SCHOOL OF COMPUTATION, INFORMATION AND TECHNOLOGY — INFORMATICS

TECHNISCHE UNIVERSITÄT MÜNCHEN

Master's Thesis in Biomedical Computing

Uncertainty in Bone Tumor Classification - Enchondroma vs. Atypical Cartilaginous Tumor

Inés del Val Guardiola

SCHOOL OF COMPUTATION, INFORMATION AND TECHNOLOGY — INFORMATICS

TECHNISCHE UNIVERSITÄT MÜNCHEN

Master's Thesis in Biomedical Computing

Uncertainty in Bone Tumor Classification - Enchondroma vs. Atypical Cartilaginous Tumor

Unsicherheit bei der Klassifizierung von Knochentumoren - Enchondrom vs. Atypischer Knorpeltumor

Author:	Inés del Val Guardiola
Supervisor:	Prof. Dr. Daniel Rückert
Advisor:	Florian Hinterwimmer, M. Sc.
Submission Date:	15.12.2023

I confirm that this master's thesis is my own work and I have documented all sources and material used.

Munich, 15.12.2023 Inés del Val Guardiola

Acknowledgments

I would like to pay special regards to Prof. Dr. Daniel Rückert for allowing me to develop this project. I also want to express my deepest gratitude to my advisor Florian Hinterwimmer for his advice, support, and guidance during my thesis. Without his dedication, passion for taking this project forward, and willingness to help me resolve all the obstacles encountered throughout the project, this work would not have been the same. In addition, I wish to thank Dr. med. Jan Neumann, Dr. med. Sarah Consalvo, and Dr. med. Christina Valle for their valuable medical feedback. Lastly, I would like to thank the Department of Orthopaedics and Sports Orthopaedics for welcoming me as a part of the team during my thesis project.

Abbreviations

ACT Atypical Cartilaginous Tumor

DL Deep Learning

XAI Explainable Artificial Intelligence

SHAP SHpley Additive exPlanations

AI Artificial Intelligence

WHO World Health Organization

CT Computed Tomography

MR Magnetic Resonance

ML Machine Learning

CV Computer Vision

CNN Convolutional Neural Network

NN Neural Network

NLP Natural Language Processing

ViT Vision Transformer

CLS Classification

MLP Multi-Layer Perceptron

LN Layer Normalization

TP True Positive

TN True Negative

FP False Positive

FN False Negative

TPR True Positive Rate

TNR True Negative Rate

FPR False Positive Rate

AUC-ROC Area Under the Receiver Operating Characteristic

Grad-CAM Gradient-weighted Class Activation Mapping

LIME Local Interpretable Model-agnostic Explanations

CSV Comma Separated Value

GPU Graphical Processing Unit

CELoss Cross-Entropy Loss

Abstract

Cartilaginous tumors, notably Enchondroma and Atypical Cartilaginous Tumor (ACT), pose a diagnostic challenge, requiring differentiation for appropriate therapeutic decisions. With invasive biopsies carrying risks and recent guidelines reevaluating their role, there is a critical need for non-invasive diagnostic tools. This study addresses the challenge of distinguishing between Enchondroma and Atypical Cartilaginous Tumor (ACT) from conventional radiographs. Deep Learning (DL) is explored as a powerful solution, filling a gap in existing research on this specific classification task. A dataset of 635 patients, with 528 diagnosed with Enchondroma and 107 with ACT, is utilized, comprising plain radiographs and relevant clinical variables. The lesions are validated through histopathology diagnosis, and the dataset includes valuable information on radiologist annotations, offering insights into the uncertainties inherent in their diagnoses. The study employs Explainable Artificial Intelligence (XAI) techniques, specifically SHapley Additive exPlanations (SHAP), to enhance model interpretability and assess the contribution of individual pixels to the model's prediction. Results showcase a Vision Transformer model's notable metrics, achieving an accuracy of 0.744, sensitivity of 0.857, and specificity of 0.455. Both original X-ray images and resized version demonstrate similar performance, with resizing offering the advantage of focusing more on the tumor area. SHAP results typically highlight pixel contributions within the tumor area. Evaluation of misclassified samples by the radiologist reveals promising accuracy, with our model correctly predicting 7 out of 10 misclassified Enchondromas and 3 out of 14 misclassified ACT cases. A clinical assessment by bone tumor experts indicates that the model often focuses on areas within the tumor that typically elude radiologists' attention. While no consistent radiological pattern aiding differentiation between the two entities is evident, experts suggest an expanded dataset could lead to more precise conclusions. With further refinements and a larger dataset, this DL-based approach holds promise as a precise tool for the initial assessment and management of patients with cartilaginous tumors.

Contents

1. Introduction

1.1. Motivation

Cartilaginous tumors represent a clinical and diagnostic challenge due to their varied radiographic presentations. Among these, Enchondroma and Atypical Cartilaginous Tumor (ACT) are two prevalent entities that require differentiation for adequate therapeutic management. While Enchondromas commonly necessitate no treatment, ACTs, with their locally aggressive nature, may require curettage. The diagnostic process traditionally relies on imaging and histopathology, but invasive biopsies, though informative, carry inherent risks and may not always be feasible. Recent medical guidelines have reconsidered the role of biopsy in the diagnostic workflow for intermediate lesions, leading to a lack of a standardized diagnostic gold standard. Accurate non-invasive diagnostic tools are thus highly sought after to improve the initial assessment and management of patients with cartilaginous tumors.

1.2. Objective

Deep Learning (DL) has emerged as a powerful tool for addressing complex medical challenges [28]. However, its application to bone tumors has been limited by factors such as low incidence rates, diverse pathology types and difficulties in data collection [41]. While studies like that of He et al. [31] have shown success in primary bone tumor classification, specific research on distinguishing between Enchondroma and ACT remains scarce. Our study aims to fill this gap, developing and validating a DL model for accurate classification, reducing reliance on invasive diagnostic procedures, and improving diagnostic pathways for patients with cartilaginous tumors. Our clinical impact lies in the distinct approach we take by integrating Explainable Artificial Intelligence (XAI) techniques to interpret the model's predictions, making our Artificial Intelligence (AI) system more transparent and understandable to humans.

1.3. Outline of this work

This work is structured in six sections, addressing the challenges and advancements in utilizing DL for the differentiation between Enchondroma and ACT. The introductory section lays the foundation by presenting the motivation, objectives, and outline of the subsequent content. A detailed literature review follows, including medical background and DL fundamental concepts. The material and methods section provides insights into the dataset, preprocessing techniques, and experimental setups. Results highlight the model baseline performance, SHAP enhancements, and comparison across different test sets. The discussion section interprets the results from clinical and mathematical perspectives, addressing limitations and considerations for generalizability. Concluding this work, the study's implications, potential improvements, and directions for future research are outlined.

2. Literature Review

2.1. Medical Background

2.1.1. Bone Tumors

When cells divide abnormally and uncontrollably, they can form a mass of tissue. This mass is called a tumor or neoplasm [3]. Several kinds of tumors can grow in bones:

1. Primary bone tumors: grow from bone tissue.
2. Secondary or metastatic tumors: develop from cancer cells in another part of the body and spread to the bone.

This study focuses on primary bone tumors, a category that represents some of the rarest neoplasms in humans, accounting for about 0.2% of all tumors in the human body [20]. Due to their rarity, only specialized centers can have enough experience in managing these neoplasms, and a multidisciplinary team approach is mandatory, to avoid errors in their diagnosis and treatment [50].

2.1.2. WHO Classification

Since 1967, the World Health Organization (WHO) classification of tumors has provided practical guidance to pathologists, radiologists, and clinicians involved in these oncologic multidisciplinary teams. Improved understanding of tumor biology, led the WHO to reclassify selected bone tumors in 2020 [56]. Table 2.1 shows the current classification, which includes eight histologic families and three categories based on the risk for local recurrence and metastasis: benign, intermediate (locally aggressive or rarely metastasizing), and malignant [33].

In this project, we center our attention on two distinct tumor types highlighted in Table 2.1: Enchondroma and ACT. Both tumors belong to the histological family of chondrogenic tumors, also known as cartilage tumors. Chondrogenic tumors, the second-largest group of bone tumors, are characterized by tumor cells producing a chondroid matrix [16]. To fully understand why the differentiation between these two entities is crucial, we will briefly outline some key properties and clinical features associated with each.

Table 2.1.: WHO classification of bone tumors and categories of their biological potential [33].

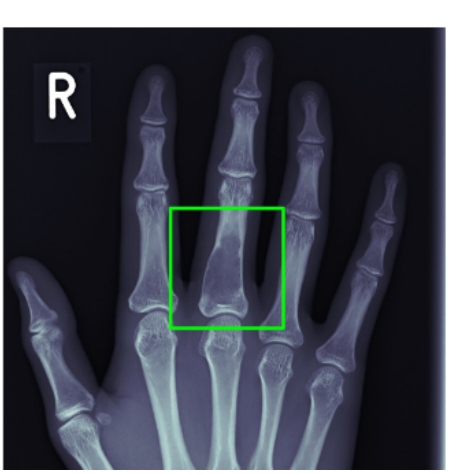
	Benign	Intermediate (locally aggressive)	Malignant
Chondrogenic	Subungual exostosis	Synovial chondromatosis	Chondrosarcoma grade 1
	Bizarre parosteal osteochondromatous proliferation	Atypical cartilaginous tumor	Chondrosarcoma grade 2
	Periosteal chondroma		Chondrosarcoma grade 3
	Enchondroma		Periosteal chondrosarcoma
	Osteochondroma		Clear cell chondrosarcoma
	Chondroblastoma		Mesenchymal chondrosarcoma
	Chondromyxoid fibroma		Dedifferentiated chondrosarcoma
	Osteochondromyxoma		

2.1.3. Enchondroma

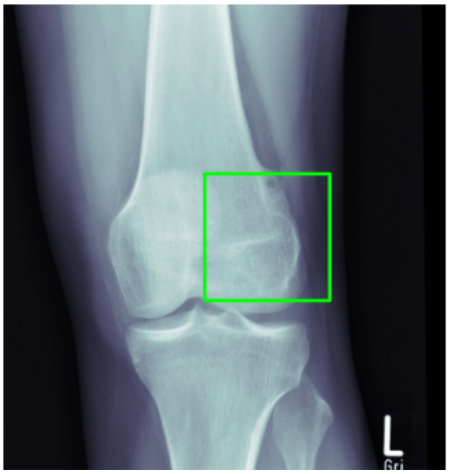
An Enchondroma is a benign hyaline cartilage neoplasm that arises within the medullary cavity of the bone. They are relatively common and account for 10-25% of all benign bone tumors [20]. While they can be diagnosed at any age, they are most frequently identified in middle-aged patients. They rarely cause pain or other symptoms, so most are diagnosed incidentally during imaging [4]. The hands and feet are the most common sites of involvement, with Enchondromas being notably prevalent as the most common bone tumor of the hand [4]. Additionally, these neoplasms may affect long tubular bones such as the proximal humerus and proximal and distal femur, while their occurrence in flat bones is exceedingly rare [16].

2.1.4. Atypical Cartilaginous Tumors (ACT)

According to the 2020 WHO classification, chondrogenic tumors in the pelvis and in the axial skeleton are referred to as chondrosarcomas grade I, while chondrogenic tumors



(a) X-ray image showing an Enchondroma located in the hand.



(b) X-ray image showing an Enchondroma located in the femur.

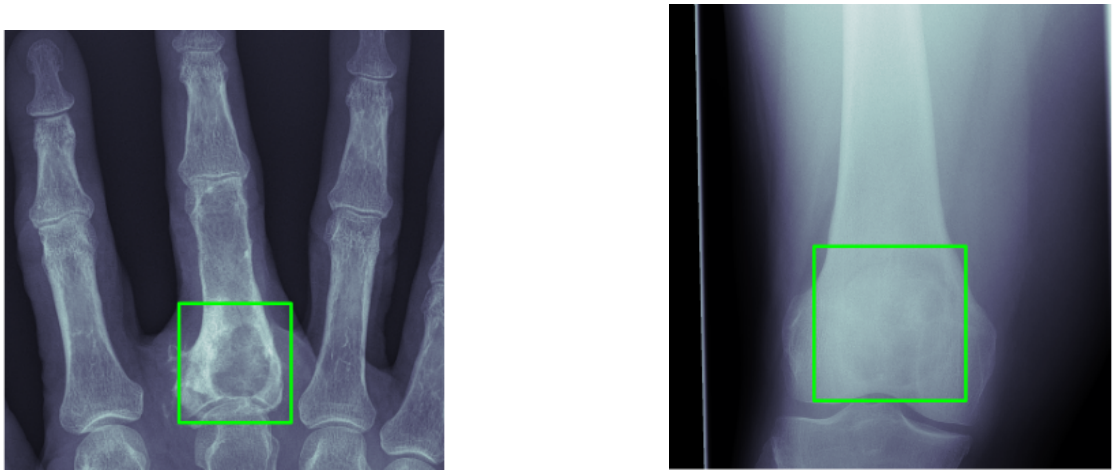
Figure 2.1.: Examples of images showing an Enchondroma, both obtained from our institution.

in the region of the extremities with the same histology are referred to as ACT, due to the more favorable prognosis [16]. ACTs are typically diagnosed in the third to sixth decades of life, affecting both sexes equally [57]. They may present with pain and swelling, but can also be asymptomatic. The most common sites of involvement are the femur, humerus, and tibia, with rare occurrences in the short tubular bones of the hands and feet [19].

2.1.5. Differential diagnosis

For adequate therapeutic management, an accurate differentiation between Enchondromas and ACTs is crucial [23]. ACTs typically require a curettage and clinical as well as imaging follow-ups. In contrast, Enchondromas in the majority of cases, do not require surgical treatment or follow-up unless symptomatic [23]. Despite intensive research efforts, the differentiation between these two entities continues to be a major diagnostic challenge [16].

In the broader context of bone tumors, combining clinical information, imaging, and histology leads to the most accurate diagnostic results [50]. The current diagnostic workflow is illustrated in Figure 2.3. First, we have a patient who arrives at the clinic with non-specific symptoms. A general practitioner with limited diagnostic tools and limited experience performs some physical examination. If there is suspicion



(a) X-ray image showing an ACT in the hand.

(b) X-ray image showing an ACT in the femur.

Figure 2.2.: Examples of images showing an ACT, both obtained from our institution.

of a tumor, the patient is referred to a specialized center where the imaging data is taken to reach a reliable diagnosis. X-ray imaging is prioritized as the initial step since it is the most helpful imaging modality when establishing the initial differential diagnosis [13]. In cases of diagnostic complexity, the next step is Computed Tomography (CT) and, if malignancy is suspected, Magnetic Resonance (MR) is taken for local staging [50]. Additionally, a bone biopsy may be indicated to confirm the cancerous or non-cancerous diagnosis of a bone tumor or investigate further an abnormality [50]. The entire diagnostic process can take several months. Providing support tools to general practitioners for immediate decision-making could significantly accelerate the diagnostic timeline, leading to an enhancement in patient care.

Before the WHO 2020 update, the final diagnosis of Enchondroma and ACT relied on a combination of clinical examination, imaging, and histology [23]. However, due to the invasive nature, associated risks, and feasibility issues of biopsy, imaging has assumed a major role in the differential diagnosis between these two subtypes of tumors [23]. X-rays continue to be the first mandatory step but nothing else is usually added [50]. Figure 2.4 illustrates the changes in imaging criteria required by the new medical guidelines. Differentiating these tumors radiographically remains challenging, particularly when located in the long bones [21]. An Enchondroma usually appears as a well-marginated lesion with no soft tissue invasion [4], while lesions over 5-6cm with cortical expansion tend to favor ACT [57].

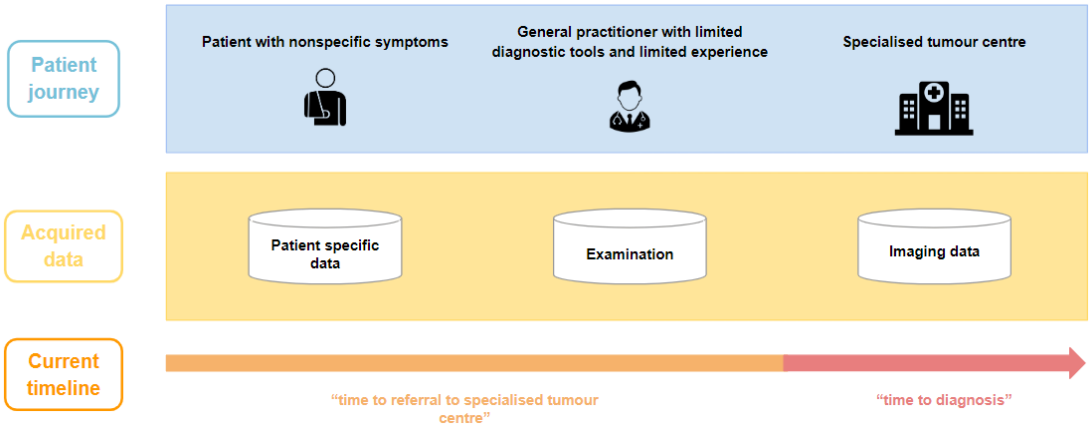


Figure 2.3.: Diagnostic workflow for bone tumors.

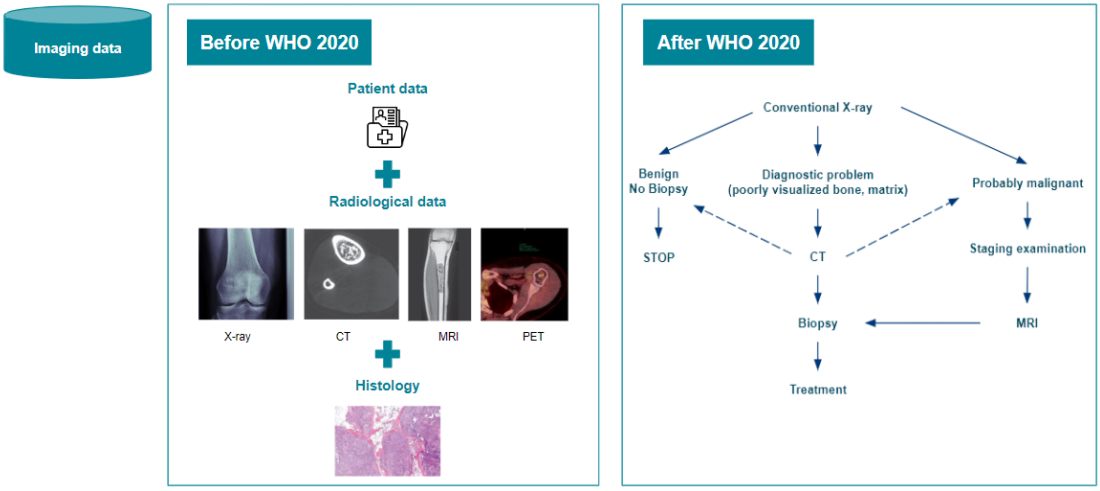


Figure 2.4.: Imaging criteria required before and after the WHO 2020 version.

