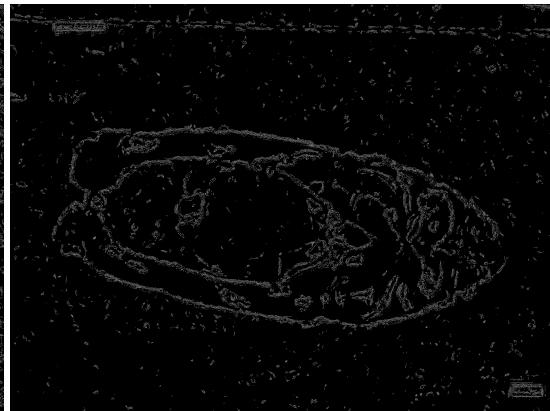


Figure 4.24 canny 4 10

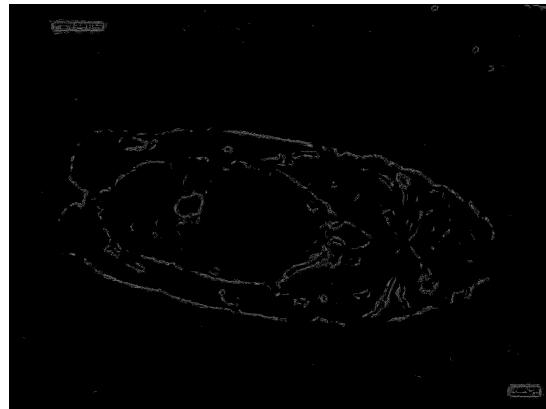


Figure 4.25 canny 6 15

canny 算法相对于 sobel 算法稍显复杂-引入了阈值检测，非极大值抑制等步骤。canny 算法引入了两个阈值，分别为低阈值和高阈值。其中，当图像的梯度值大于高阈值时，被认为是边缘；当图像的梯度值小于低阈值时，被认为不是边缘；当图像的梯度值在两者之间时，如果与高阈值的边缘相连，则被认为是边缘，否则被认为不是边缘。这样的处理可以有效的去除图像中的噪声，得到更加准确的边缘检测结果。

通常情况下，高阈值和低阈值的比值在 2:1 到 3:1 之间。在这里我们选择阈值比为 2.5 : 1，探究不同阈值对边缘检测的影响。

取低阈值为 2.4.6，此时对应的高阈值为 5.10.15。canny 算法的结果如下所示。

在三张 canny 算法的结果中，可见Figure 4.24的效果最好，其能在保证边缘细节得到大部分保留的情况下，去除了大部分的噪声。因此我们选择 canny 算法的阈值为 4.10。

总结

对比 sobel, laplacian 和 canny 算法的结果，sobel 算法的效果一般，对于底噪不是能很好的去除，边缘检测效果还算显著。laplacian 算法最差，边缘甚至已经不可见，这可能是因为该算法对噪声最敏感。canny 算法的效果最好，能够在保证边缘细节的情况下，去除大部分的噪声。因此我们选择 canny 算法作为图像预处理的方法。

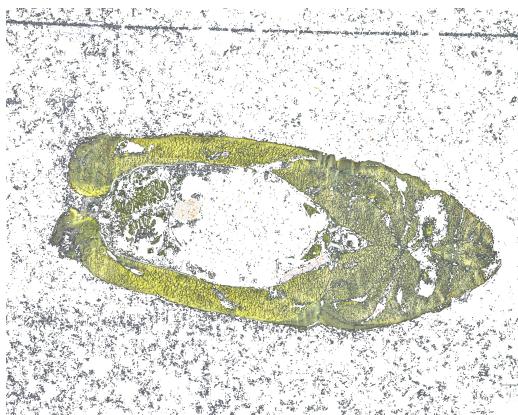


Figure 4.28 final

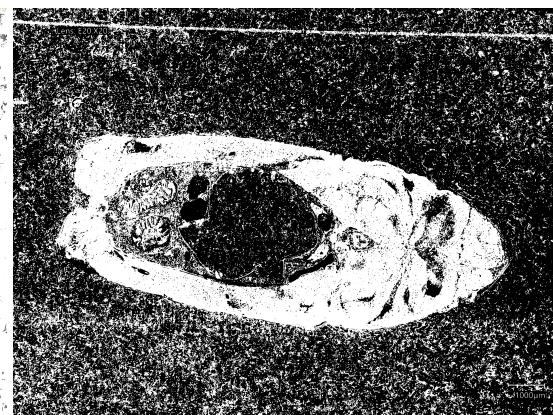


Figure 4.29 fingerprint

4.4.2 阈值分割

考虑到生物组织样本的主体是黄色，石蜡是白色，我们可以通过设置一个阈值，将图像中的白色部分分割出来，那么剩下的就是生物组织部分。在这里使用 python 的 opencv 库中进行操作。首先将图像进行对比度增强，增加饱和度，更好的凸显出生物组织的颜色 (Figure 4.26)。之后读取图像的每个像素点，将黄色周围半径 15 左右的像素点进行保留（约为图像宽的百分之一），其他的色块进行去除。(如Figure 4.27)。