2.2. Overview of Deep Learning

The term "Artificial Intelligence" was first introduced in 1956, when it was broadly referred to as "thinking machines" [7]. In simple terms, AI can be defined as the ability of a machine to learn and recognize patterns from enough representative examples and to use this information effectively for decision-making on unseen data [7]. Machine Learning (ML) is a subfield of AI and DL is the subset of ML [7]. In traditional ML techniques, most of the applied features need to be identified by a domain expert

to reduce the complexity of the data, whereas DL algorithms try to learn high-level features mimicking the learning process of the human brain. This eliminates the need for domain expertise and manual feature extraction [44]. Figure 2.5 shows how DL algorithms do not need an expert to identify features but are capable of automatic feature engineering.

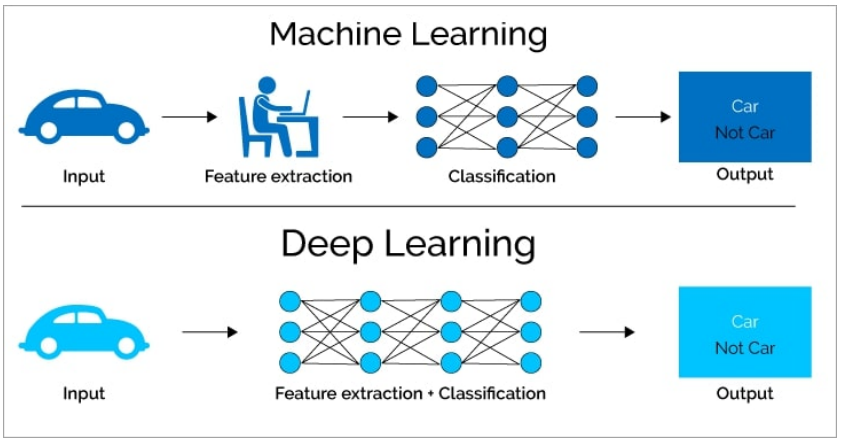


Figure 2.5.: Difference between ML and DL [44].

2.2.1. Deep Learning Basics

A significant advantage that sets DL apart from traditional ML, and greatly contributes to its popularity, is its reliance on vast datasets. The "Big Data Era" in technology opens up huge amounts of opportunities for new innovations in this field [44].

DL is mainly based on artificial neural networks. The earliest and simplest neural network is a single neuron, also called *perceptron*, which linearly combines multiple inputs, performs a nonlinear transformation, and outputs a scalar value [53], as shown in Figure 2.6. Mathematically, a perceptron can be written as a function taking a vector x as input and outputting a scalar value y where xj is the j-th dimension of x, b is the bias term, and $g()$ is an activation function, which enables that a perceptron can represent nonlinear functions. wj and b are the parameters to be learnt from the data [53]. The mathematical expression is:

$$y = g\left(\sum_{j=1}^{d} w_j x_j + b\right)$$

Multiple connected nodes/neurons form a neural network. These nodes are connected in an acyclic graph, where outputs of some neurons can become inputs to other

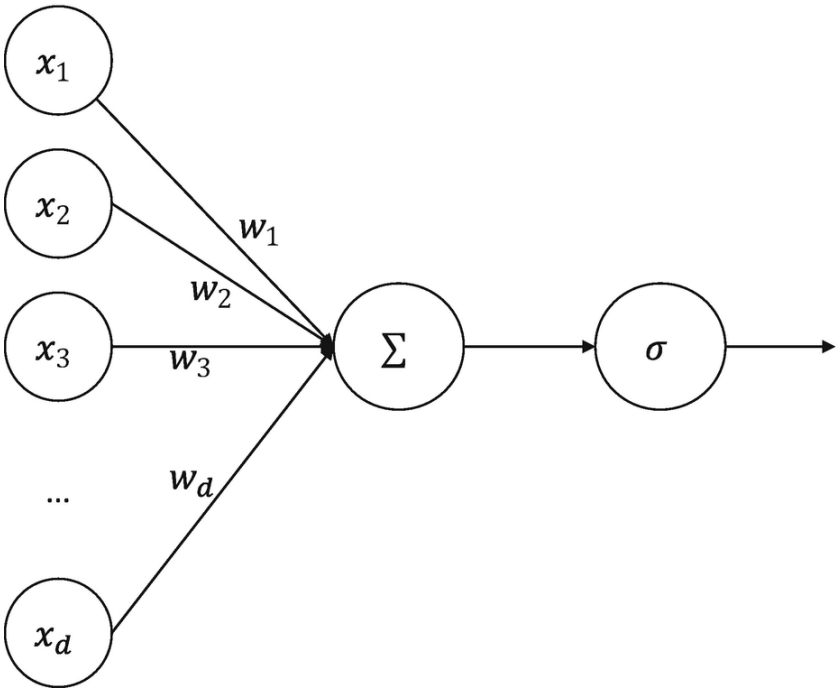


Figure 2.6.: The simplest neural network: perceptron, which was originally inspired by biological neural systems [53].

neurons [68]. Feedforward networks are the most basic neural networks, where the information flows in one direction: from the input x, through the hidden nodes, and finally to the output y [53]. Figure 2.7 shows a 3-layer neural network with three inputs, two hidden layers of 4 neurons each, and one output layer.

2.2.2. Computer Vision

DL has seen a dramatic resurgence in the past years, largely driven by increases in computational power and the availability of massive new datasets [18]. Some of the greatest successes have been in the field of Computer Vision (CV) [18]. CV is a very broad field that seeks to develop techniques to help computers *see* and understand the content of digital images [65]. The problem of CV appears simple because it is so effortless for humans. Nevertheless, it remains an unsolved problem based both on the limited understanding of how human vision works and because of the complexity of the physical world [65]. CV has seamlessly integrated into our society, finding applications in diverse fields such as healthcare and medicine. The subsequent section explores the primary challenges of applying DL in the medical domain. Then, the next

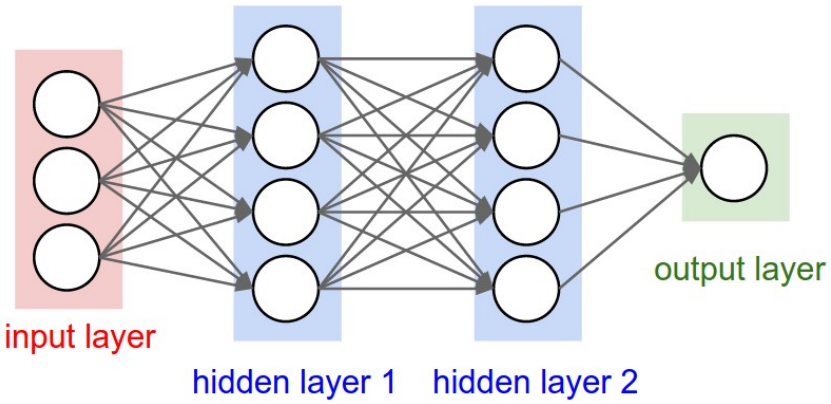


Figure 2.7.: Example of neural network that uses a stack of fully-connected layers [68].

section introduces DL models that are specially designed for image classification.

Challenges of DL in Medicine

DL has extended its applicability to various aspects of medicine such as medical image classification. Many studies have demonstrated promising results in complex diagnostics across disciplines like radiology, dermatology, ophthalmology, and pathology [18]. However, one of the main barriers to the wide adoption of DL methods in clinical practice is usually caused by the small size of medical datasets, often ranging in the hundreds or thousands of images [71]. Limited data may result in models that are less capable of capturing the underlying patterns from the training data, leading to poor generalization performance. In clinical scenarios, dataset imbalances further complicate matters, potentially biasing models toward predominant classes [10]. Additionally, the heterogeneity of medical image data, especially with X-rays, introduces challenges in learning consistent features due to variations in imaging conditions and quality. To overcome issues related to limited dataset size, transfer learning is commonly employed, enabling models to leverage knowledge from broader datasets and adapt to specific tasks [70]. However, the absence of pre-trained models on medical image datasets poses a challenge, as pre-training on unrelated datasets like ImageNet may not optimize features for medical tasks. While extensive research has been conducted to address these challenges, the need for larger medical image datasets remains paramount. Achieving more significant datasets is crucial for the development of accurate DL systems, which can effectively support healthcare professionals [18].

2.2.3. Model Architectures for Image Classification

Convolutional Neural Networks

Convolutional Neural Networks, also known as CNNs, were introduced by LeCun with the famous LeNet model architecture capable of recognizing handwritten digits [39]. CNNs are a type of Neural Network (NN) that processes data with a grid-like topology, such as an image [48]. We can interpret a digital image as a binary representation of visual data. It contains pixel values to denote the brightness and color of each pixel [48]. Typically, CNNs are composed of the following layers:

1. Convolution Layer: is the core building block of the CNN that carries the main portion of the network's computational load [48]. It requires a few components: input data, a kernel or filter, and a feature or activation map. Imagine you have a color image made up of a matrix of pixels in 3D. This means that the input has three dimensions: height, width, and depth. Now, a 3D kernel or filter with learnable parameters shared for all pixels [48], slides across the receptive field of the input. At each position, a dot product is calculated between the input pixels and the filter, and the result is fed into an output array [34]. This process, known as convolution, repeats as the filter moves through the entire image, shifting by a certain step each time, called a stride. The final output from the series of dot products is called the feature map [34]. Figure 2.8 illustrates the convolution operation.

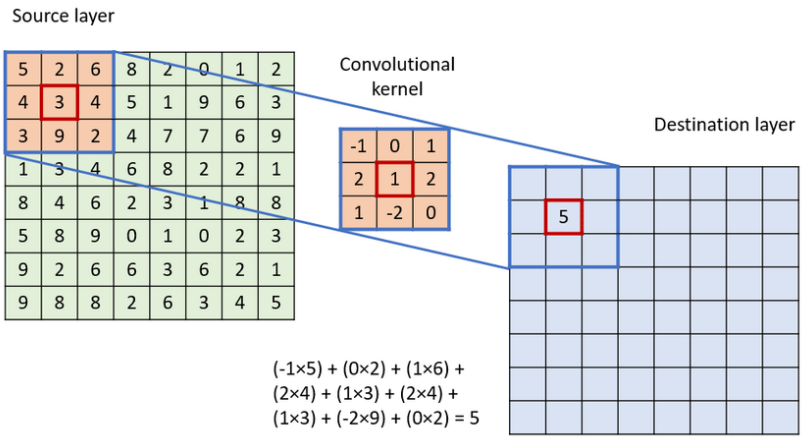
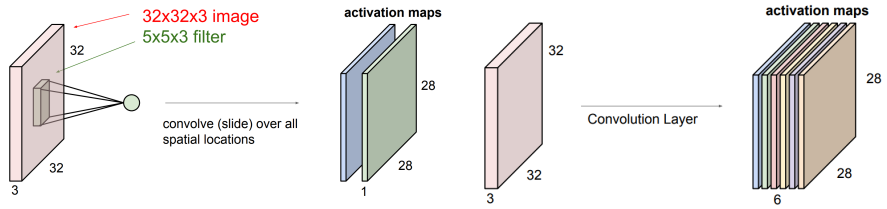


Figure 2.8.: Schematic illustration of a convolution operation [52].

There are multiple filters in a convolutional layer and each of them generates a feature map [24]. Therefore, the output of a layer is not just one map but a set of

them, all stacked together [24]. These feature maps capture different aspects or patterns in the input image, helping the network recognize various features as it processes the data.Figure 2.9 illustrates the process in a Convolution Layer.



(a) Blue and green activation maps are obtained from sliding blue and green filters respectively over all spatial locations [69].

(b) If we apply 6 5x5 filters to the image, we get 6 separate activation maps. By stacking them up we get an output of size 28x28x6 [69].

Figure 2.9.: Illustration of the Convolution Layer.

2. Pooling Layer: the convolution process generates a large amount of data, which makes it hard to train the NN [24]. A pooling layer compresses the result from the convolutional layer, making the spatial size of the representation smaller and easier to work with [69]. Besides reducing the amount of computation and the number of parameters, pooling provides translation invariance. This means the network can recognize an object, no matter where it appears in the image [48]. The pooling operation is processed on every feature map individually. The most popular operations are max pooling (picking the highest value) and average pooling (taking the average) illustrated in Figure 2.10. In the last step, the input is flattened out and and sent to a regular NN for the final classification [24].

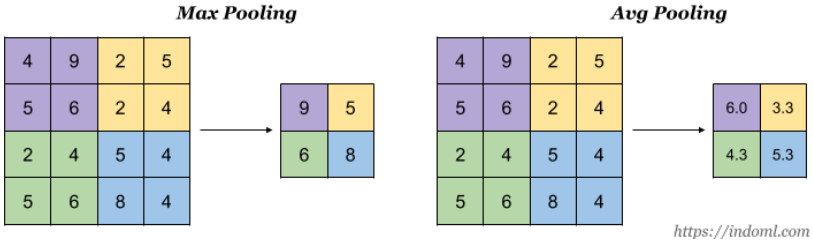


Figure 2.10.: Max pooling and average pooling operations [12].

3. Fully Connected Layer: the flattened output is fed to a feed-forward NN. This layer provides the model with the ability to understand the image: there is a flow

of information between each input pixel and each output class [24].

Figure 2.11 shows a typical CNN architecture for image classification. The first layer extracts low-level features (e.g. colors, gradient orientation, edges, etc.), while subsequent layers extract high-level features. CNNs have significantly impacted the field of medical

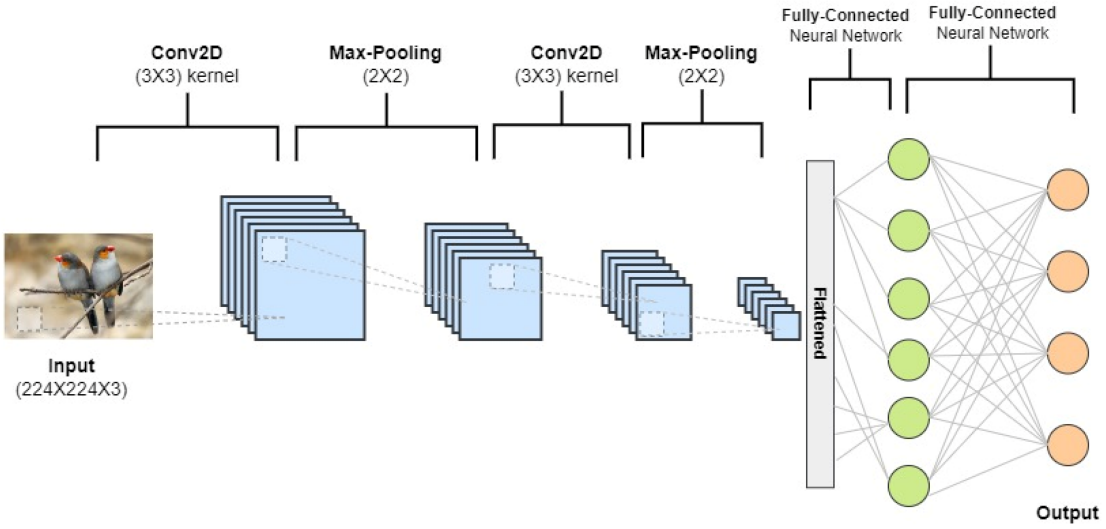


Figure 2.11.: Example of an architecture of a CNN [46].

imaging due to their ability to learn highly complex representations [59]. However, the local receptive field in the convolution operation limits capturing long-range pixel relationships [59]. Recent work has shown that attention-based transformer modules can fully replace the standard convolutions in deep NN by operating on a sequence of image patches, giving rise to Vision Transformers (ViT) [59]. With the attention-based mechanism, they overcome the limitation of CNNs in encoding long-range dependencies.

Vision Transformers

Following their success in Natural Language Processing (NLP), Transformers have been successfully applied to image classification tasks achieving state-of-the-art performance in benchmark datasets such as ImageNet [15]. Capitalizing on these advances in CV, the medical imaging field has also witnessed a growing interest in Transformers [59]. Dosovitskiy et al. [15] designed Vision Transformer (ViT) in 2021. An overview of the model architecture is depicted in Figure 2.12. Next, we will explain the intuition behind how and why it works.

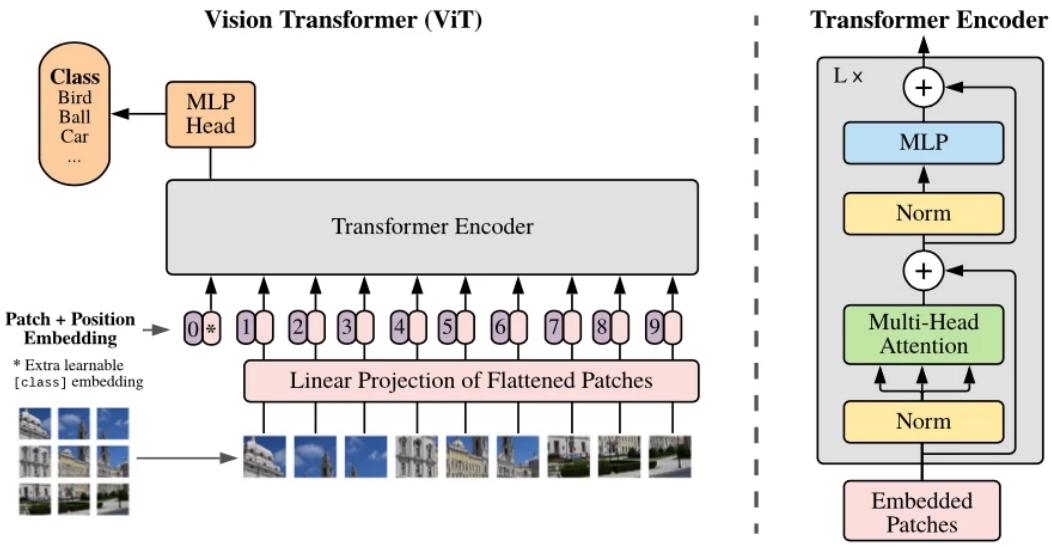


Figure 2.12.: ViT model overview [15].

Patch Embeddings: the standard Transformer receives the input as a 1D sequence of token embeddings.So to handle 2D images, we break the image $x \in (\mathbb{R}^{\mathbb{H}\times\mathbb{W}\times\mathbb{C}})$ into a smaller 2D patches [15]. (H, W) is the resolution of the original image and (P, P) is the resolution of the image patch, therefore $N = \frac{HW}{P^2}$ is the resulting number of patches. This process is similar to how sentences are broken into words in NLP [51].

Linear Projection: before passing the patches into the Transformer block, each patch is processed through a linear projection or embedding layer to map the image patch "arrays" to patch embedding "vectors" [51]. These vectors serve as a condensed way to understand the content of the image [38]. This process can be visualized in Figure 2.13.
Learnable Embeddings: for classification, ViT applies the same logic as in the popular BERT models [14], by adding an extra learnable [CLS] embedding at the beginning of the patch sequence. This [CLS] token is converted into a token embedding. What makes it special is that it does not represent a real image patch, it starts as a "blank slate" [51]. Second, after going through several encoding layers, the [CLS] output becomes the input into the classification head, helping the transformer learn a "general representation" of the entire image [51].

Positional Embeddings: transformers do not have any default mechanism that considers the order of patch embeddings [51]. We enable order with positional embeddings, which are learned vectors with the same dimensionality as the patch embeddings. These embeddings converge during training, showing high similarity to neighboring

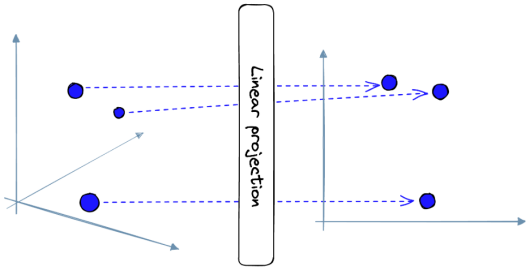


Figure 2.13.: The linear projection layer attempts to transform arrays into vectors while maintaining their "physical dimensions". This means similar image patches should be mapped to similar patch embeddings [51].

position embeddings. For example in Figure 2.14, the patch in the first row and column has its highest similarity in the top-left corner which represents the position very well. After adding the positional embeddings, the patch embeddings are complete. They are

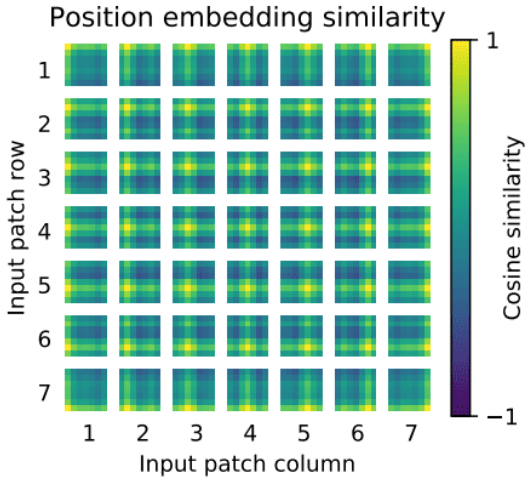


Figure 2.14.: Position embeddings sharing the same column and row show higher cosine similarity [15].

then processed by the ViT model as a standard transformer [51].

Transformer Encoder Block: the Transformer encoder consists of alternating layers of multiheaded self-attention and Multi-Layer Perceptron (MLP) blocks. Layer normalization (LN) is applied before every block to maintain stability, and residual connections after every block to facilitate information flow and gradients [15]. Detailed explanations of self-attention and multi-head attention are provided below to enhance understanding.

Self-Attention: to illustrate this mechanism consider the sentence "The animal didn't cross the street because it was too tired." When the model processes the word "it," self-attention enables it to associate "it" with "animal". As the model processes each word, self-attention allows it to examine other positions in the input sequence for context clues [2]. This is illustrated in Figure 2.15. The underlying mathematical

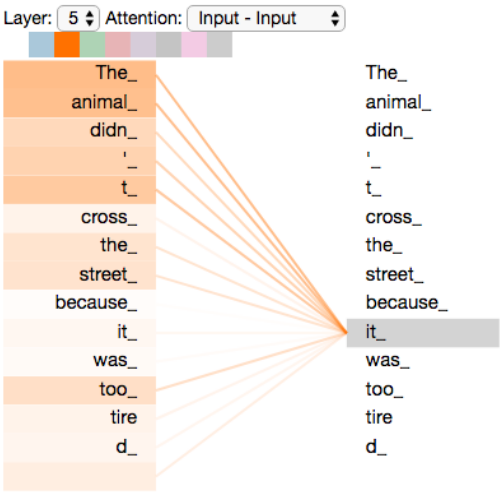


Figure 2.15.: As we are encoding the word "it", part of the attention mechanism was focusing on "The Animal" and baked a part of its representation into the encoding of "it" [2].

concept involves:

1. Creating three vectors (Query, Key, and Value) for each word embedding by multiplying the embedding with learned weight matrices W^Q, W^K, W^V during training.

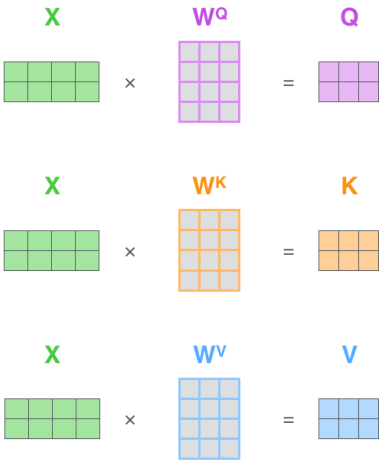
2. Calculating scores for each word by taking the dot product of the Query and Key vectors. For instance, when processing the self-attention for the word in position 1, the first score is $q1 \cdot k1$ and the second score is $q1 \cdot k2$.

3. Normalizing the scores by dividing them by $\sqrt{dk}$ (where dk is the dimension of the key vectors) and applying a softmax operation. This step determines the emphasis placed on each word at the current position.

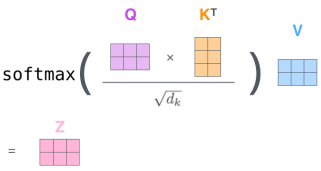
4. Multiplying each Value vector by its corresponding softmax score. This step helps diminish the influence of irrelevant words by scaling them with small numbers.

5. Summing up the weighted value vectors to produce the output of the self-attention layer at this position (for this word), and the resulting vector is then fed into the feed-forward neural network.

In the actual implementation, all these steps are done in matrix form for faster processing. Figure 2.16a shows how Query, Key, and Value matrices are calculated. In addition, steps 2 to 5 can be condensed into the formula in Figure 2.16b to calculate the outputs of the self-attention layer. This explanation can be translated to computer



(a) Every row in the matrix X corresponds to a word in the input sentence [2].

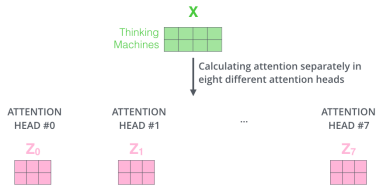


(b) Self-attention calculation in matrix form [2].

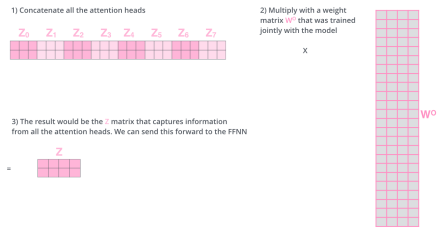
Figure 2.16.: Matrix calculation of self-attention.

vision, where patch embeddings replace word embeddings, and the self-attention scores calculated for each patch determine how much focus to place on other parts of the image.

Multi-Head Attention: Vaswani et al. [72] expanded self-attention by introducing multiple attention heads. This means that we have multiple sets of Query, Key, and Value weight matrices (in the paper they use 8 attention heads [72]). Using multiple heads allows the model to focus on different parts of the input simultaneously. To input the scores from these attention heads into the feed-forward NN, Figure 2.17 illustrates how the matrices calculated for each attention head are combined into a single matrix. This final matrix contains information from all the attention heads.



(a) Same calculation but 8 times with 8 different weight matrices.



(b) Concatenate the 8 matrices and then multiply them by an additional weight matrix WO.

Figure 2.17.: Multi-Head Attention [2].

MLP Head: takes the output feature vector of the [CLS] token and maps it to a classification prediction. This is usually implemented by a small feed-forward network or even a single linear layer [40].

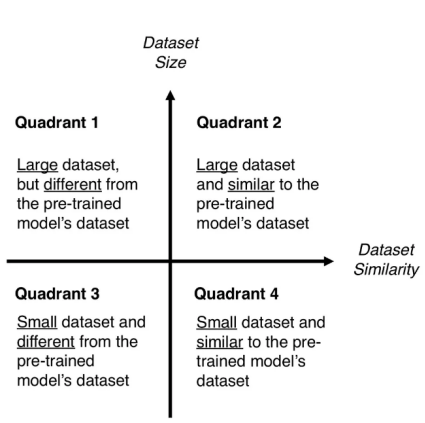
2.2.4. Transfer Learning

Transformers need a lot of training data to fully utilize their capabilities [15]. However, in the medical domain, large datasets are limited. ViT suffers from a lack of inductive bias when trained on small datasets, resulting in poor generalizability [15]. In DL, inductive bias refers to the set of assumptions that are built into a learning algorithm. These assumptions guide the learning process and help the model generalize from the training data to predict unseen data. Transfer learning is a powerful technique and it can help address inductive bias in ViT. The intuition behind this technique for image classification is that if a model is trained on a large and general enough dataset, this model will effectively serve as a generic model of the visual world [67]. The most common practice is to pre-train a model on a very large dataset (e.g. ImageNet, which contains 1.2 million images with 1000 categories) and then use the model either as an initialization or a fixed feature extractor for the task of interest [70]. There are two major transfer learning scenarios:

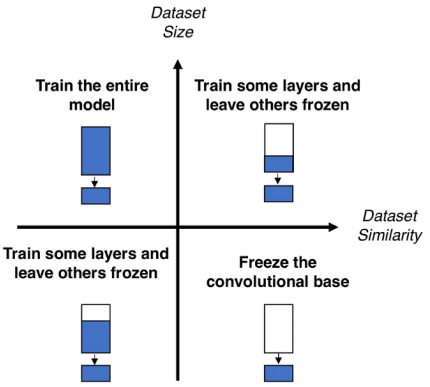
1. Feature extraction: start with a pre-trained model and add a new classifier on top of the model, which will be trained from scratch. You do not need to re-train the entire model because the base model already contains features that are generically useful for classifying pictures. However, the final classification part is specific to the original classification task, and subsequently specific to the set of classes on which the model was trained [67].

2. Fine-tuning: unfreeze a few of the top layers of the frozen model base and jointly train both the newly added classifier layers and the last layers of the base model [67]. You can fine-tune all layers or just specific higher-level layers. The earlier layers capture more general features, while the later layers focus on specific details.

Choosing the appropriate transfer-learning strategy depends on factors like the size of the new dataset and its similarity to the original dataset [70]. Figure 2.18 provides guidance on selecting the most appropriate transfer-learning strategy for each situation.



(a) Size-similarity matrix.



(b) Decision map for fine-tuning pre-trained models.

Figure 2.18.: Visual summary of the transfer learning strategy to use based on dataset size and similarity [45].

2.2.5. Evaluation of Deep Learning Models

To effectively assess DL models and determine the most suitable one, a robust validation strategy is crucial, accompanied by evaluation metrics specific to the clinical task. Considering the clinical context, the use of proper metrics is vital, as some metrics may be favorable from a data science perspective but might not align with clinical requirements, and vice versa. This section introduces two key components: first, the implementation of k-fold cross-validation for representative results, and second, the exploration of suitable metrics for binary image classification tasks.

K-fold Cross-Validation

A good validation strategy is basically how you split your data to estimate future test performance. To determine the efficacy of a model, it is important to split the dataset into training, validation, and testing [26]. The training set is used to train the model and update the model's parameters, the validation set is used for hyperparameter tuning and to assess the model's performance on data that the model hasn't seen during training and the test set is used to understand how well the model will perform on new, unseen data [26].

K-fold cross-validation is an advanced way of splitting the dataset for training. The need for cross-validation strategies arises because DL models are susceptible to overfitting on training samples, due to their large learning capacity [9]. Overfitting occurs when an algorithm becomes too specialized for the training data and does not generalize well on new unseen data [58]. Consequently, the accuracy of the model's predictions on its training set is not a reliable indicator of the model's future performance [63]. To avoid being misled by an overfitted model, generalization performance must be measured on a holdout test set which is independent of the training data [9]. However, especially in the medical domain, these external test sets might be too small to make a reliable prediction [49]. A solution for this problem is using K-fold cross-validation. It consists of splitting the available data into K partitions (typically K = 4 or 5), instantiating K identical models, and training each one on K-1 folds while evaluating on the remaining fold [11]. The validation score for the model used is then the average of the K validation scores obtained [11]. Figure 2.19 illustrates the procedure of a 5-fold cross-validation.

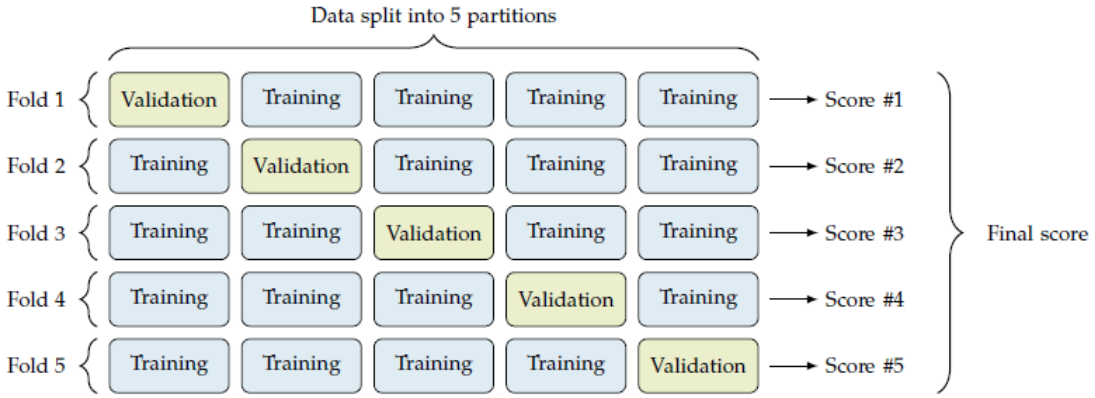


Figure 2.19.: 5-fold cross-validation [29].

Metrics

Several performance measures are defined for the evaluation of binary classification models, each offering unique insights into the model's effectiveness.

Confusion Matrix

The confusion matrix is one of the most common methods to present the results obtained by a classifier [43]. As shown in Figure 2.20, it presents True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) predictions in the form of a matrix, where the y-axis shows the predicted classes while the x-axis shows the true classes. These four outcomes are of prime significance since they are employed to formulate all the other performance measures [5]:

- **True Positive (TP):** the positive sample is correctly identified by the classifier.
- **True Negative (TN):** the negative sample is correctly identified by the classifier.
- **False Positive (FP):** the negative sample is incorrectly identified by the classifier as positive.
- **False Negative (FN):** the positive sample is incorrectly identified by the classifier as negative.

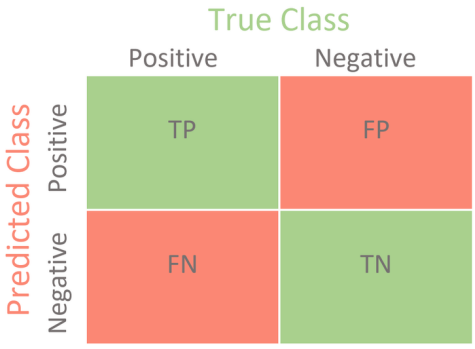


Figure 2.20.: Binary confusion matrix [36].

Accuracy

The accuracy is the ratio between correctly classified samples and the total number of samples [32]. The formula for calculating accuracy is the following:

$$\text{ACC} = \frac{\#\,\text{correctly classified samples}}{\#\,\text{total predictions}} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} \tag{2.1}$$

Sensitivity

Sensitivity is often referred to as Recall or True Positive Rate (TPR. It measures the model's ability to correctly identify positive cases. It is calculated as the ratio between correctly classified positive samples and all samples assigned to the positive class, as seen below:

$$\text{REC} = \frac{\#\,\text{true positive samples}}{\#\,\text{samples classified positive}} = \frac{\text{TP}}{\text{TP + FN}} \tag{2.2}$$

This metric is also regarded as being the most important for cancer research since it is far preferable to not "miss" anyone with cancer even if that means "tagging" some patients as having cancer that actually do not have the disease [30].

Specificity

Specificity is the negative class version of sensitivity and is also known as True Negative Rate (TNR) [32]. It measures the model's ability to correctly identify negative cases. Specificity is important when you want to minimize false positive errors. It is calculated as the ratio between correctly classified negative samples and all samples assigned to the negative class, as seen below:

$$\text{SPEC} = \frac{\#\,\text{true negative samples}}{\#\,\text{samples classified negative}} = \frac{\text{TN}}{\text{TN + FP}} \tag{2.3}$$

F1 Score

The F1 Score is the harmonic mean of precision and recall [32]. It provides a balanced measure of the model's performance that takes both false positives (precision) and false negatives (recall) into account. Precision is calculated as the ratio between correctly classified samples and all the samples assigned to that class [32]. In imbalanced problems, the F1 Score is less affected and it provides a more informative measure of how well a classifier performs on the minority class. It is calculated with the following equation:

$$\text{F1} = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} = 2 \cdot \frac{\text{TP}}{\text{TP + FP + FN}} \tag{2.4}$$

AUC-ROC

The Area Under the Receiver Operating Characteristic Curve (AUC-ROC) provides a comprehensive assessment of the classifier's performance across various decision thresholds. An ROC curve is a graph that visualizes the trade-off between a TPR and a False Positive Rate (FPR). As visualized in Figure 2.21, the higher the TPR and the

lower FPR is for each threshold the better, so classifiers that have steeper curves (closer to the top-left corner) are better [1]. The AUC-ROC Score is calculated as a summary metric to quantify the overall performance of the model, so the higher the value the better the model discriminates between positive and negative classes.

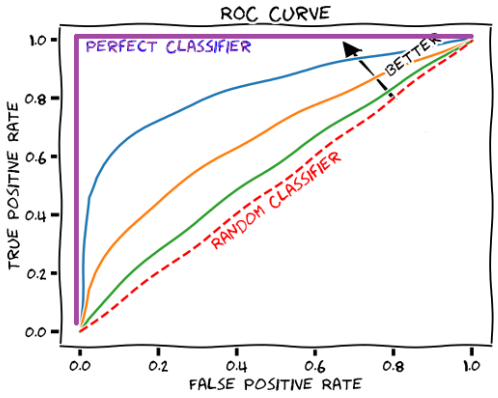


Figure 2.21.: ROC curve [47].