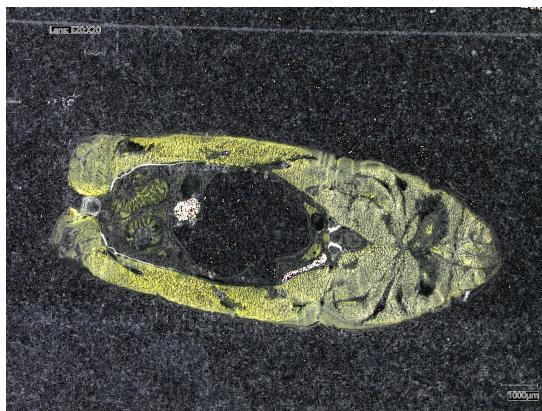


Figure 4.26 enhanced image

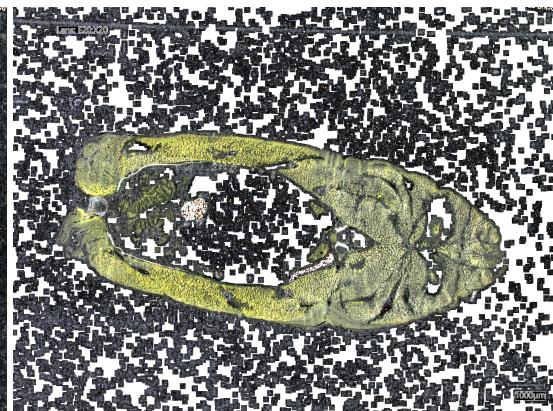


Figure 4.27 yellow picture

但是观察发现，这种方法对于生物组织和石蜡的分割效果并不好，因为生物组织在切片过程中会掉落碎片组织，出现在标本各处，进而影响黄色像素点的识别。此时还需要进一步的处理，去除黑色色块。此时只需要将黑色色块进行掩码反转，使其变为白色即可。结果如图Figure 4.28所示。

4.4.3 另一种阈值分割方法-指纹算法

在进行文献综述的时候，发现有一篇论文是基于 otsu 算法改进的分割方法用于进行指纹分割。考虑到切片样本和指纹都属于生物组织，因此我们尝试使用论文中提到的算法进行分割。结果如图Figure 4.29所示。

4.4.4 小结

根据上文提到的图像预处理方法，我们可以看到，边缘检测和阈值分割的效果都不错，都能够很好的突出生物组织的特征，去除石蜡的干扰。对此我们可以设置三组数据集，分别是经过边缘检测后的图像，经过阈值分割后的图像和经过指纹算法分割后的图像。这三组数据集将作为我们的训练集，用于下一节的模型训练。

4.5 模型 2：预处理图像 + 简单的 *cnn* 网络

在这里基础模型选用在上一节模型 1 中表现最好的模型 1c。在这里我们将模型 1c 的输入改为经过预处理后的图像，即经过边缘检测后，阈值分割和指纹算法分割后的图像。在这里模型的架构不变，只是输入的数据发生了变化。

所有的模型 2 采用和模型 1c 同样的架构构成，分别由三个包含 32 个特征图，卷积核为 3*3 的卷积层和最大池化层，一个包含 256 个神经元的全连接层组成。模型 2a 的输入为经过 **canny** 边缘检测后的图像。模型 2b 采输入为经过阈值分割后的图像。模型 2c 输入为经过指纹算法分割后的图像。

结果如下：

4.5.1 小结

在图中，对比了模型 2a、2b 和 2c 的训练和验证准确度以及损失的变化情况。模型 2a 和模型 2c 的训练和验证准确度在经过约 8 个训练周期后开始趋于稳定，其中训练准确度接近于 100%，而验证准确度稳定在 65% 和 75% 左右。尽管准确度较高，两者的验证损失仍旧较高，都在 1 以上。这可能指示了模型对训练数据过拟合，而对未见数据的泛化能力有限。

对于模型 2b，其在约 10 个训练周期后开始收敛，与模型 2a 和 2c 相比，模型 2b 拥有最高的的验证准确度，约为 82% 左右，但是其损失显著低于模型 2a 和 2c，在 1-1.2 波动。这表明模型 2b 在泛化到验证集上时的性能更优，损失更低，反映了模型更好的适应性和鲁棒性。

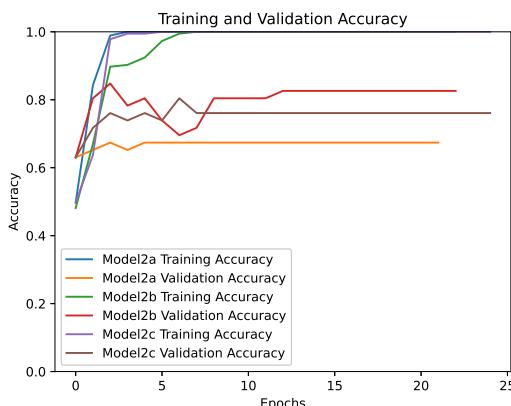


Figure 4.30 Model 2 accuracy

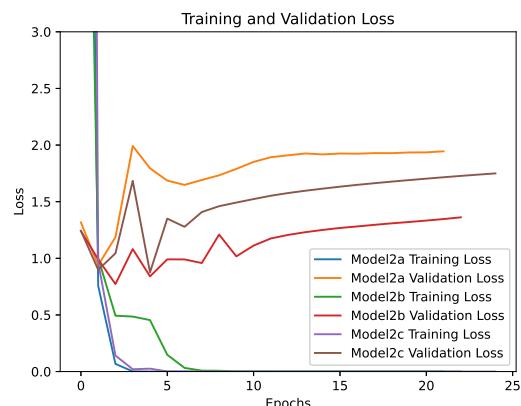


Figure 4.31 Model 2 loss



Figure 4.32 origin

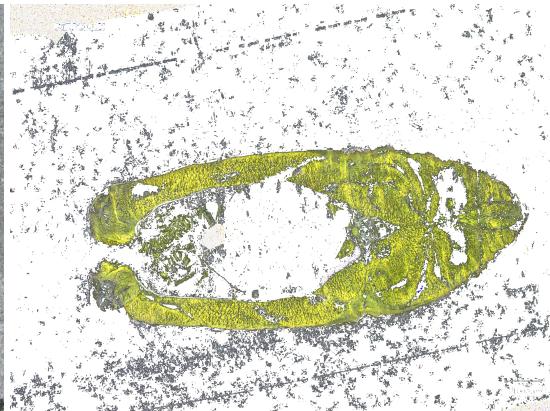


Figure 4.33 yellow

可能原因是模型 2a 和 2c 可能处理的是灰度图像，而模型 2b 处理的是彩色图像。彩色图像包含的 RGB 通道信息可以提供更丰富的特征，从而可能增强了模型的特征提取和泛化能力。然而，即使彩色图像提供了额外信息，前处理步骤，尤其是模型 2b 的阈值分割，可能会导致重要细节的丢失，这反过来可能会影响到模型在特定图像上的表现。这种情况下，模型的预处理步骤需要仔细检查，以确保不会因过于激进的图像简化而丢失关键信息。