2.2.6. Explainable AI

In the previous subsection, different evaluation metrics were presented, which are used to assess the model's performance. While these metrics quantify the effectiveness of a model, they often do not explain why the model is making a certain prediction. Many DL models are seen as "black-box" structures, opaque and non-intuitive due to their complexity [66]. This lack of transparency and understanding can have serious consequences for building trust and adopting these models [35]. XAI seeks to provide insight into the decision-making ability of an AI system. It helps to understand how, when, and why predictions are made [35]. In other words, XAI techniques are used to make AI systems more transparent, interpretable, and understandable to humans. This is particularly important in medical diagnosis, where these systems need to explain the logic of making a certain decision to gain the trust of medical specialists, regulators and patients [62]. There are broadly two types of approaches to explain the results of deep neural networks in medical imaging - those using standard attribution-based methods and those using domain-specific techniques. The majority of the papers explain DL in medical imaging diagnosis using attribution-based methods [62]. These methods assess the importance of individual features, pixels, or regions within a medical image indicating their contribution to the model's decision, thus helping medical professionals focus their attention on the highlighted area.

SHapley Additive exPlanations

SHAP is a feature attribution method based on Shapley Values, a concept coming from game theory that calculates the contributions of each player to the outcome of a game [42]. The Shapley Value is calculated with all possible combinations of players. Given N players, it has to calculate outcomes for 2^N combinations of players. In the case of images, the "players" are the features (e.g., pixels in an image), and the "outcome of a game" is the model's prediction [22]. However, calculating the contribution of each feature is not feasible for large numbers of N, therefore SHAP does not attempt to calculate the actual Shapley Value but uses sampling to calculate the value [22]. SHAP values require a background dataset, which is a set of data instances used to represent the baseline or reference state [64]. Then, the values are computed by evaluating the model's output with and without the presence of individual features or groups of features. This background dataset helps establish the absence of the features [64]. One of the advantages of using SHAP is that it provides localized and fine-grained explanations for medical image classification. This is particularly important in medical diagnosis to highlight anomalies. Furthermore, SHAP values are model-agnostic which means they can be applied to a wide range of ML models [42].

In recent years, SHAP has been widely used in radiology to explain the decision of image-based diagnostic models, including those for identifying tumors, fractures, or other abnormalities in different medical imaging modalities. In a study by Ravi et al. [54], SHAP was employed to support radiologists' decision-making in the detection of COVID-19 and pneumonia cases from chest X-rays, based on the extent of the infection and severity level. Additionally, Bhandari et al. [6], proposed a DL model for predicting pulmonary disorders, employing diverse XAI techniques like Gradient-weighted Class Activation Mapping (Grad-CAM), Local Interpretable Model-agnostic Explanations (LIME) and SHAP for result interpretation. Figure 2.22 displays SHAP explanations for four outputs in a pulmonary disorder prediction, showing red pixels in the first explanation image indicating the positive contribution of predicting COVID-19. However, due to dataset complexity, specific features of each disorder are challenging to discern. SHAP has also demonstrated its versatility beyond medical imaging, showcasing impressive results in ImageNet classification. An illustration of this capability is evident in Figure 2.23, where SHAP provides explanations for animal predictions, highlighting distinct features that can help in their differentiation.

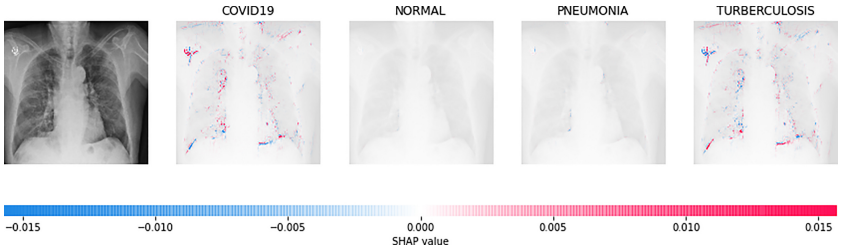


Figure 2.22.: SHAP explanations for pulmonary disorder prediction [6].

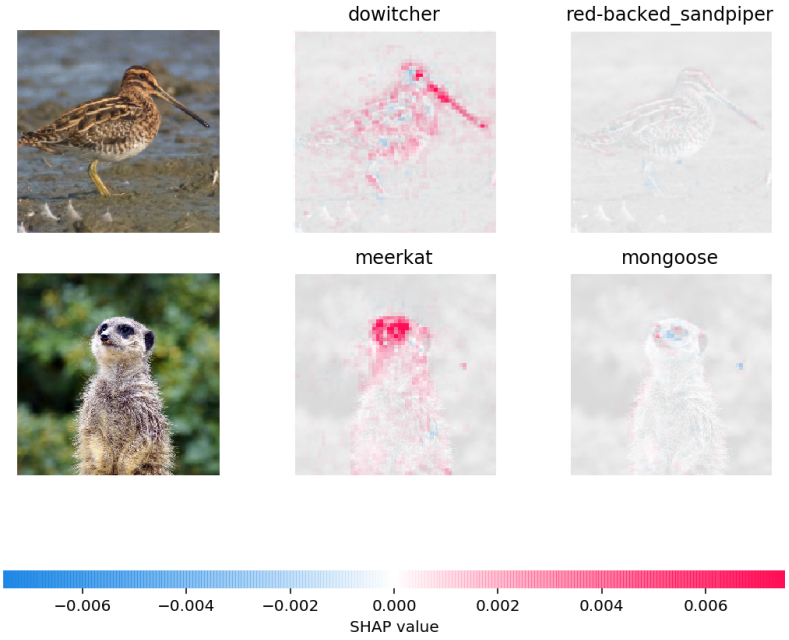


Figure 2.23.: SHAP explanations for ImageNet classification [60].

2.2.7. Related Work

DL has been widely applied to address many difficult medical problems [28]. However, little research exists regarding DL applied to bone tumors due to their relatively low incidence rates and the variability in tumor location and pathology types, making it challenging to gather sufficient imaging data [41]. Moreover, even less DL research exists focusing on bone tumor classification using plain radiographs [41]. He et al. [31] developed a DL model to classify primary bone tumors from preoperative radiographs, achieving accuracy levels comparable to subspecialists and outperforming

junior radiologists. The model was based on the EfficientNet-B0 architecture with pretraining on ImageNet. The study highlighted the challenges in differentiating benign from malignant lesions and demonstrated the potential for enhancing classification accuracy by evaluating the zone of transition, a critical indicator in this differentiation. While some research has explored the use of radiomics and ML in distinguishing cartilaginous tumors, particularly in classifying Chondrosarcoma from Enchondroma, these studies often focused on more general bone tumor classification [27]. For instance, in a recent study [17], ML models were constructed to differentiate Chondrosarcoma from Enchondroma using radiomic features extracted from MRI data, achieving high performance with a NN model. Despite obtaining remarkable results, this prior study did not specifically address the differentiation between ACT and Enchondroma. To the best of our knowledge, only one study by Gassert et al. [23] has tackled this specific classification task. However, their approach involved evaluating various imaging criteria using a combination of CT and MR images, rather than utilizing a DL-based approach. Therefore, we can confidently say that our study is the first to establish a DL algorithm for the differentiation between ACT and Enchondroma using conventional radiographs.

3. Materials and Methods

In this chapter, we present the materials and methods used in our research. The materials encompass the critical components of our study, while the methods detail the approaches and techniques applied to achieve our research objectives.

3.1. Materials

This section serves as the foundation for our research. To begin, we introduce the dataset and clarify the methodology used to split the data. We then explore the preprocessing techniques employed to prepare the dataset for optimal compatibility with our model. Additionally, we detail the strategies used for data augmentation. Lastly, the resources and computing infrastructure utilized are presented.

3.1.1. Dataset

The dataset plays a crucial role in obtaining a well-performing DL model. It serves as the primary source of information for these models to identify features and anomalies within medical images. If the dataset contains a lot of outliers, errors, or noise it is hard to extract the underlying patterns of the data during training [25]. Therefore, the performance of the model is directly linked to the quality of the dataset. Moreover, a representative and substantial amount of data is required for the model to generalize effectively on unseen data.

Dataset Structure

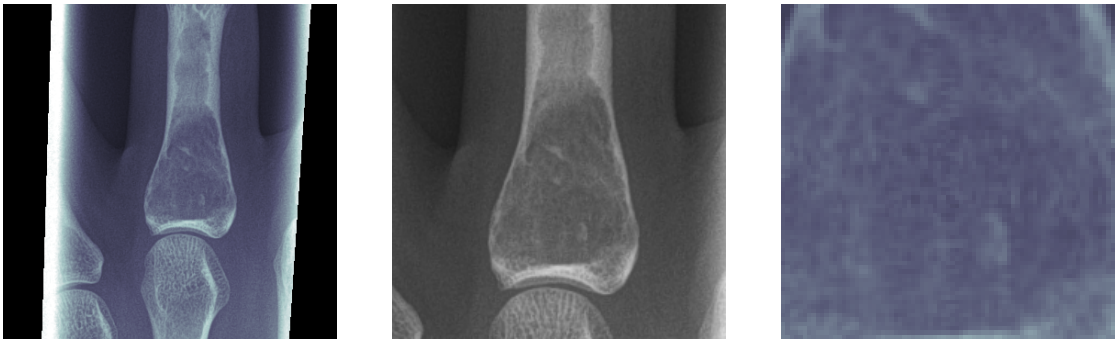
A total of 635 patients with Enchondroma or ACT lesions confirmed by histopathology diagnosis were collected from the Department of Orthopaedics and Sports Orthopaedics at Klinikum rechts der Isar. Of these, 528 were diagnosed with an Enchondroma, while 107 were found to have an ACT. Plain radiographs and relevant clinical variables, including patient demographics, were collected. In our data directory, three folders contain the imaging data accompanied by a Comma Separated Value (CSV) file with the corresponding labels and metadata:

1. '*og*': contains the original X-ray images in '.png' format, each identified with a filename which is used to connect the image with its corresponding patient metadata. Contains a total of 499 X-ray images. This is because, from the CSV file 136 entries were removed since corresponding images were not found in the folder.

2. '224*px*_10*bb*': contains the radiographic images resized to 224 x 224 pixels. This resizing is particularly advantageous, given that X-ray images are typically large and contain non-relevant information, such as black pixels. These images were prepared by tumor experts who employed bounding boxes to crop the image around the region of interest. This folder comprises 354 images, with some X-ray images absent from the original dataset.

3. '*only_tumor*': contains the radiographic images but in a smaller size, designed to emphasize only the tumor region. These images while informative, often lack the tumor boundaries, a crucial indicator for distinguishing between the two tumor types. These 374 images were likewise provided by tumor experts.

It's worth noting that all three folders use identical filenames to identify the same set of images. Figure 3.1 shows the original X-ray image, the image resized to 224, and only the tumor of the same case.



(a) Original X-ray image from 'og' folder.

(b) Bounding box from 224*px*_10*bb* folder.

(c) Smaller image from *only_tumor* folder.

Figure 3.1.: Patient case example with the three image sizes.

Each of these three folders is further organized into four subfolders:

- 'rACT_ACT': TN cases where the radiologist annotated them as ACT and biopsy confirmed it is an ACT.

- 'rACT_Ench': FN cases where the radiologist annotated them as ACT but biopsy confirms is an Enchondroma.
- 'rEnch_Ench': TP cases where the radiologist annotated them as Enchondroma and biopsy confirmed it is an Enchondroma.
- 'rEnch_ACT': FP cases where the radiologist annotated them as Enchondroma but biopsy confirms is an ACT.

For the purposes of our methodology, results, and discussion we will focus on the *og* and 224*px*_10*bb* images. Figure 3.2 illustrates the structure of our dataset.

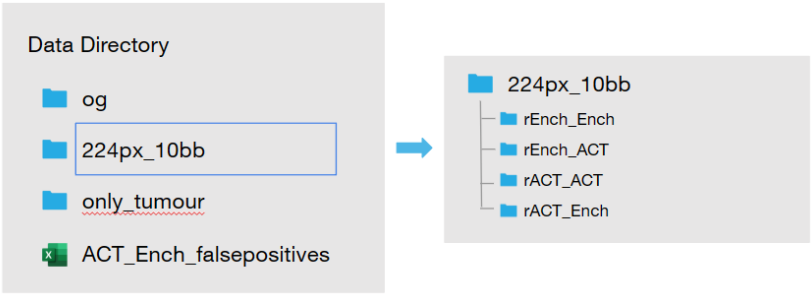


Figure 3.2.: Data directory structure.

Clinical and Radiological Data

Each image in our dataset is associated with a corresponding entry in the CSV file, providing essential patient and diagnostic information. The 'pseudo' serves as an anonymized patient identifier, while the 'file_name' links each patient's clinical data to their respective X-ray image. Additional patient identification numbers include 'AccNr,' 'rad_FallNr,' and 'pat_nr.' Patient demographics, including age, sex, and the imaging year, are also part of the dataset. Localization details specify the affected bone and its side (right or left). The 'Entity' column records histopathological diagnoses, while the 'comment' column provides radiologist annotations and supplementary notes. This last column is particularly valuable, as it captures the radiologist's diagnostic uncertainty, leading to the organization of our images into four subfolders, as previously mentioned. Figure 3.3 displays three entries from the CSV file with the aforementioned columns. A clear example of the radiologist's uncertainty is shown in the third entry and last column. In the context of our task, only the 'pseudo,' 'file_name,' and 'Entity' columns provide pertinent information. 'Entity' serves as the ground-truth label for our supervised binary classification. Additionally, we introduced a 'folder_name' column,

pseudo	file_name	AccNr	rad_FallNr	pat_nr	age	Entity	localisation	sex	year	side	comment
YNDcUdsCtIY	1.2.840.113654.2.70.1.178881985051049612057022329087502878672774_1.png	13557403	961154141	1487033	49	Enchondroma	Femur	M	2005	re	ACT
MQmQZW5oU3U	1.2.840.113654.2.70.1.148076959469259586913410057557127560762109 7.png	135394238	965777020	1890015	21	Enchondroma	Femur	W	2005	re	ACT
mbCVgdkR9IA	1.2.840.113654.2.70.1.215062235701449935475530841763253096605750_1.png	84183560	963847134	2194515	60	Enchondroma	Upper arm	W	2014	re	radiologically uncertain

Figure 3.3.: CSV file containing patient and diagnostic information.

which specifies the subfolder containing the image, thereby preserving the radiologist's annotation. Figure 3.4 shows the clinical data used for our task.

	pseudo	file_name	folder_name	Entity
0	QtK1z8SwbsU	1.2.840.113654.2.70.1.22266724559284778941527 3...	rEnch_Ench	Enchondrom
1	QtK1z8SwbsU	1.2.840.113654.2.70.1.32360509850784933147806 5...	rEnch_Ench	Enchondrom
2	hfUUPg12ZA8	1.2.840.113654.2.70.1.12847837146592343273101 1...	rEnch_Ench	Enchondrom
3	hfUUPg12ZA8	1.2.840.113654.2.70.1.29823471033789954771248 4...	rEnch_Ench	Enchondrom
4	rDHrRqnx_pE	1.2.840.113654.2.70.1.19506318993846205133298 0...	rEnch_Ench	Enchondrom
5	rDHrRqnx_pE	1.2.840.113654.2.70.1.29860794615500683342294 7...	rEnch_Ench	Enchondrom

Figure 3.4.: Relevant clinical information used for our task. The added column *folder_name* preserves the radiologist's annotation.

Data Split

To achieve optimal data splitting, we followed the methodology detailed in Subsection 2.2.5. Initially, we experimented with an 80/10/10 ratio for partitioning the dataset into training, validation, and test sets. However, we observed improved results with a 70/20/10 ratio. Consequently, our test set comprises 39 data samples. Given that multiple images can pertain to the same patient, our dataset splitting process relies

on the 'pseudo' information, which uniquely identifies each patient. This ensures that each patient exclusively belongs to either the test set or not on the test set. Furthermore, we execute a stratified split, taking the 'Entity' column into account, as our dataset is imbalanced. This approach helps achieve representative results by preserving a percentage of samples for each class. During model training, we used k-fold cross-validation technique, as elaborated in Subsection 2.2.5. Specifically, we employed a 5-fold cross-validation, a common choice for such tasks. We maintained fixed splits across different runs for comparative purposes and applied a stratified split.

Data Preprocessing

To prepare the input data to meet the model's requirements, several preprocessing steps are necessary. These preprocessing steps will be different for the original images and the resized images. In the case of our ViT, which is pre-trained on ImageNet-21k, we need to ensure that the images have a consistent resolution of 224 x 224 pixels. To standardize the size of the original images to the desired dimension of 224 pixels, we employed a resizing process. This involved calculating a ratio relative to the 224-pixel size, determining the new dimensions using this ratio, and then positioning the resized image onto a blank square canvas using padding. The resizing process is illustrated in Figure 3.5. For the bounding boxes, we can skip the resizing steps as we already have

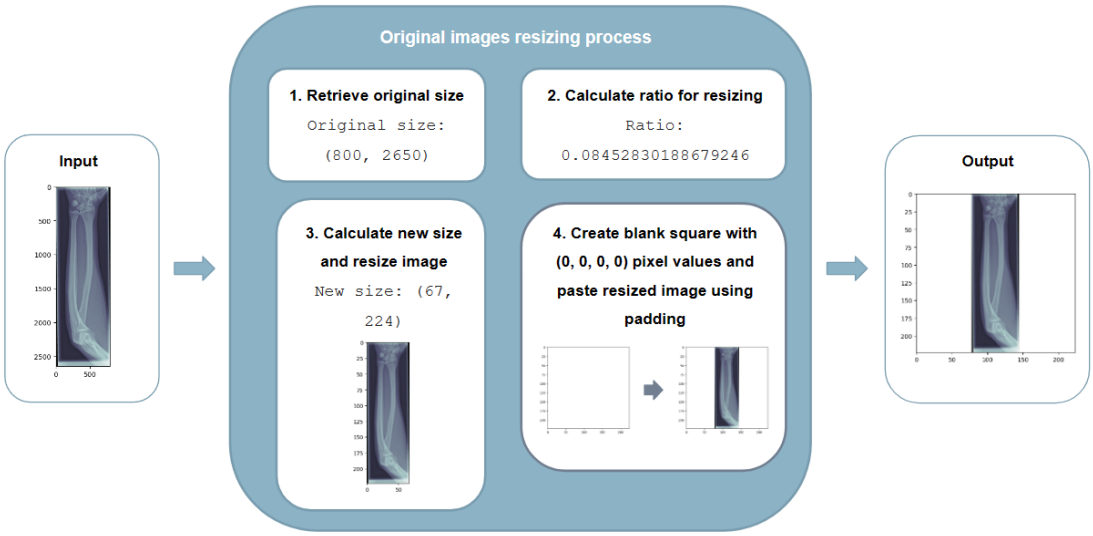
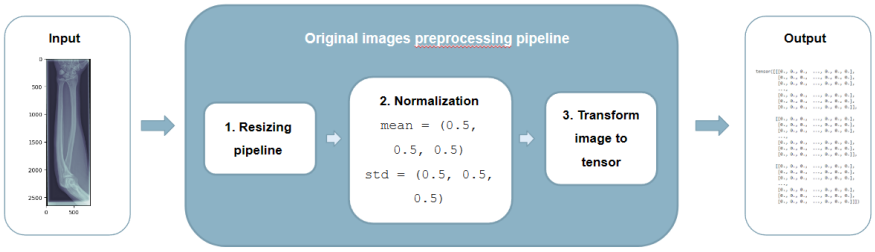


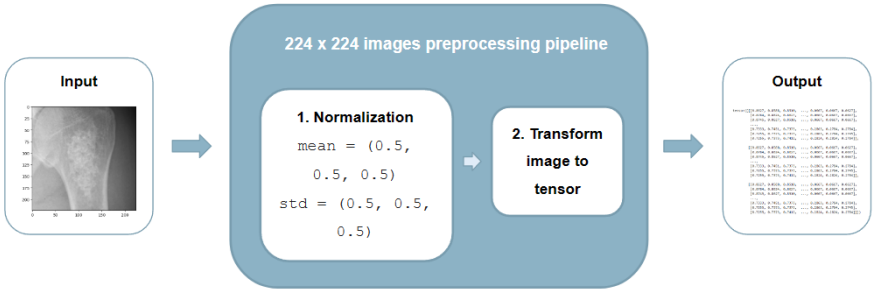
Figure 3.5.: Resizing steps for X-ray original images.

images at the required resolution. The subsequent preprocessing step for both image

sizes involves normalizing the images across the RGB channels. This is achieved by applying a mean of (0.5, 0.5, 0.5) and a standard deviation of (0.5, 0.5, 0.5). These specific mean and standard deviation parameters are provided by the ViT's feature extractor ensuring that the input data is correctly formatted. To facilitate passing a numeric representation of the image into the model, we must transform the image data into a tensor. This conversion allows the model to work with the data effectively. Figure 3.6 illustrates the steps for preprocessing the original images (Figure 3.6a) and the bounding boxes (Figure 3.6b). For the labels, we perform label encoding, a method that translates



(a) Preprocessing pipeline for the original images.