一个丢失关键信息的例子如下所示：

Figure 4.33 是模型 2b 训练集（经过黄色阈值分割后的图像）中的一张图片，对比原图（Figure 4.32）可以观察发现，原本切片中能够被接受的水平褶皱瑕疵被阈值分割算法显著增强了，这有可能会影响模型的训练效果，即模型会在一定程度上与 horizontal line 混淆。

由此可以看出，对于图像预处理，其实并不能显著的提高模型的训练效果，反而可能会因为过于激进的预处理而丢失关键信息，导致模型的训练效果下降。在后面将会尝试使用迁移学习的方法，使用预训练好的大规模深度学习模型，将其迁移到我们的数据集上，以提高模型的训练效果。

4.6 模型 3：原始图像 + 迁移学习

现在我们已经尝试过了简单的 CNN 网络，以及对图像进行预处理后的 CNN 网络。既然训练结果不是很理想，那我们为什么不去尝试更大更深的模型？在这一节，我们尝试使用迁移学习的方法，使用预训练好的大规模深度学习模型，将其迁移到我们的数据集上，以提高模型的训练效果。

正如在第三节 methodology 里提到的，在这里将使用 VGG16, VGG19 和 InceptionV3 三个模型进行迁移学习。这三个模型都是在 ImageNet 数据集上训练好的模型，具有已经训练好的权重。在这里为了避免迁移学习过拟合，不仅使用了原有的早停法，还限制了模型的学习率为 $1e-5$ （对于 InceptionV3 模型，学习率为 $1e-4$ ）。

model3a 是使用 VGG16 模型进行迁移学习的模型。model3b 是使用 VGG19 模型进行迁移学习的模型，VGG19 和 VGG16 相比只是在中间增加了 3 个额外的卷积层，其他则与 VGG16 相同。model3c 是使用 InceptionV3 模型进行迁移学习的模型，其中 InceptionV3 是一个相对于 VGG16

和 VGG19 更加复杂的模型，其在训练过程中引入了 Inception 模块，能够更好的提取图像的特征。

在这里统一将模型的输入调整为 224*224，因为 VGG16 和 VGG19 模型在预训练时的输入层是 224*224 的图像，而 InceptionV3 的默认值为 299*299 的图像。

并且，在基础模型后面还需要添加一个全局平均池化层，一个全连接层。全局平均池化层的作用是将每个特征图减小到一个单一的数值，全连接层的作用是将全局平均池化层的输出转换为我们需要的输出，其输出节点的个数则和分类的数量相等。

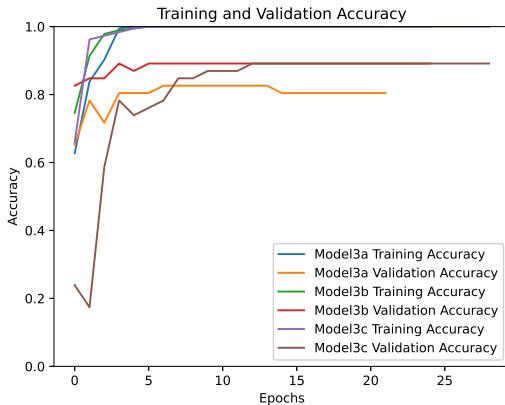


Figure 4.34 Model 3 accuracy

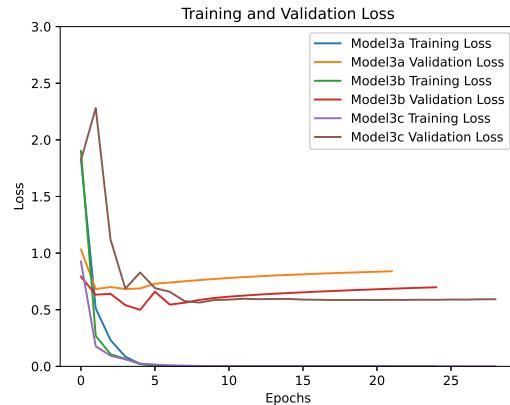


Figure 4.35 Model 3 loss

对比三个模型，可以观察得到 model3b 和 model3c 的验证集的准确度显著高于 model3a，约为 90% 左右。观察损失图，可以得出 model3c 在这三个模型中验证集的损失是最低的，model3b 次之，model3a 表现最差。这可能是因为 InceptionV3 模型的复杂度更高，能够更好的提取图像的特征。而 VGG16 和 VGG19 模型相对于 InceptionV3 模型而言，更加简单，可能在提取图像特征上存在一定的局限性。此外，model3b 性能显著高于 model3a 可以说明，VGG19 模型相对于 VGG16 模型而言，多出的三个卷积层能够更好的提取图像的特征。符合模型的复杂度越高，其训练效果越好的规律。

4.6.1 小结

对比 VGG16，VGG19 和 InceptionV3 三个模型，可以发现 InceptionV3 的训练效果最好，其训练准确度和验证准确度收敛于 1 和 0.9 左右，损失收敛于 0.6 左右。这说明 InceptionV3 模型的训练效果最好，其泛化能力最强。

4.7 模型选择总结

横向对比模型系列，模型 1，模型 2 和模型 3，可以发现模型 3 的训练效果最好。特别是模型 3c。究其原因，可能是因为模型系列 3 是基于大规模图像识别的超深卷积网络，其在训练过程中能够更好的提取图像的特征，构建自己的特征空间。值得注意的是，模型 3c，属于 InceptionV3，在架构上具有模块化的设计，包括了多个“inception 模块”。其包含了多尺度的卷积层，同时在

同一层内并行运行。在特征提取上，Inception 模块可以在同一层内捕捉不同尺度的特征，使得网络能够自适应地选择更合适的特征表示。在处理深度上，InceptionV3 利用批量标准化和残差连接来帮助训练深层网络，可以显著解决梯度消失的问题。

因此，我们选择模型 3c 作为我们的最终模型，用于后续的进一步应用和测试。

5 Presentation of experimental or analytical results/descriptions of final constructed product

在这一节我们主要讨论模型的测试结果和模型进一步改进的空间。

5.1 带入测试集验证准确度

在这里我们将训练好的模型应用到额外准备好的测试集上，计算模型的准确度。

Figure 5.1 是模型在测试集上的准确度：（准确度定义为标签与模型预测一致的样本数占总样本数的比例）

Table 5.1 Model accuracy on test set

Category	Accuracy(%)
Normal	98.4
Horizontal Line	95.6
Vertical Line	80.0
Slope	96.1
Other	95.2

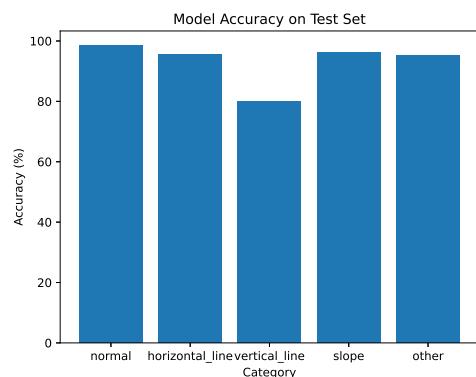


Figure 5.1 Model Accuracy on Test Set

可见，模型预测结果在 normal 上表现较好，而在 vertical_line 上表现较差。这可能是由于用于测试的样本量较少，导致模型学习不足，难以判断。

5.2 模型的进一步提高（改变输入分辨率）

在这里我们讨论模型的进一步提高的空间。

将高分辨率图片缩放为 InceptionV3 模型默认的 299x299 大小的确可能导致信息和细节的丢失，特别是对于原始分辨率远高于此标准的图像。例如，从 VHX7000 设备采集的 2880x2160 分辨率图像就含有大量的细节，直接缩放可能不利于模型捕捉到所有的细微差别，尤其是在医学影像或其他细节丰富的领域。

改变模型的输入层接受更大的图片尺寸是一个潜在的解决方案。这样做的优势是它允许模型处理更高分辨率的图像，保留更多的原始信息和细节，可能导致更好的性能和更高的准确度。此

Table 5.2 model accuracy on test set

	normal	horizontal_line	vertical_line	slope	other
accuracy(%)	98.4	96.7	85.6	96.5	96.5

外 InceptionV3 的架构设计有助于处理更大图片，因为其含有多个大小不一的卷积核，这使得它能够捕捉不同尺度的特征。

受制于实验室机器性能(显存为 16G)，在这里将图像缩放到分别为原图像的 0.4 倍，即 1152*864，进行再一次训练。

新的模型为 model4，训练效果如下所示：

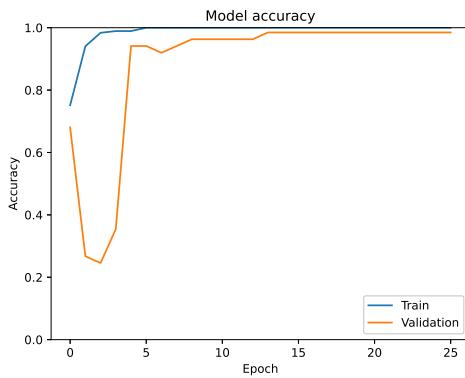


Figure 5.2 Model-4 accuracy

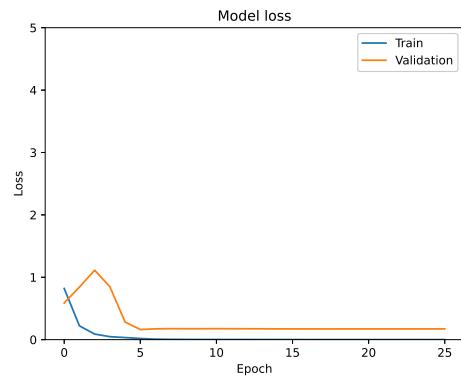


Figure 5.3 Model-4 loss

观察训练准确度和损失随步长的变化可以发现，模型的性能有惊人的明显提升。训练和验证准确度都接近 1，同时验证损失降至 15% 左右，这通常表明模型具有很强的泛化能力。这种情况下，模型不仅在训练数据上表现出色，而且能够很好地泛化到新的、未见过的数据上。

将其再一次带入测试集进行准确度评估，结果如Table 5.2：

对比修改分辨率前后的模型准确度，可以发现模型的准确度虽有提升但并不显著的，可能是由于准确度已经很接近于 1，提升的空间较小导致的。

5.3 探究机器的最佳切削角

使用预先准备好的各个切削角度的图片，从 8-12 度，每 0.5 度一个样本，一共 9 组数据，每组数据包含 100 张图片，使用模型 4 来评估每组的良品率。此时，找到良品率最高的数据组，即为该机器的最佳切削角度。

由Figure 5.3可知，最佳切削角度为 10 度。

另外，从Figure 5.4得出若要在保证切削质量为百分之 80 的情况下，切削角度应在 9 度到 10.5 度之间。

Table 5.3 Normal accuracy on different angle

Angle	Accuracy(%)
8	80
8.5	81.5
9	83.5
9.5	93.3
10	96.6
10.5	88.8
11	84.2
11.5	66.6
12	62.2