(b) Preprocessing pipeline for the bounding boxes.

Figure 3.6.: Different preprocessing steps for the original images and the bounding boxes.

the class labels of the two target classes into numerical values. The following table shows this label encoding: After applying these preprocessing techniques, each image

Entity	Numerical Value
Enchondroma	1
ACT	0

Table 3.1.: Label Encoding

in the dataset is represented as a dictionary containing two tensors: the normalized

'pixel_values' and the label, which is represented as a numerical value. An example is illustrated in Figure 3.7.

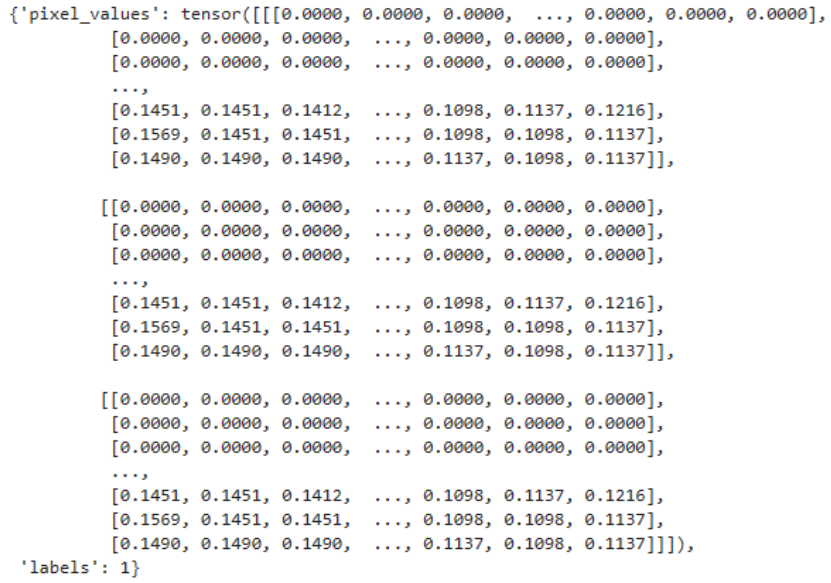


Figure 3.7.: After the preprocessing pipeline, each image is represented as a dictionary with two keys: *'pixel_values'* and *'labels'*

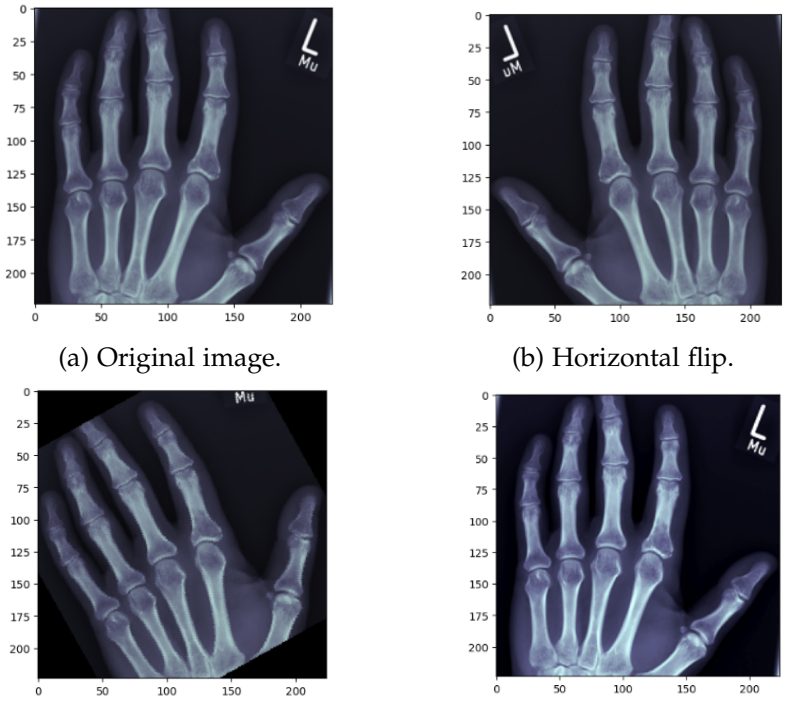
.

Data Augmentation

One of the primary challenges we encounter is the limited dataset. To address this limitation and enhance the initial dataset, we employ data augmentation, a technique that involves expanding the training set by creating modified data based on the existing samples [61]. This approach is essential for preventing overfitting and improving the model's performance. In the context of image data, augmentation is a straightforward process. It involves applying various geometric transformations, such as flipping, rotating, or shifting the images. In our work, we adopt data transformations similar to those proposed by Günther et al. [29] and Bloier et al. [8], as they worked with a similar dataset and identified the following data augmentations as the most effective:

- Random horizontal flip (see Figure 3.8b)
- Random rotation in the range of -30 to 30 degrees (see Figure 3.8c)

By implementing these transformations, we aim to enhance the dataset and improve the model's robustness and generalization capabilities. Additionally, we explore Color Jitter transformations (see Figure 3.8d), which modify contrast, brightness, and saturation. This exploration is particularly aimed at enhancing the visibility of tumor borders, a critical indicator of tumor aggressiveness.



(a) Original image. (b) Horizontal flip.

(c) Rotation. (d) Color Jitter.

Figure 3.8.: Different data augmentation techniques applied.

3.1.2. Resources

Software Setup

For our software environment, we used several tools and libraries to facilitate the development and training of our DL models. Our workflow is based on Python 3.9.12 programming language, while PyTorch[1] 1.12.1 framework is used as the foundation for creating, training, and evaluating these models. To reduce training times and

[1]The official PyTorch website can be found here: https://pytorch.org/

take full advantage of our hardware, we integrated the CUDA Toolkit[2] 11.3.1. For experiment tracking, visualization of results, and model parameters storage, Mlflow[3] 1.27.0 platform is used, ensuring reproducibility. Lastly, Git is used for version control of our code.

Hardware Setup

For both model training and inference, we utilized a DGX Station A100 with four Nvidia Corporation graphical processing units (GPUs), each with 80GB of memory. This powerful workstation is equipped with 64 high-performance 2.25 GHz cores and boasts 512 GB of DDR4 system memory. Our hardware environment is supported by the Linux/Ubuntu 20.04 distribution.

3.2. Methods

In this section, we provide a detailed explanation of our approach, which consists of two main tasks: classification and interpretability. We will start by explaining the key components of each task, including details about the baseline model and the specific SHAP variant employed. Following that, we will discuss a series of experiments conducted to enhance the model's performance and optimize the interpretability of our findings."

3.2.1. Proposed Approach

Classification

As mentioned in Subsection 2.2.7, our study is the first to use a DL-based approach for the differentiation between Enchondroma and ACT using conventional radiographs. Given the proven effectiveness of ViT models in image recognition, surpassing state-of-the-art CNN with reduced computational resources, we decided to use only a ViT model rather than a CNN architecture, for our classification task. To enhance interpretability and assess the contribution of image pixels in model predictions, we will compute SHAP values on our test images. Figure 3.9 illustrates the proposed framework to address both tasks.

[2]The NVIDIA CUDA Toolkit documentation can be found here: https://developer.nvidia.com/cuda-toolkit

[3]The machine learning lifecycle tool can be found here: https://mlflow.org/

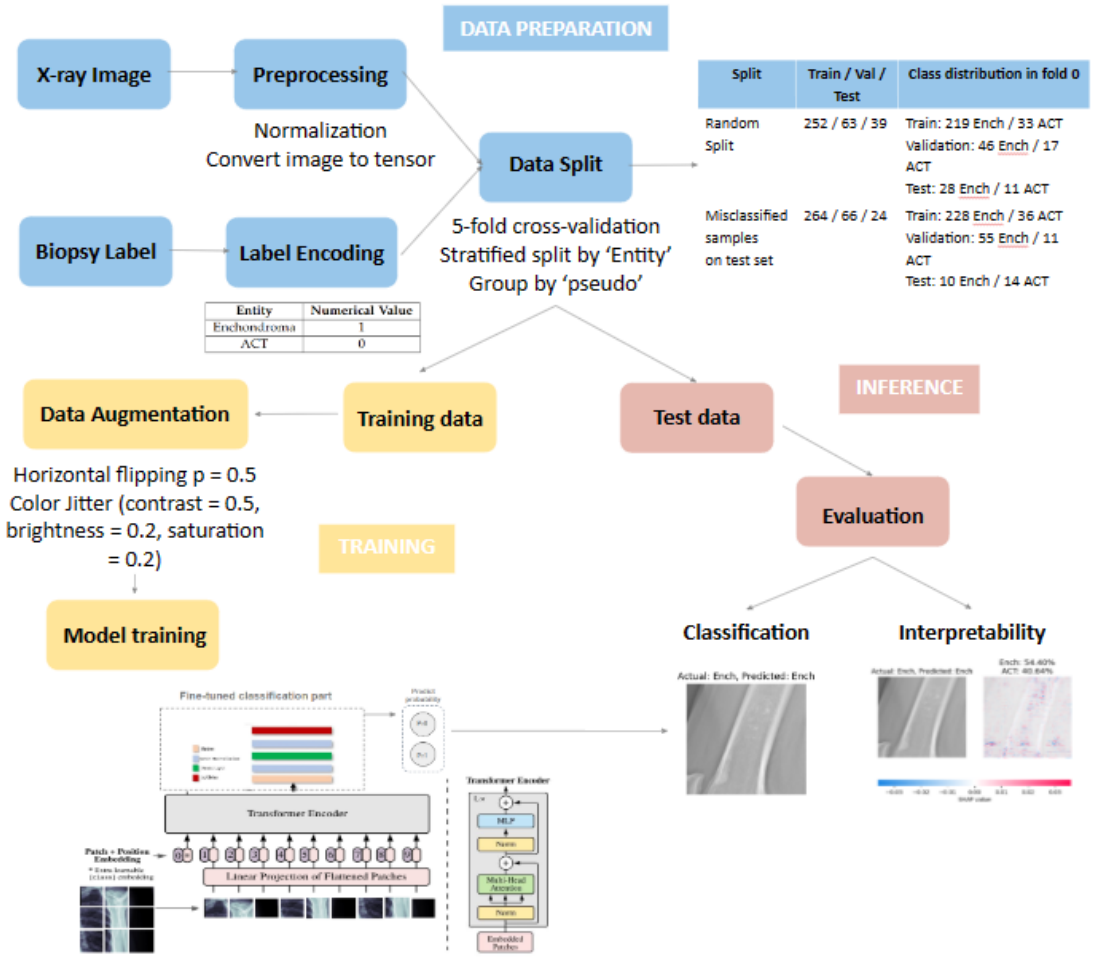


Figure 3.9.: Proposed framework.

Interpretability

As explained in Subsection 2.2.6, SHAP values serve as a tool for model interpretability, requiring a background dataset that represents the baseline or reference state. In our approach, we employ all training images as a reference distribution, utilizing Gradient Explainer to compute SHAP values on the test images. The process to compute SHAP values works as follows:

1. Background Dataset: the model undergoes training with the training set, which serves as the background dataset. This set is chosen because it likely represents the typical distribution of features that the model has learned to make predictions

on. Using it as the background allows SHAP values to be calculated with respect to a familiar distribution.

2. SHAP Computation: for each test image, SHAP values help decompose the model's prediction. Gradient Explainer calculates the gradients of the model's output y with respect to the input features xi. The gradient of each feature is $\frac{\partial y}{\partial xi}$.

   For a specific test image, you have its feature values:

   $$\mathbf{x}_{\text{test}} = (x_{1,\text{test}}, x_{2,\text{test}}, \ldots, x_{n,\text{test}}) \tag{3.1}$$

   Correspondingly, you have the feature values from the training set:

   $$\mathbf{x}_{\text{train}} = (x_{1,\text{train}}, x_{2,\text{train}}, \ldots, x_{n,\text{train}}) \tag{3.2}$$

   The differences between the test image's feature values and the corresponding feature values in the training set are calculated:

   $$x_{\text{diff}} = x_{\text{test}} - x_{\text{train}} \tag{3.3}$$

   Each gradient is then multiplied by the respective difference:

   $$x_{\text{diff}} \times \left(\frac{\partial y}{\partial x_i}\right) \tag{3.4}$$

3. Feature Attribution: the resulting product gives the SHAP values for each feature, indicating their contribution to the difference between the model's prediction for the test image and the expected prediction based on the training set.

4. Interpretation: SHAP uses colors for explanations:
   - Red pixels: signify a positive contribution to the prediction.
   - Blue pixels: signify a negative contribution to the prediction.

   The magnitudes of SHAP values indicate the strength of each feature's impact on the model's decision for the specific test instance.

5. Insights into Decision-Making: analyzing SHAP values provides insights into which features are influential in the model's decision for a given test image, based on the training data.

3.2.2. Experimental Setup

Experiment 1: Baseline Model

For the baseline model, we use Google's ViT Base variant model which is pre-trained on ImageNet-21k dataset at a resolution of 224 x 224 pixels, consisting of 14 million images and 21k classes. The images are presented to the model as a sequence of fixed-size patches with a resolution of 16 x 16. The pre-trained weights and model configuration can be downloaded from Hugging Face[4], which is an open source platform that provides a wide range of pre-trained transformer models.

Hyperparameter Optimization

The model is trained with the gradient-based Adam optimizer introduced by Kingma et al. [37], which is very popular in training ML models since it can adapt the learning rate during training, reducing it for parameters that have larger gradients. This adaptivity can lead to faster convergence in many cases. As for the loss function, Cross-Entropy Loss (CELoss) is used since it is a common choice in binary classification problems. This loss function increases as the predicted probability diverges from the actual label. However, when dealing with imbalanced datasets, it can lead to a bias in the model's predictions towards the majority class. We have initially experimented with CELoss and indeed, the predictions of the model are skewed to the most frequent class which is Enchondroma. To address our imbalance, we transition to WeightedCELoss, which allocates more penalty to the minority class so that these minority class samples are detected more accurately [55]. The results of these experiments can be found in Appendix B. To appropriately set the weights, we have assigned each class a weight that is inversely proportional to its frequency. Due to variations in dataset sizes between the original images and bounding boxes, distinct weights have been assigned to each. Given that the ACT class is the minority class, we assign a higher weight to this class to rectify the imbalance and enhance the model's performance in recognizing minority class instances. The equations used to calculate the weights for both the original images and bounding boxes are as follows:

Original Images Weights:

$$\text{Weight ACT} = \frac{499 \text{ samples}}{2 \text{ classes} \times 85 \text{ ACT samples}} = 2.94 \tag{3.5}$$

$$\text{Weight Ench} = \frac{499 \text{ samples}}{2 \text{ classes} \times 414 \text{ Ench samples}} = 0.61 \tag{3.6}$$

[4]The official Hugging Face website can be found here: https://huggingface.co/

Bounding Boxes Weights:

$$\text{Weight ACT} = \frac{354 \text{ samples}}{2 \text{ classes} \times 61 \text{ ACT samples}} = 2.90 \tag{3.7}$$

$$\text{Weight Ench} = \frac{354 \text{ samples}}{2 \text{ classes} \times 293 \text{ Ench samples}} = 0.60 \tag{3.8}$$

To find the best set of hyperparameters, manual tuning can be performed however it is very time-intensive. Therefore, simple techniques such as grid search are used to automate this process. For the batch size, we used values from 2 to 16, which describes how many samples are taken into account for one update step of the model's weights. The best results were obtained with a batch size of 4, which is used for all the experiments. The minimum number of epochs is set to 50 and the maximum to 300. However, this maximum is never reached since early stopping is employed. This technique stops training when the validation loss in our case, stops decreasing for a period of time of 20 epochs. As a starting point, the default parameters used by Kingma et al. [37] are chosen. The learning rate is one of the most important hyperparameters since it defines the pace at which a model updates its weights. It is initially set to the default value of 0.001. We also tried different values from 1e-3 to 1e-6 and compared the results. For the other parameters, we maintained the default setting of $\beta_1 = 0.9$, $\beta_2 = 0.9999$, and $\epsilon = 1e - 8$. Further experiments can be found in Appendix B.

Table 3.2.: Hyperparameter settings

Parameter	Initial Parameters	Optimized Parameters
Input Size	224 x 224	224 x 224
Loss Function	CELoss	WeightedCELoss
Optimizer	Adam	AdamW
· Learning Rate	0.001	0.001
Scheduler	step size = 10 gamma = 0.5	-
Batch size	16	4
min epochs	50	50
max epochs	300	300

Transfer Learning

To adapt the pre-trained ViT model to our specific task, we adjusted the classifier layers to align with the target task's two classes. Then, the model is fine-tuned by training the model on our dataset. As explained in Subsection 2.2.4, there are different transfer learning strategies. Initially, we experimented with unfreezing some of the top layers of the model's base by setting 'Requires Grad: True', indicating that these layers will undergo training and their weights will be updated during backpropagation. This strategy encompassed training the newly added classifier and unfrozen layers. Alternatively, we tested the efficacy of freezing all layers by setting 'Requires Grad: False', except the newly added classifier layers. This approach restricted training only to the newly introduced layers. These strategies are commonly used for leveraging pre-trained features while facilitating adaptation to the specific characteristics of our dataset. We assessed the model's performance under these two training strategies, presenting the 5-fold cross-validation results in Subsection 4.2.1.

Experiment 2: Original images vs. Bounding Boxes

In the second phase of our experiments, we performed a comparative analysis between training the model with the original X-ray images and utilizing the bounding boxes with a resolution of 224 x 224 pixels. This experiment aimed to discern the impact of focusing on relevant regions of interest on both model performance and interpretability through SHAP values. As an initial step, we trained the model on the original X-ray images. This training process involved implementing diverse transfer learning strategies, exploring various data augmentation techniques, and optimizing hyperparameters as detailed in Experiment 1. Upon achieving notable validation results, we evaluated the model's performance on the test set, with the metrics explained in Subsection 2.2.5. To understand the model's decision-making processes, SHAP values were computed on the test images, utilizing the entire set of training images as the background dataset. Since the original X-ray images contain non-relevant information like the background, we proceeded to retrain the model using the bounding boxes. Replicating the baseline model, we maintained the same setup used for training with the original images to ensure a direct comparison. We followed the same evaluation process for the model trained on the bounding boxes and the results were compared against the performance of the model trained on original images.

Experiment 3: Enhancing SHAP

In an effort to refine SHAP results and gain deeper insights into model interpretability, we conducted a series of targeted experiments.

Anatomical Subgroup Analysis

We explored the impact of using specific anatomical subgroups, specifically 'thigh' and 'hand' as background datasets for SHAP computations on the test set. This experiment was motivated by the idea of whether anatomical context plays a significant role in shaping the model's interpretability. With this analysis, we sought to uncover potential variations in feature contributions based on the anatomical characteristics of the images.

Threshold

In our continuous efforts to enhance and simplify SHAP result visualizations, we implemented a threshold mechanism. This involved the application of a predefined threshold to emphasize pixels with the greatest influence on model predictions. The primary objective was to reduce visual noise in SHAP visualizations, emphasizing the most impactful pixels and offering a clearer, more focused representation of feature importance. The process begins with computing SHAP on a specific test image, resulting in a list containing two arrays, each corresponding to the SHAP values for a specific class. Subsequently, we calculate the indices of the highest positive and negative SHAP values for both arrays and extract the actual values. A threshold is then established for positive and negative values based on a percentage of the highest positive and negative values. We chose a percentage of 0.75 for all test images, as it yielded optimal results. By applying this threshold, values between the highest positive value and the threshold are set to the highest positive value, resulting in uniform red color intensity for these pixels. Similarly, values between the highest negative value and the threshold are set to the highest negative value, creating uniform blue intensity for these pixels. The remaining values retain their original intensity. Figure 3.10 provides a visual representation of this threshold mechanism.

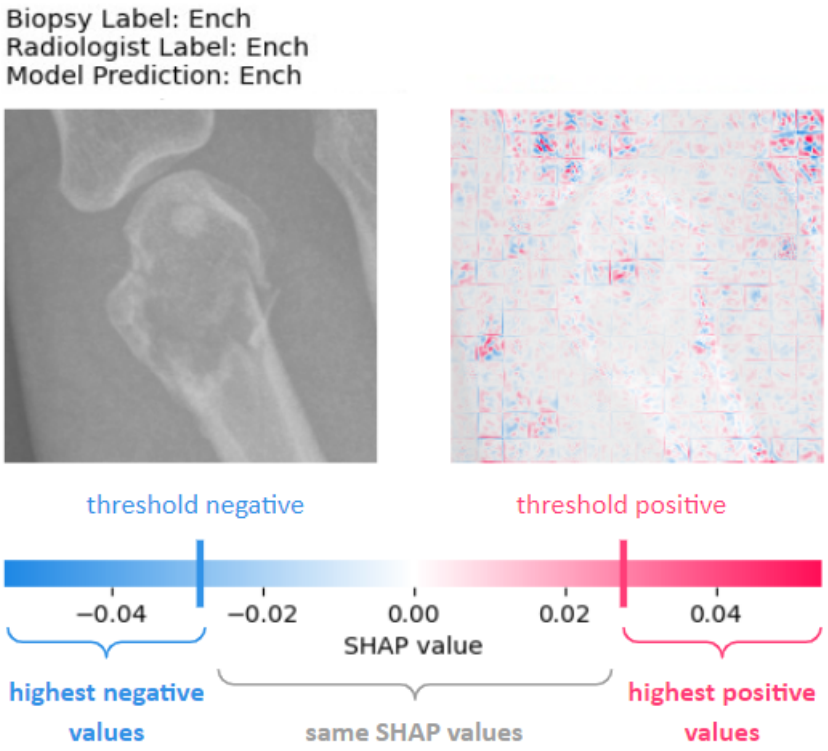


Figure 3.10.: Threshold mechanism illustrated.

Quantitative Results

Due to the complexity of SHAP visualizations and difficult radiological interpretation, we have opted to present quantitative measures for a more comprehensive understanding of the results. The computed metrics are as follows:

1. Mean Absolute SHAP: this metric represents the average impact of corresponding features on the model's prediction. A higher mean value indicates a more substantial influence.

2. Sum of SHAP: the sum of SHAP values represents the overall impact of the features on the model's prediction. A positive-sum suggests features contributing positively, while a negative sum suggests features contributing negatively.

3. Percentage of Red Pixels: a higher percentage of red pixels suggests a stronger positive contribution to the prediction, whereas a lower percentage implies a weaker positive impact.

4. Percentage of Blue Pixels: a higher percentage of blue pixels suggests a stronger negative contribution to the prediction, whereas a lower percentage implies a weaker negative impact.

We will analyze these metrics across samples and for both classes to gain insights and comparisons.

Experiment 4: Misclassified Samples Evaluation

Finally, in the last experiment, we realized that with our initial random allocation of samples into training, validation, and test sets, all cases that were misclassified by the radiologist were used in the training data and we could not evaluate the model's performance on these specific cases. To address this concern and ensure a more comprehensive evaluation, we exclusively allocate the misclassified samples, those cases where the radiologist's diagnosis diverges from the actual histopathology, as the test set. Conversely, the training set exclusively comprised samples that were correctly classified by the radiologist. By adopting this approach, we aimed to evaluate the model's robustness in handling ambiguous cases. The interpretability in these cases is crucial for uncovering potential discrepancies between radiologist interpretations and model predictions.

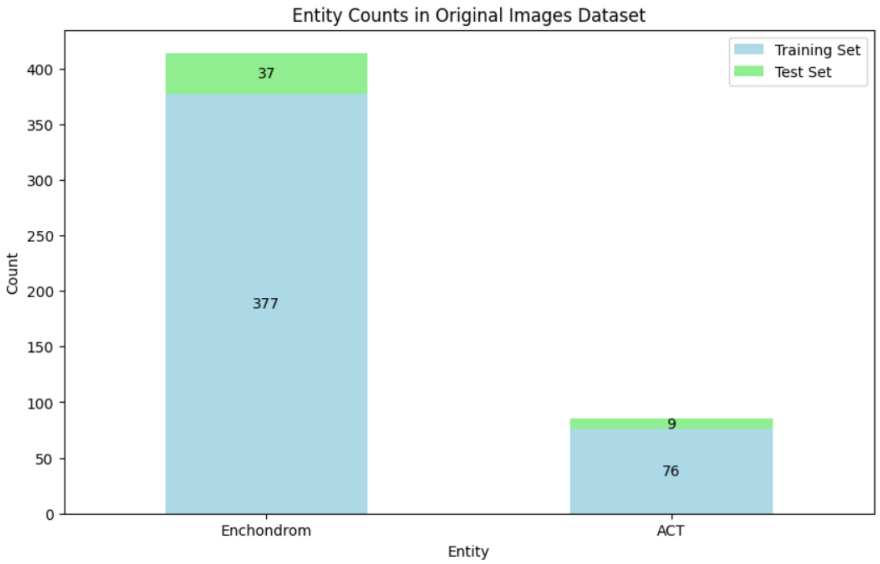
4. Results

In this chapter, the results obtained from the different experimental configurations are shown. Initially, we present dataset statistics to gain insights into the distribution of classes. The subsequent sections delve into specific experiment settings, showcasing outcomes and insights derived from various scenarios. The first two experiments assess the performance of our baseline model under different training strategies. The third experiment focuses on refining and simplifying SHAP values visualizations. Lastly, the fourth experiment examines the model´s performance on samples misclassified by the radiologist.

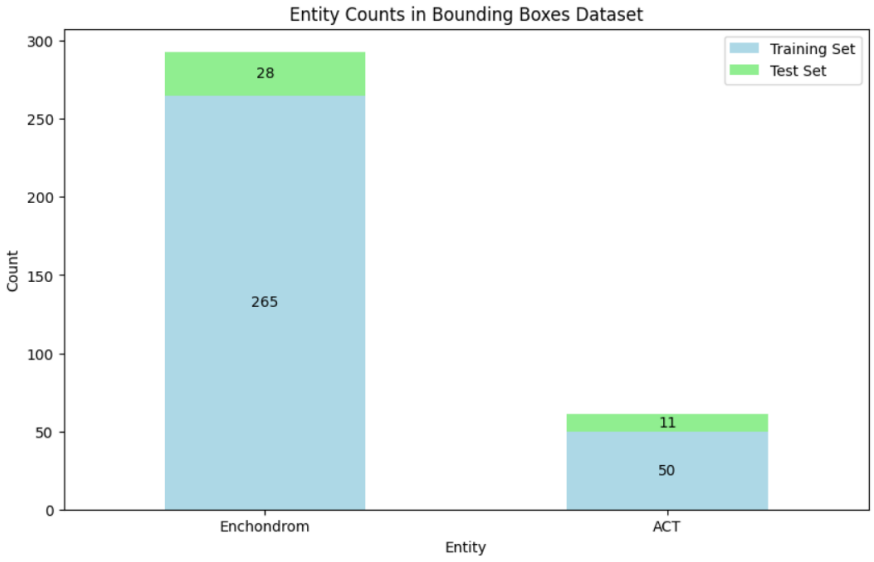
4.1. Dataset Statistics

4.1.1. Distribution of classes

In the following, dataset statistics are visualized. Exploring the distribution of classes can help us understand the impact of class imbalance on our task. Figure 4.1 illustrates the number of samples per entity in both the original X-ray images dataset and the bounding boxes, providing clarity on the train and test split. As we can see, there is a notable class imbalance with Enchondroma being the majority class and ACT being the minority class. In Figure 4.1a, Enchondroma represents 83.2% of the training set and 80.4% of the test set, while ACT accounts for 16.8% of the training set and 19.6% of the test set. In figure 4.1b Enchondroma represents 84.1% of the training set and 71.8% of the test set, while ACT makes up 15.9% of the training set and 28.2% of the test set. It is worth noting that the class distribution in the test set mirrors the imbalance in the training set. This ensures that the model's performance is evaluated in a way that reflects the real-world distribution of classes. To address this class imbalance, we have employed mitigation strategies, including the use of Weighted CELoss instead of the standard CELoss, as elaborated in Subsection 3.2.2. Additionally, our model evaluation employs several metrics because accuracy alone might not be a reliable indicator.



(a) Distribution of Enchondroma and ACT samples in our original dataset and an indication of train and test split.



(b) Distribution of Enchondroma and ACT samples in our bounding boxes dataset and an indication of train and test split.

Figure 4.1.: Distribution of classes in original and bounding boxes datasets.

4.1.2. Misclassified Samples

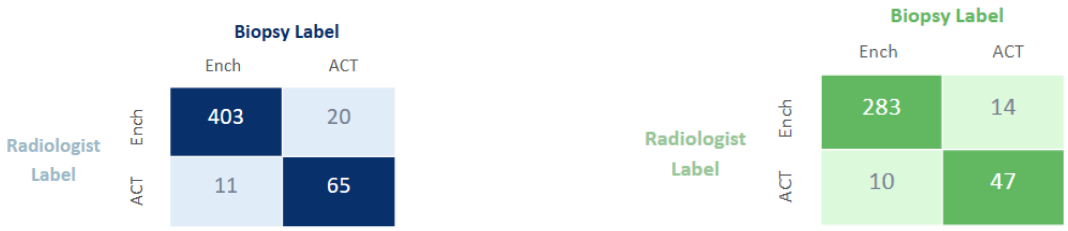
As explained in Subsection 3.1.1, our clinical dataset incorporates a column detailing the radiologist's annotation and supplementary notes, offering valuable insights into diagnostic uncertainty. The original images dataset comprises 499 X-ray images, featuring 414 cases of Enchondroma and 85 cases of ACT, all confirmed through histopathology diagnosis. Within the original images dataset we have:

- 403 TP cases accurately diagnosed as Enchondroma by both the radiologist and biopsy.
- 20 FP cases initially labeled as Enchondroma by the radiologist but confirmed as ACT by biopsy.
- 11 FN cases labeled as ACT by the radiologist but identified as Enchondroma by biopsy.
- 65 TN cases consistently diagnosed as ACT by both the radiologist and biopsy.

In the bounding boxes dataset, which comprises 354 X-ray images due to 145 images missing from the original dataset, 293 cases are classified as Enchondroma, and 61 as ACT lesions. Within the bounding boxes dataset we have:

- 283 TP cases accurately diagnosed as Enchondroma by both the radiologist and biopsy.
- 14 FP cases initially labeled as Enchondroma by the radiologist but confirmed as ACT by biopsy.
- 10 FN cases labeled as ACT by the radiologist but identified as Enchondroma by biopsy.
- 47 TN cases consistently diagnosed as ACT by both the radiologist and biopsy.

The following confusion matrices illustrate the misclassified samples by the radiologist, for both the original images dataset (figure 4.2a) and the bounding boxes dataset (figure 4.2b). Overall, the data suggests a substantial number of accurate diagnoses (TP and TN), highlighting the reliability of both radiologist annotations and biopsy results. However, the presence of discrepancies (FP and FN cases) emphasizes the challenges in accurate diagnosis based only on imaging data.



(a) Biopsy and radiologist labels confusion matrix in the original images dataset.

(b) Biopsy and radiologist labels confusion matrix in the bounding boxes dataset.

Figure 4.2.: Illustration of the misclassified samples by the radiologist in both original images and bounding boxes datasets.

4.2. Experiments

4.2.1. Experiment 1: Baseline Model

Hyperparameter Optimization

For the baseline model, we use Google's ViT pre-trained on ImageNet-21k dataset at a resolution of 224 x 224 pixels. The Base variant is chosen since we have a limited dataset. We conducted a comparison of metrics on the cross-validation results, utilizing both the initial hyperparameters and those optimized as specified in Subsection 3.2.2. Although our initial experimentation involved CELoss, for result comparisons, we have used WeightedCELoss for both parameter settings. We have also set 100 epochs for both scenarios. With the initial parameters, the model achieves an accuracy of 0.886 ± 0.006, as shown in Figure 4.1 in the first row. The first value refers to the mean average accuracy of the 5-fold cross-validation and the second is the standard deviation. The results show that optimizing the hyperparameters has led to improvements in various performance metrics.

Table 4.1.: Hyperparameter optimization for the ViT model. The metrics represent the mean of the cross-validation folds and the std.

Hyperparameters	Accuracy	AUC-ROC	F1-Score	Sensitivity	Specificity
Initial	0.886 ± 0.006	0.850 ± 0.011	0.929 ± 0.005	0.909 ± 0.009	0.790 ± 0.026
Optimized	0.910 ± 0.009	0.853 ± 0.018	0.945 ± 0.007	0.944 ± 0.012	0.762 ± 0.036

Transfer Learning

As mentioned in Subsection 3.2.2, we explored two transfer learning strategies: freezing most layers of the model while training some of the top layers and the newly added classifier layers, and exclusively training the new classifier layers. For both strategies, we adopted the optimized hyperparameters, as they consistently delivered superior performance.

Table 4.2.: Transfer learning strategies for the ViT model. The metrics represent the mean of the cross-validation folds and the std.

Training strategy	Accuracy	AUC-ROC	F1-Score	Sensitivity	Specificity
Some top layers and classifier layers	0.900 ± 0.021	0.812 ± 0.02	0.939 ± 0.032	0.952 ± 0.033	0.671 ± 0.044
Only classifier layers	0.910 ± 0.009	0.853 ± 0.018	0.945 ± 0.007	0.944 ± 0.012	0.762 ± 0.036

Data Augmentation

As detailed in Subsection 3.1.1, our approach involved the integration of various data augmentation techniques to augment the training set. Table 4.3 shows the results obtained with the various augmentations performed. Further experimentation is shown in Appendix B. We applied random horizontal flipping with a probability of 0.5, random rotation with a range of degrees (-30, 30) and random color jitter transformations with contrast = 0.5, brightness = 0.2 and saturation = 0.2. Applying horizontal flipping results in an enhancement in the model's performance across various metrics.Adding also color jitter transformations, shows very similar metrics to those achieved with horizontal flipping. Given these observations, we opt to further explore the first (higlighted in an intense green) and third augmentation techniques (higlighted in light green) in subsequent experiments.

Table 4.3.: Data augmentation techniques for the ViT model. The metrics represent the mean of the cross-validations folds and the std.