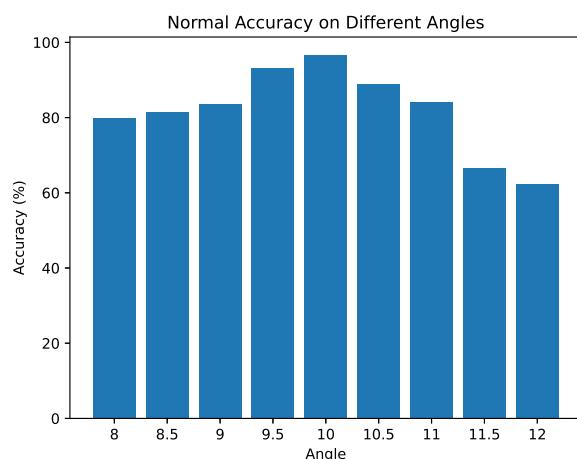


Figure 5.4 Model Accuracy on Different Angle

5.4 模型通用性

在上文的实验中我们采用的是鱼的卵巢为组织切片，而在实际应用中，我们可能会遇到其他组织切片，如其他器官或其他动物标本等。因此，我们需要考虑模型的通用性。

在这里有另一组已经采集好的数据集，是鱼的肺切片，在这一共将其分成 4 类，分别是 good, normal, bad,other 四类（样本如Figure 5.5到Figure 5.8）。现在更改输入数据集，保持原有 model4 架构不变，采用 1152*864 分辨率图片作为输入数据进行训练。

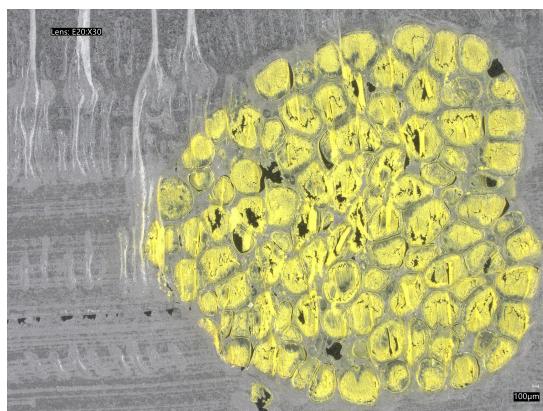


Figure 5.5 good fish lung

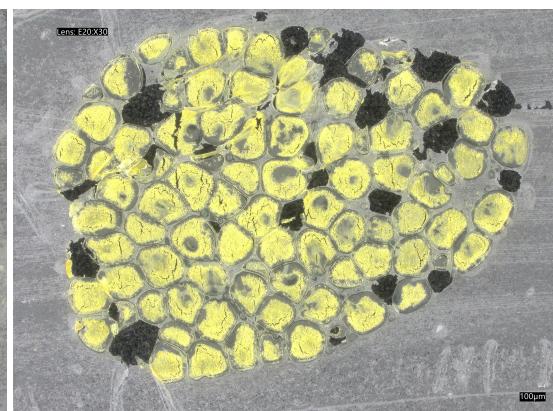


Figure 5.6 normal fish lung

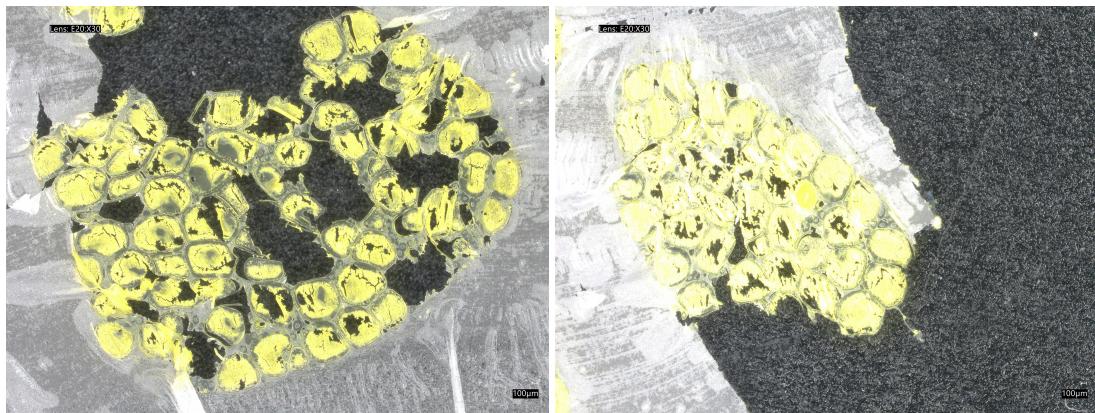


Figure 5.7 bad fish lung

Figure 5.8 other fish lung

训练的准确度和损失如Figure 5.9和Figure 5.10所示。

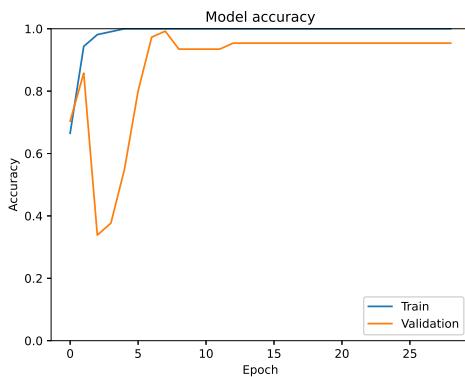


Figure 5.9 Model-5 accuracy

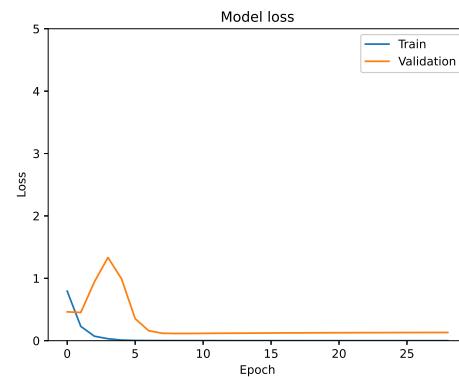


Figure 5.10 Model-5 loss

通过观察图片可以得出：模型 5 的训练和验证准确度迅速上升并保持在高位，表明模型在这两个数据集上均有良好表现，损失图显示训练损失快速降低并趋于零，而验证损失在初始阶段出现尖峰后迅速降低并稳定，整体来看，这些迹象表明模型具有较好的拟合能力和泛化性能。