Data augmentation	Accuracy	AUC-ROC	F1-Score	Sensitivity	Specificity
Horizontal flip	0.909 ± 0.009	0.851 ± 0.015	0.944 ± 0.007	0.944 ± 0.012	0.757 ± 0.032
Horizontal flip + Random Rotation	0.849 ± 0.008	0.705 ± 0.008	0.910 ± 0.006	0.932 ± 0.010	0.478 ± 0.0162
Horizontal flip + Color Jitter	0.898 ± 0.010	0.835 ± 0.011	0.937 ± 0.007	0.937 ± 0.013	0.733 ± 0.026

4.2.2. Experiment 2: Original Images vs. Bounding Boxes

In the second experiment, our objective was to compare the model's performance during training with the original X-ray images against training with bounding boxes. This evaluation was conducted based on classification metrics and interpretability results. First, Table 4.4 presents the outcomes of training the model with these two image configurations, utilizing the best data augmentation techniques identified in Experiment 1. It is essential to mention that different weights were applied in the Weighted CELoss, as explained in Subsection 3.2.2, to account for the different dataset sizes in the original and bounding box datasets. The results indicate a slightly superior performance for the model trained on original images with both data augmentation techniques compared to the model trained on bounding boxes. However, despite having fewer samples, the model trained on bounding boxes performs well, especially with high sensitivity. Overall, the model demonstrates similar performance when trained with both datasets. Further, we evaluated the model's performance using the test sets for both original images and bounding boxes, employing the two most effective data augmentation techniques. Results are shown in Table 4.5. In both datasets, the incorporation of color jitter transformations, including contrast, brightness, and saturation, significantly contributed to enhanced overall performance, especially in terms of specificity.

Table 4.4.: Comparing the model's performance when training with original images and bounding boxes, using the two best data augmentation techniques. The metrics represent the mean of the cross-validations folds and the std.

Dataset	Data Augmentation	Accuracy	AUC-ROC	F1-Score	Sensitivity	Specificity
Original Images	Horizontal flip	0.909 ± 0.009	0.851 ± 0.015	0.944 ± 0.007	0.944 ± 0.012	0.757 ± 0.032
	Horizontal flip + Color Jitter	0.898 ± 0.010	0.835 ± 0.011	0.937 ± 0.007	0.937 ± 0.013	0.733 ± 0.026
Bounding Boxes	Horizontal flip	0.898 ± 0.003	0.818 ± 0.029	0.939 ± 0.002	0.960 ± 0.008	0.675 ± 0.055
	Horizontal flip + Color Jitter	0.882 ± 0.015	0.793 ± 0.044	0.931 ± 0.008	0.950 ± 0.010	0.635 ± 0.083

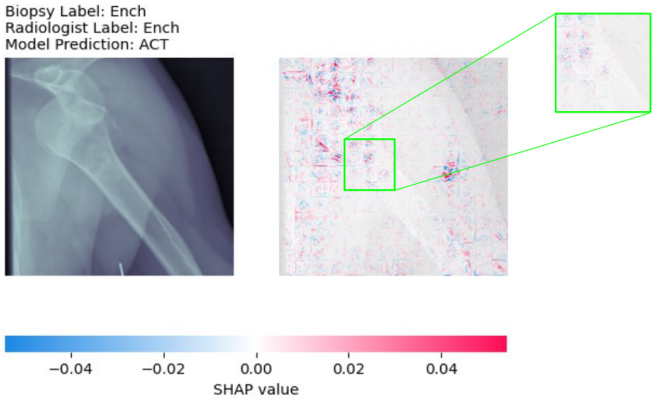
Table 4.5.: Comparing the model's performance when training with original images and bounding boxes, using the two best data augmentation techniques. The metrics represent the results of the test set.

Dataset	Data Augmentation	Accuracy	AUC-ROC	F1-Score	Sensitivity	Specificity
Original Images	Horizontal flip	0.739	0.544	0.842	0.865	0.222
	Horizontal flip + Color Jitter	0.761	0.641	0.849	0.838	0.444
Bounding Boxes	Horizontal flip	0.744	0.628	0.833	0.893	0.364
	Horizontal flip + Color Jitter	0.744	0.656	0.828	0.857	0.455

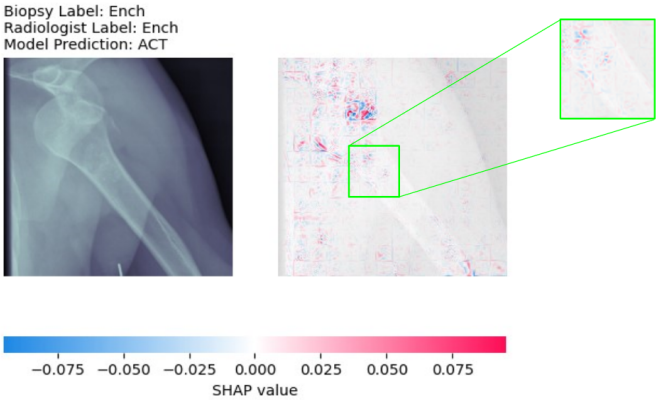
The following will explore the results of computing SHAP values on the test images. We will contrast the SHAP values computed under different scenarios: first, when training with original images and employing only horizontal flipping; second, under the same setup but incorporating color jitter transformations; third when training with bounding boxes and exclusively employing horizontal flipping; and lastly, under

identical conditions but integrating color jitter transformations. Our aim is to evaluate the influence of training with the two datasets and to discern whether contrast, brightness, and saturation contribute to the refinement of SHAP values. Additionally, within the results, we incorporate the 'Biopsy Label', 'Radiologist Label', and 'Model Prediction' to explore cases where the model's prediction aligns with the biopsy but the radiologist's diagnosis is incorrect. This inclusion can potentially help in addressing these challenging and uncertain cases for the radiologist. Additional SHAP results examples can be visualized in Appendix C.

Patient Case 1: figures 4.3 and 4.4, illustrate SHAP values computed on a test image corresponding to the same patient case. The key observation is that, when trained with the original images, the model's prediction is "ACT," deviating from the biopsy and radiologist's labels. However, when trained with bounding boxes, the model's prediction aligns correctly. By visualizing the model's decision-making under these diverse configurations, we discern that with bounding boxes, the model emphasizes the tumor borders, evident in the cropped boxes displaying a concentration of red pixels in this region, an indicative marker of Enchondroma. In contrast, with the original images, while there is some focus on the tumor, the features contributing most to the model's prediction lie outside the tumor area.



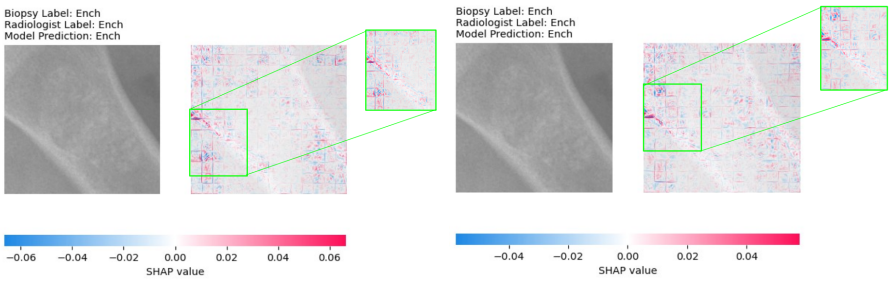
(a) SHAP computed on test image when training with horizontal flipping.



(b) SHAP computed on test image when training horizontal flipping + color jitter.

Figure 4.3.: SHAP values computed on patient case 1 with different training configurations for the original images.

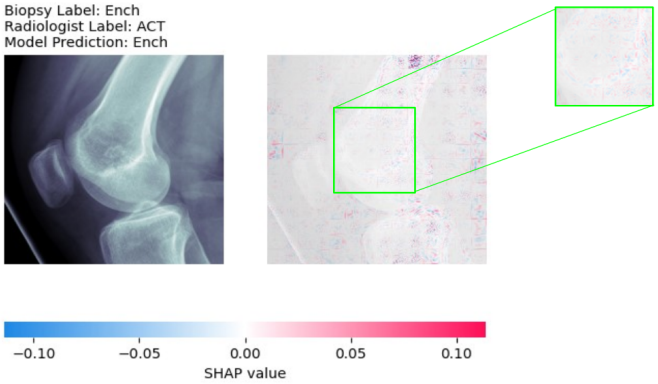
Patient Case 2: figure 4.5 highlights a scenario where the model's prediction aligns with the biopsy label, while the radiologist label is incorrect. In this case, we only visualize the original images. Given the random split employed during training with both the original images and bounding boxes, this specific case for the bounding boxes may be part of the training set rather than the test set. With the incorporation of color jitter transformations, we can see that the pixels within the tumor area prominently contribute to the model's prediction. Even when employing only horizontal flipping, the model correctly predicts the class, but its focus shifts primarily to other regions of the bone.



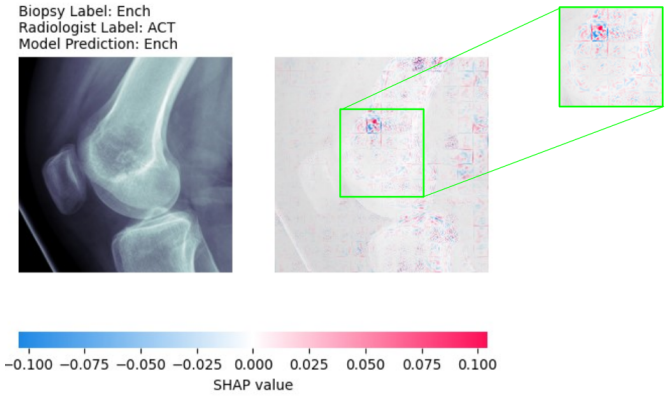
(a) SHAP computed on test image when training with horizontal flipping.

(b) SHAP computed on test image when training with horizontal flipping + color jitter.

Figure 4.4.: SHAP values computed on patient case 1 with different training configurations for the bounding boxes.



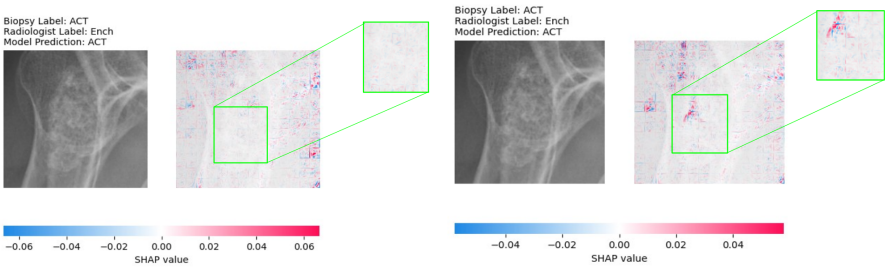
(a) SHAP computed on test image when training horizontal flipping.



(b) SHAP computed on test image when training with-horizontal flipping + color jitter.

Figure 4.5.: SHAP values computed on patient case 2 with different training configurations.

Patient Case 3: figure 4.6 highlights another case where the model's prediction aligns with the biopsy label, while the radiologist label predicts is an Enchondroma. In this case, we only visualize the bounding boxes. As in the previously discussed scenario, the introduction of color jitter transformations reveals that pixels within the tumor area play a significant role in shaping the model's prediction.

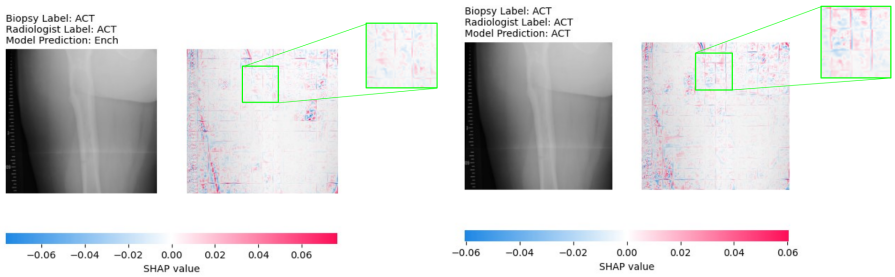


(a) SHAP computed on test image when training with horizontal flipping.

(b) SHAP computed on test image when training with horizontal flipping + color jitter.

Figure 4.6.: SHAP values computed on patient case 3 with different training configurations.

Patient Case 4: figure 4.7 shows a case where the model's prediction aligns with the biopsy and radiologist label when training with horizontal flipping and color jitter transformations, but diverges in its prediction when trained only with horizontal flipping. If we take a closer look at Figure 4.7b, there is a slightly stronger contribution of the pixels within the tumor area to the model's prediction. In contrast, Figure 4.7a, the pixels that have the most influence on the prediction are situated in the background of the image.



(a) SHAP computed on test image when training with horizontal flipping.

(b) SHAP values computed on test image when training with horizontal flipping + color jitter.

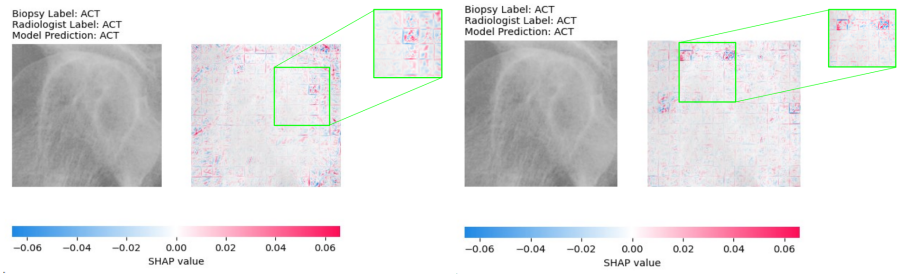
Figure 4.7.: SHAP values computed on patient case 4 with different training configurations.

4.2.3. Experiment 3: Enhancing SHAP

Anatomical Subgroup Analysis

In this experiment, and for the sake of result simplicity, we opted for a single training configuration: training the model with bounding boxes and employing horizontal flipping along with color jitter transformations. In the following, we proceed to evaluate the influence of anatomical context. To achieve this, we compare SHAP values derived from two scenarios: first, as the baseline, we use all training images as the background dataset and compute SHAP on all test images; second, restrict the analysis to only the training images where the tumor is located in the thigh and computing SHAP solely on the corresponding test images. We repeat this comparison for cases involving tumors in the hand as well.

Patient Case 5: figure 4.8 reveals that restricting the background dataset to only thigh training images has no relevant impact on the SHAP results. In both scenarios, it is illustrated that the pixels influencing the model's prediction are primarily concentrated along the borders of the tumor.

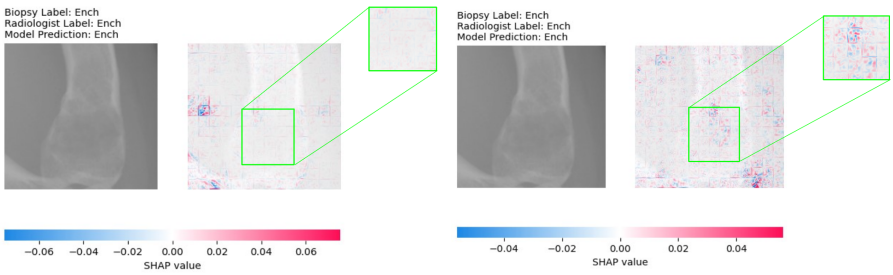


(a) SHAP values computed on test image using all training images as background dataset.

(b) SHAP values computed on test image using only thigh training images as background dataset.

Figure 4.8.: SHAP values computed on patient case 5 with different background datasets.

Patient Case 6: however, figure 4.9 demonstrates a notable distinction when employing only hand training images as the background dataset, SHAP values differ, with pixels subtly highlighted within the tumor area.



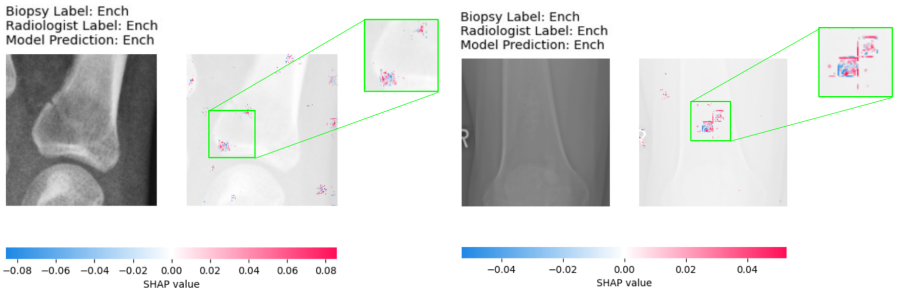
(a) SHAP values computed on test image using all training images as background dataset.

(b) SHAP values computed on test image using only hand training images as background dataset.

Figure 4.9.: SHAP values computed on patient case 6 with different background datasets.

Threshold

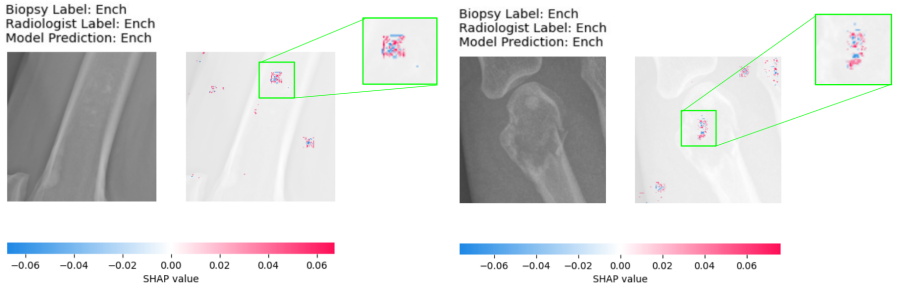
In the previous experiment, SHAP saliency maps contained a lot of red and blue pixels, making it challenging to identify the pixels that have the most influence on the model's prediction. To enhance interpretability, we applied the threshold mechanism outlined in Subsection 3.2.2 to all SHAP values computed on test images. For the sake of clarity in presenting results, we have opted to show examples exclusively from the SHAP values obtained when training the model with bounding boxes, employing horizontal flipping, and incorporating color jitter transformations. Illustrated in figure 4.10, figure 4.11 and figure 4.12 are the SHAP values computed for various patient cases using the threshold mechanism. As observed, in all instances, the pixels that contribute the most to the prediction appear within the tumor area, typically concentrated at the center of the tumor. Furthermore, in each of these cases, the model accurately predicted the entity, and there is alignment with both the biopsy and radiologist labels. More SHAP results examples are shown in Appendix C.



(a) SHAP values computed on patient case 7 using threshold.

(b) SHAP values computed on patient case 8 using threshold.

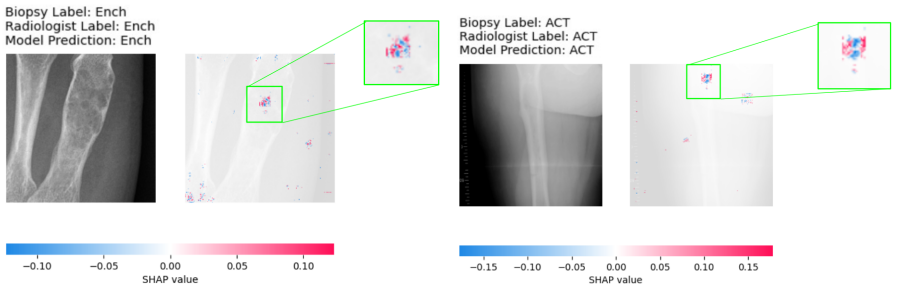
Figure 4.10.: SHAP values computed on patient cases 7 and 8 using threshold.