将其带入测试集进行测试，结果如Figure 5.11所示：

Table 5.4 Model accuracy on test set

label	accuracy(%)
bad	94.1
good	98.2
normal	94.7
other	95.0

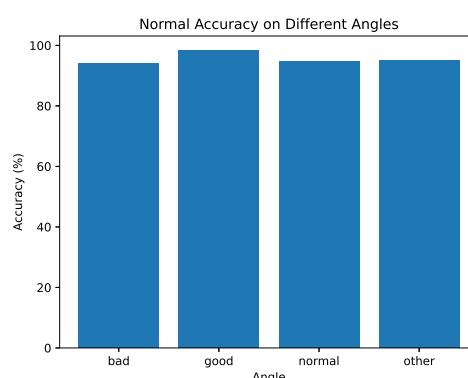


Figure 5.11 Model Accuracy on Test Set

可见，模型预测结果在全部的标签中均有 90% 以上的准确度，表现非常理想。这说明模型具有较好的通用性，可以应用到其他组织切片的分类中。

6 Discussion and conclusions

6.1 Discussion of results

正如上文中所详述，这里将详细讨论研究过程中的发现和结果。

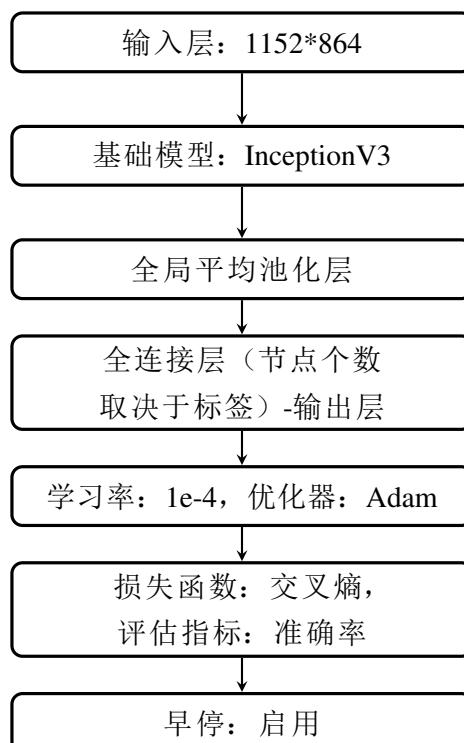
在第 4 节中，我们试图建立模型，尝试了从简单的 CNN 模型开始，之后进行图像的预处理，发现效果都不理想的情况下尝试了迁移学习的方法，得出采用 InceptionV3 模型的迁移学习效果最好。

在这其中值得注意的是伴随着不同模型的尝试，当模型参数调整或模型架构变得更加复杂时（例如 InceptionV3），模型性能显著提高，即验证集的准确度越来越高，损失越来越低。

此外，对比模型系列 1 和 2，我们发现在在图像分类的任务中使用预处理图像的手段去辅助机器提取特征的行为不是非常有效。采用图像处理有可能会导致重要细节及信息的丢失，进而影响机器对特征的提取，从而影响模型的准确度和性能。

在第 5 节中，我们对模型进行了应用测试，首先选取了额外的测试集测试模型的准确度，发现模型在所有测试集上的准确度均大于 85%。之后带入使用模型对不同切削角进行评估，发现若要在保证切削质量为百分之 80 的情况下，切削角度应在 9 度到 10.5 度之间。最后，我们使用了另外的鱼的肺泡切片的图像数据集对模型进行二次验证，发现模型对于测试集的标签预测准确率均在 90% 以上，反应模型能够很好地应用于其他数据集。

最终的采取的模型如下：



6.2 Future work

分类方法的改进

本研究提供了多个改进和深化的路径。尤其是，所采用的分类方法已经展现出理想的结果；然而，不足的是，五个分类仍旧是不充足的。通过提升分类的多样性，可以进一步得到切削角度和样本良品率的关系，进而提供更精确的分析。

此外，在拥有足够多的分类点的情况下，可以考虑从分类框架向线性分析方法过渡。在分类种类足够多的情况下，可以近似认为这是线性关系下的离散点-即可以通过线性拟合来得到最佳切削角度和样本良品率的线性关系。

在统计和数据科学中，当分类问题能够被线性化（及分类的标签能够被离散化，如上文中的切削角度）的时候-通常将其称之为线性判别分析（Linear discriminant analysis）。

在模型显示出显著能力和鲁棒性的情况下，采用线性判别分析——特别是在确定最优切割角度方面——可能呈现出一种更精细的方法，用于关联组织质量与切割参数。这样的方法有可能简化切割参数的预测准确性，并有助于更精确地控制组织切割过程。

当然，将分类问题转变为线性判别分析问题的难度是非常大的。其中一个关键的问题是，二分类问题的模型预测输出结果是概率值，而线性回归问题的本质是找到切削角度和切片质量的关系，即自变量是切削角度，因变量是切削质量。此外，显然切削角度和切片质量也不是简单的一元线性关系，模型需要处理的数据将会非常复杂。因此，这一方面的改进需要更多的理论和实践的探索。

此外，线性模型的训练和验证需要巨量的数据（甚至高出几个数量级），而这些数据采集和收集将会是一个非常漫长且困难的过程。不仅如此，线性回归模型的训练和验证也需要更多的计算资源。目前在采用 tensorflow 框架下选择 InceptionV3 模型，输入分辨率为 1152*864 的情况下，显卡的显存已经达到了极限，因此在这一方面的改进需要更多的硬件和更加强大的计算资源。所以，这一方面的改进是一个长期的目标，需要更多的资源和时间。

关于线性判别分析的研究，则是一个更加深入的方向，需要更多的理论和实践的探索。比如 Jie Wen 提到的 Robust Sparse Linear Discriminant Analysis[21]. 该方法在线性判别分析的基础上，引入了稀疏性，使得模型更加稳健。

性能提升与优化

随着本研究目标向大规模应用前进，性能提升成为了必须面临的重要挑战。性能优化不仅仅局限于算法效率的提升，还包括模型框架的扩展性、稳定性和部署能力的增强以及底层的语言和代码优化。

显然，针对计算资源的利用和调度策略，我们应当寻求更高效的计算框架和并行处理算法。例如，利用分布式计算资源，可以显著缩短模型训练的时间，并提高处理大型数据集时的效率。同时，考虑到能耗和计算成本的约束，优化模型的计算结构和参数设置，以期达到在有限资源下最大化计算输出的目的。

《Analysis of the Application Efficiency of TensorFlow and PyTorch in Convolutional Neural Network》这篇文章总结了两种主流框架 tensorflow 和 pytorch 在卷积神经网络中的区别，其中 tensorflow 具有更低的误差率和更小的收敛步长，而 pytorch 具有更快的训练速度 [22]。

Pascal Fua 在《Comparing Python, Go, and C++ on the N-Queens Problem》一文中给出了一种优化深度学习性能的方法。通过对解释性语言 Python、编译型语言 go 和 C++ 在 N 皇后问题上的性能，发现 runtime 语言在处理循环和数据流时具有明显优势，因此在深度学习中，一个提升性能的方法就是使用如 numba 编译器与 cython, pybind 进行编译 [23]。

切片流程的优化

在我们的研究中，我们还发现，如果能在切片过程中实时评估切削质量，并根据评估结果调整切削参数，将会在流程上显著提升切片质量。

具体的反馈调节流程包括，通过在切片机器上方安装摄像头对从刀片上切削下来的样本进行数据采集，然后将照片输入模型进行实时评估，然后根据评估结果控制机器的切削的给进速度和角度等参数，改进下一个切片样本的质量，实现样本质量的可控性和保证样本质量。

显然，如果要达成这一切仍然是个挑战。首先，需要一个清晰的相机和高效的实时图像处理系统，能够对图像数据进行采集；然后需要一个预训练好的模型和具有强大性能的电脑，能够快速内对图像进行评估并且给出结果，并根据结果确定需要给机器提供修改的参数；最后需要一个高效的控制接口，能够保证修改后的参数能够及时传递给切片机器。最后，这整个系统对时间的要求也是非常高的，需要在两次切削之间内完成所有的操作。

一个很好的例子就是《Convolutional neural networks applied to microtomy: Identifying the trimming-end cutting routine on paraffin-embedded tissue blocks》。在这篇文章当中，作者提出了一种使用 cnn 网络对组织切片进行识别并进行修整的方法，通过在切片机器上安装摄像头，对切片过程进行实时监控，然后将图像输入 cnn 网络进行识别，最后根据识别结果调整切片机器的参数，实现了切片过程的自动化 [24]。通过将切片机，相机，深度学习模型等独立部分联系在一起，为我们在切削过程中实时评估切削质量和修改切削参数提供一个可行的解决方案。

6.3 *Conclusions*

这项调查研究使我们对生物医学组织切片机的活检参数优化有了重要的洞见。我们对各种切割角度的专注实验，结合深度学习技术的严格应用，不仅提高了我们对组织采样的理解，而且引发了未来如何处理此类任务的范式转变。

从全面的实验工作和分析调查中可以明显看出，通过对复杂的卷积神经网络的迁移学习，产生了一个能够以高精度评估组织样本质量的强大框架。通过将模型应用于各种组织类型，进一步证实了其适应性，这突显了其在组织病理学领域作为一种多功能工具的广泛适用性和潜力。

此外，在寻找最佳模型的过程中，还探究了图像属性和模型复杂性之间的关系。我们发现，将图像进行一定程度的预处理后作为输入层可能会无意中丢弃重要的细节和信息，这说明图像预处理是可以被省略的。此外，还发现通过调整输入层为更高分辨率的图像，模型的准确性和预测性能显著提高，这说明保持原始数据的完整性的重要性。

总的来说，这个项目的结果不仅强化了深度学习在生物医学器械应用中的不可或缺性，而且为继续探索和创新优化组织切片技术奠定了基础，最终有助于推进生物切片技术的改良。

7 Project management, consideration of sustainability and health and safety

7.1 *Subsection 5.1*

7.2 *Subsection 5.2*

References

- [1] Murphy, K.P., 2012. Machine learning: a probabilistic perspective. Cambridge: The MIT Press, pp.32.
- [2] Zimmermann, M., Lampe, J., Lange, S., Smirnow, I., Königsrainer, A., Hann-von-Weyhern, C., Fend, F., Gregor, M., Bitzer, M. & Lauer, U.M. (2009). Improved reproducibility in preparing precision-cut liver tissue slices. *Cytotechnology*, 61(3), 145-152. <https://doi.org/10.1007/s10616-009-9246-4>
- [3] Klimuszko, E., Orywal, K., Sierpinska, T. et al. (2018) 'Evaluation of calcium and magnesium contents in tooth enamel without any pathological changes: in vitro preliminary study', *Odontology*, 106(4), pp. 369-376. <https://doi.org/10.1007/s10266-018-0353-6>
- [4] Hrzenjak, A., Moinfar, F., Tavassoli, F.A., Strohmeier, B., Kremser, M.-L., Zatloukal, K. and Denk, H. (2005) 'JAZF1/JJAZ1 gene fusion in endometrial stromal sarcomas: molecular analysis by reverse transcriptase-polymerase chain reaction optimized for paraffin-embedded tissue', *The Journal of Molecular Diagnostics*, 7(3), pp. 388-395. [https://doi.org/10.1016/S1525-1578\(10\)60568-5](https://doi.org/10.1016/S1525-1578(10)60568-5)
- [5] Song, L., Mino, M., Yamak, J., Nguyen, V., Lopez, D., Pham, V., Fazelpour, A., Le, V., Fu, D., Tippin, M., Uchio, E. and Zi, X. (2022) 'Flavokawain A reduces tumor-initiating properties and stemness of prostate cancer', *Frontiers in Oncology*, 12. <https://doi.org/10.3389/fonc.2022.943846>
- [6] Azuma, T., Murata, Y., Hokii, Y., Akiyama, S. and Shinozaki, Y. (2023) 'Evaluation of micro-cracks formed by grinding in machinable lithium silicate', *Dental Materials*, 39, p. e10. Abstracts of the Academy of Dental Materials, Annual Meeting, Athens, Greece, 2022. <https://doi.org/10.1016/j.dental.2023.08.022>
- [7] Veer, F.A. (2022) 'Looking at the foundations of structural glass with a digital microscope', in *Structures and Architecture. A Viable Urban Perspective*, 1st ed. CRC Press, pp. 7. eBook ISBN: 9781003023555.

- [8] Guachi-Guachi, L., Ruspi, J., Scarlino, P., Poliziani, A., Ciancia, S., Lunni, D., Baldi, G., Cavazzana, A., Zucca, A., Bellini, M., Pedrazzini, G.A., Ciuti, G., Controzzi, M., Vannozzi, L. and Ricotti, L. (2023) 'Convolutional neural networks applied to microtomy: Identifying the trimming-end cutting routine on paraffin-embedded tissue blocks', *Engineering Applications of Artificial Intelligence*, 126(B), p. 106963. <https://doi.org/10.1016/j.engappai.2023.106963>
- [9] Andrearczyk, V. and Whelan, P.F. (2017) 'Deep Learning in Texture Analysis and Its Application to Tissue Image Classification', in Depeursinge, A., Al-Kadi, O.S. and Mitchell, J.R. (eds.) *Biomedical Texture Analysis*. Academic Press, pp. 95-129. ISBN 9780128121337. <https://doi.org/10.1016/B978-0-12-812133-7.00004-1>
- [10] Xu, Y., Jia, Z., Wang, L.B., Ai, Y., Zhang, F., Lai, M., Chang, E.I.C., 2017. Large scale tissue histopathology image classification, segmentation, and visualization via deep convolutional activation features. *BMC Bioinformatics*, 18(1), p.281. Available at: <https://doi.org/10.1186/s12859-017-1685-x>.
- [11] Culjak, I., Abram, D., Pribanic, T., Dzapo, H., Cifrek, M., 2012. A brief introduction to OpenCV. In: 2012 Proceedings of the 35th International Convention MIPRO, Opatija, Croatia, pp. 1725-1730. Available at: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6240859&isnumber=6240598>.
- [12] Bradski, G. and Kaehler, A., 2008. *Learning OpenCV: Computer vision with the OpenCV library*. O'Reilly Media, Inc.
- [13] Canny, J., 1986. A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6), pp.679-698. doi: 10.1109/TPAMI.1986.4767851. Available at: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4767851&isnumber=4767846>.
- [14] Yue, Y. and Zhu, J., 2017. Algorithm of fingerprint extraction and implementation based on OpenCV. In: 2017 2nd International Conference on Image, Vision and Computing (ICIVC), Chengdu, China, pp. 163-167. doi: 10.1109/ICIVC.2017.7984539. Available at: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7984539&isnumber=7984442>.
- [15] Zhou, H. and Sun, Q., 2020. Research on Principle and Application of Convolutional Neural Networks. *IOP Conference Series: Earth and Environmental Science*, 440(4), p.042055. doi: 10.1088/1755-1315/440/4/042055. Available at: <https://dx.doi.org/10.1088/1755-1315/440/4/042055>.
- [16] Simonyan, K. and Zisserman, A., 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv preprint arXiv:1409.1556*. Available at: <https://arxiv.org/abs/1409.1556>.

- [17] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z., 2015. Rethinking the Inception Architecture for Computer Vision. arXiv preprint arXiv:1512.00567. Available at: <https://arxiv.org/abs/1512.00567>.
- [18] Chollet, F., 2016. Xception: Deep Learning with Depthwise Separable Convolutions. arXiv preprint arXiv:1610.02357. Available at: <https://arxiv.org/abs/1610.02357>.
- [19] Klimuszko, E., Orywal, K., Sierpinska, T. et al., 2018. Evaluation of calcium and magnesium contents in tooth enamel without any pathological changes: in vitro preliminary study. Odontology, 106, pp.369-376. doi: 10.1007/s10266-018-0353-6. Available at: <https://doi.org/10.1007/s10266-018-0353-6>.
- [20] Gedraite, E.S. and Hadad, M., 2011. Investigation on the effect of a Gaussian Blur in image filtering and segmentation. In: Proceedings ELMAR-2011, Zadar, Croatia, pp. 393-396. Available at: <https://ieeexplore.ieee.org/abstract/document/6044249>.
- [21] Wen, J. et al., 2019. Robust Sparse Linear Discriminant Analysis. IEEE Transactions on Circuits and Systems for Video Technology, 29(2), pp.390-403. doi: 10.1109/TCSVT.2018.2799214. Available at: <https://doi.org/10.1109/TCSVT.2018.2799214>.
- [22] Novac, O-C., Chirodea, M.C., Novac, C.M., Bizon, N., Oproescu, M., Stan, O.P., Gordan, C.E., 2022. Analysis of the Application Efficiency of TensorFlow and PyTorch in Convolutional Neural Network. Sensors, 22(22):8872. Available at: <https://doi.org/10.3390/s22228872>.
- [23] Fua, P. & Lis, K., 2020. Comparing Python, Go, and C++ on the N-Queens Problem. Available at: <https://arxiv.org/abs/2001.02491>.
- [24] Guachi-Guachi, L., Ruspi, J., Scarlino, P., Poliziani, A., Ciancia, S., Lunni, D., Baldi, G., Cavazzana, A., Zucca, A., Bellini, M., Pedrazzini, G.A., Ciuti, G., Controzzi, M., Vannozzi, L. & Ricotti, L., 2023. Convolutional neural networks applied to microtomy: Identifying the trimming-end cutting routine on paraffin-embedded tissue blocks. Engineering Applications of Artificial Intelligence, 126, p.106963. Available at: <https://www.sciencedirect.com/science/article/pii/S0952197623011478>.