(a) SHAP values computed on patient case 9 using threshold.

(b) SHAP values computed on patient case 10 using threshold.

Figure 4.11.: SHAP values computed on patient cases 9 and 10 using threshold.



(a) SHAP values computed on patient case 11 using threshold.

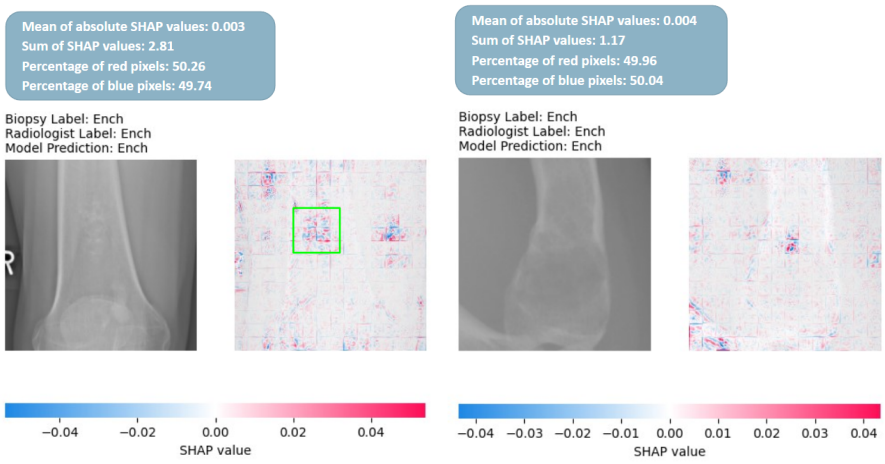
(b) SHAP values computed on patient case 4 using threshold.

Figure 4.12.: SHAP values computed on patient cases 11 and 4 using threshold.

Quantitative Results

In this experiment, our aim was to provide a more comprehensive understanding of the results. To achieve this, we calculated various metrics to quantify the SHAP results. With this, we wanted to see if we could extract some patterns from these metrics that could help in the differentiation between Enchondroma and ACT. Next, we showcase four patient cases with the metrics outlined in Subsection 3.2.2: mean of absolute SHAP, sum of SHAP, percentage of red pixels, and percentage of blue pixels. As consistent with previous experiments in this section, we employed a single training configuration: training the model with bounding boxes and incorporating horizontal flipping and color jitter transformations.

In figure 4.13 and figure 4.14, we observe that for Enchondroma cases, the mean of absolute SHAP is notably higher, indicating a more pronounced influence of features contributing to the model's prediction. This observation aligns with the positive and high sum of SHAP values in both Enchondroma cases, reflecting a stronger positive contribution from these features.



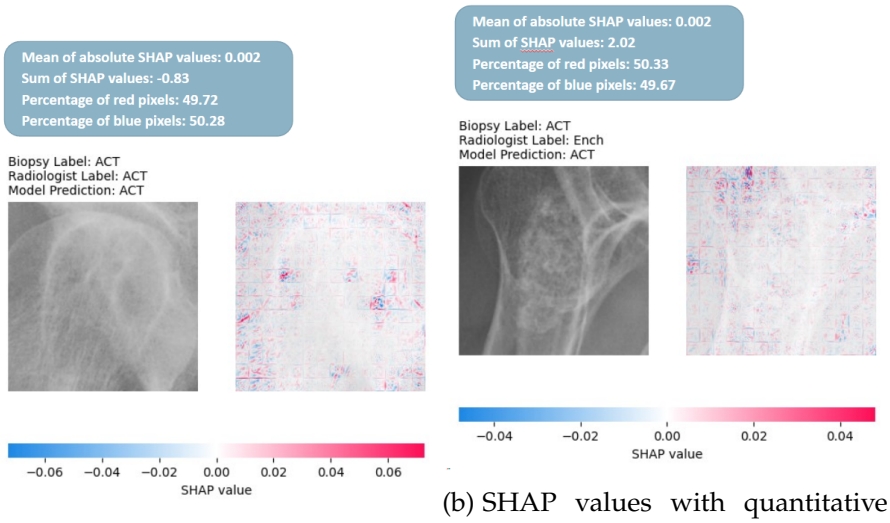
(a) SHAP values with quantitative metrics for a case where biopsy label, radiologist label, and model prediction align, confirming it is an Enchondroma.

(b) SHAP values with quantitative metrics for a second case where biopsy label, radiologist label, and model prediction align, confirming it is an Enchondroma.

Figure 4.13.: SHAP values with quantitative metrics for different patient cases.

However, in the case illustrated in figure 4.14a, the sum of SHAP is negative, implying a negative contribution to the prediction. This contradiction is further accentuated by a higher percentage of blue pixels than red pixels, suggesting a conflicting influence.

Despite these conflicting indications, the model persists in predicting ACT, showcasing the complexity of its decision-making process. Figure 4.14b is another challenging scenario where the sum of SHAP is positive, indicating a positive contribution to the prediction of ACT. Yet, upon closer inspection, we notice that the model does not seem to focus on the tumor area. Despite these challenges, these metrics can be helpful in assessing cases where model prediction and feature contributions contradict, guiding our attention toward these confusing cases.



(a) SHAP values with quantitative metrics for a case where biopsy label, radiologist label, and model prediction align, confirming it is an ACT.

(b) SHAP values with quantitative metrics for a case where biopsy label and model prediction align, confirming it is an ACT, but the radiologist annotated it as Enchondroma.

Figure 4.14.: SHAP values with quantitative metrics for different patient cases.

4.2.4. Experiment 4: Misclassified Samples Evaluation

In this experiment, our objective is to assess the model's performance specifically on samples misclassified by the radiologist, as elaborated in Subsection 3.1.1. To achieve this, we will create a new test set exclusively comprising the radiologist's misclassified samples. The model will be trained using the correctly classified samples. Both the original images dataset and the bounding boxes dataset will be employed in the experiment. Notably, the bounding boxes dataset lacks 7 cases present in the original images dataset, and evaluating these cases is a crucial aspect of our analysis. The model will be trained using horizontal flipping + color jitter transformations,

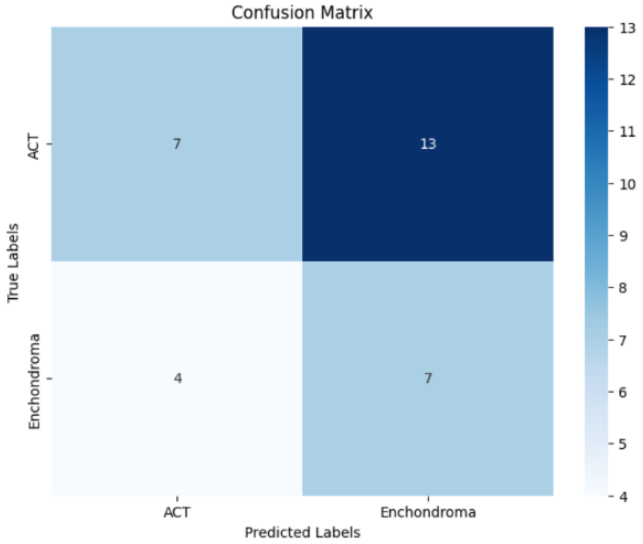
a configuration that demonstrated superior results in the context of SHAP values. Furthermore, we aim to explore the interpretability of these cases. This assessment can provide valuable insights into the model's decision-making process, particularly in complex and confusing cases for the radiologist.

Table 4.6 compares the performance of the model under two scenarios: one involving a random split for train/validation/test and the other utilizing all misclassified samples by the radiologist as the exclusive test set. In this second scenario, all correctly classified samples are employed during training. As we can see, for both the original images and bounding boxes, using the correct classified samples for training demonstrates improved performance metrics, especially in terms of specificity and AUC-ROC. This improvement in specificity suggests an increased accuracy in identifying true negative cases. These observed differences in model performance can be attributed to the nature of the data used for training. In the second scenario, where correct radiologist annotations are used, the training data may be more consistent and reliable. Consequently, the model might be exposed to fewer challenging and more confusing samples, leading to a better generalization.

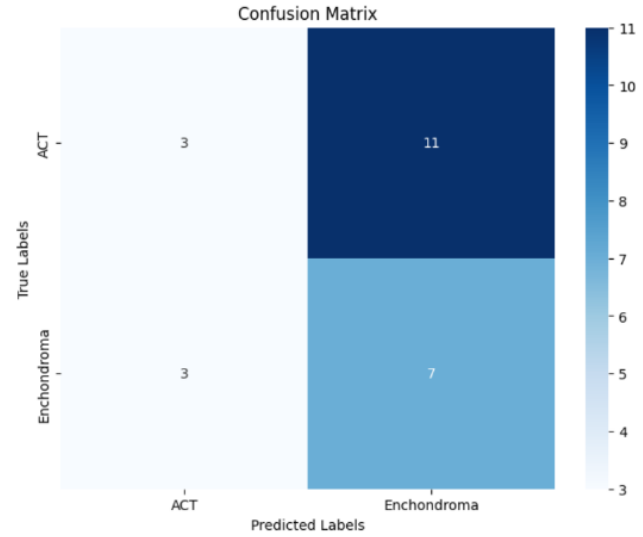
Figure 4.15 illustrates the confusion matrices obtained for both the original images (figure 4.15a) and the bounding boxes (figure 4.15b), using misclassified samples as the test set. In the original images dataset, figure4.2a showed that out of the 414 Enchondroma cases, 11 were misclassified by the radiologist. From these, the model correctly predicted 7 out of 11 cases. For the 85 ACT cases, where 20 were misclassified by the radiologist, the model accurately predicted 7 out of 20. In the bounding boxes dataset, figure 4.2b showed that out of the 293 Enchondroma cases, 10 were misclassified by the radiologist. From these, the model correctly predicted 7 out of 10 cases. For the 61 ACT cases, where 14 were misclassified by the radiologist, the model predicted 3 out of 14 cases correctly. Overall, these results demonstrate the ability of the model to correctly identify misclassified Enchondroma cases but challenges in accurately predicting misclassified ACT cases.

Table 4.6.: Comparing model performance when performing a random train/val/test split and setting all misclassified samples in the test set. The metrics represent the mean of the cross-validations folds and the std.

Dataset	Test Set	Accuracy	AUC-ROC	F1-Score	Sensitivity	Specificity
Original Images	Random Split	0.898 ± 0.010	0.835 ± 0.011	0.937 ± 0.007	0.937 ± 0.013	0.733 ± 0.026
	Misclassified Samples	0.926 ± 0.006	0.862 ± 0.026	0.957 ± 0.004	0.956 ± 0.005	0.768 ± 0.057
Bounding Boxes	Random Split	0.882 ± 0.015	0.793 ± 0.044	0.931 ± 0.008	0.950 ± 0.010	0.635 ± 0.083
	Misclassified Samples	0.913 ± 0.008	0.812 ± 0.024	0.950 ± 0.005	0.959 ± 0.013	0.665 ± 0.049



(a) Confusion matrix obtained using the misclassified original images as test set.



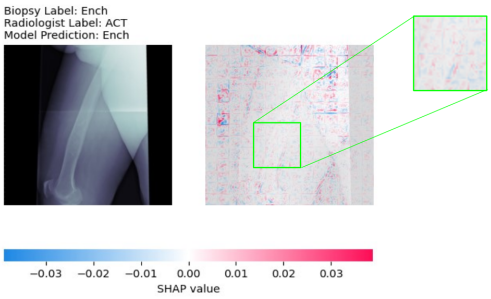
(b) Confusion matrix obtained using the misclassified bounding boxes as test set.

Figure 4.15.: Confusion matrices for the original images and bounding boxes, using misclassified samples as test set.

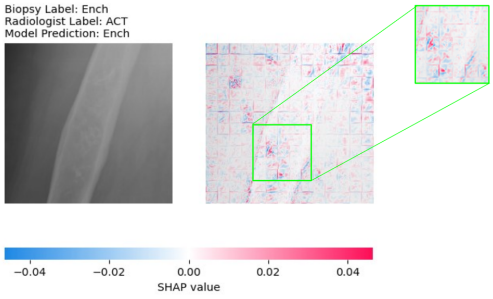
In the subsequent analysis, we computed SHAP values for test images that were misclassified by the radiologist but correctly classified by the model. We compared

the SHAP values derived from training with original images to those obtained when training with bounding boxes.

In Figure 4.16, both the original image and bounding box scenarios resulted in the model predicting the correct class. However, upon examining the saliency map, we observed that, in the original image, the pixels contributing the most to the model's prediction were in the background, while in the bounding box, pixels within the tumor area significantly influenced the prediction of Enchondroma.



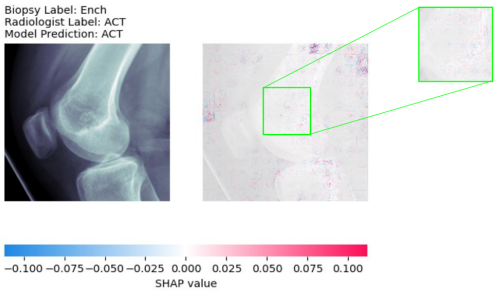
(a) SHAP values computed on misclassified test image when training with original images.



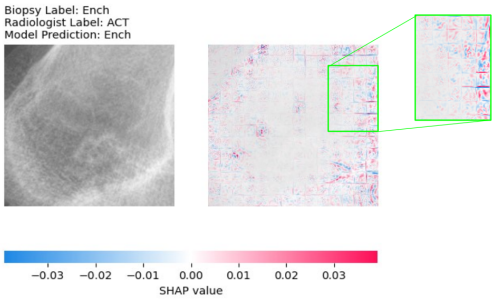
(b) SHAP values computed on misclassified test image when training with bounding boxes.

Figure 4.16.: SHAP values computed on misclassified patient case 8 with different training configurations.

Moving to Figure 4.17, we encountered a case where the model predicted the incorrect class with the original image but accurately predicted the class with the bounding box. This discrepancy implies that, with the original image, the model might not be focusing on the tumor area and might be learning from noise in the image, as indicated by the saliency map.



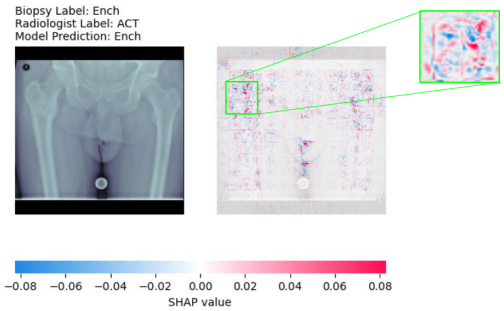
(a) SHAP values computed on misclassified test image when training with original images.



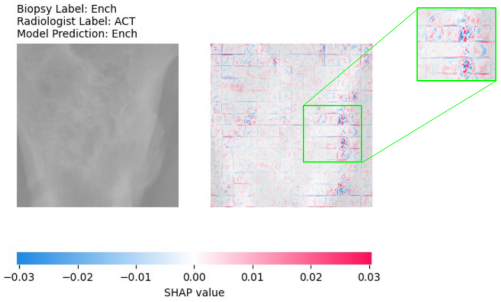
(b) SHAP values computed on misclassified test image when training with bounding boxes.

Figure 4.17.: SHAP values computed on misclassified patient case 2 with different training configurations.

In Figure 4.18, the model correctly predicted the class for both the original image and bounding box. Additionally, pixels within the tumor area contributed significantly to the correct prediction in both cases.



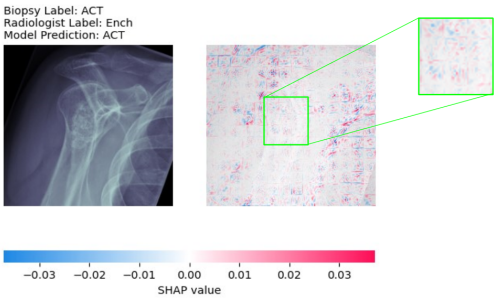
(a) SHAP values computed on misclassified test image when training with original images.



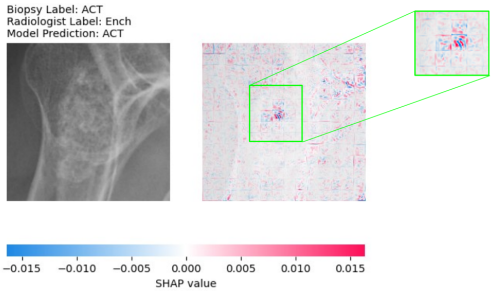
(b) SHAP values computed on misclassified test image when training with bounding boxes.

Figure 4.18.: SHAP values computed on misclassified patient case 9 with different training configurations.

Figure 4.19 showcased another instance where the model correctly predicted the class for both the original image and bounding box. However, in the original image, the model focuses on different parts of the image rather than the tumor.



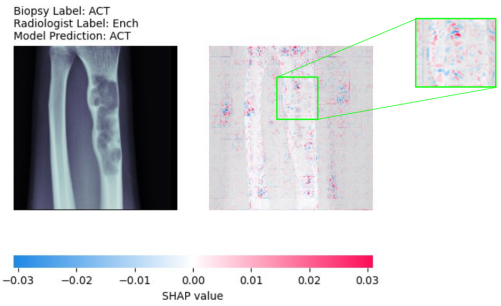
(a) SHAP values computed on misclassified test image when training with original images.



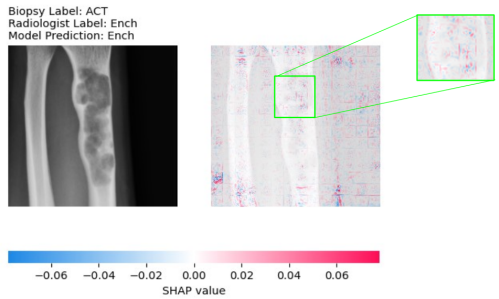
(b) SHAP values computed on misclassified test image when training with bounding boxes.

Figure 4.19.: SHAP values computed on misclassified patient case 10 with different training configurations.

Lastly, Figure 4.20 presented a scenario where the model correctly predicted the class in the original image but incorrectly predicted the class with the bounding box. The saliency map of the bounding box revealed minimal pixels in the tumor area, whereas the original image displayed some pixels with significant contributions to the correct prediction.



(a) SHAP values computed on misclassified test image when training with original images.



(b) SHAP values computed on misclassified test image when training with bounding boxes.

Figure 4.20.: SHAP values computed on misclassified patient case 11 with different training configurations.

The observed scenarios and comparisons using SHAP values suggest that, especially with the original images, the model might rely on background features. On the other hand, bounding box information can enhance the model's focus on relevant regions and improve accuracy. Instances where the model predicts the correct class for both original images and bounding boxes suggest instances where the model is more certain about its decision.

4.3. Final Framework

In this section, the final framework that yielded the highest evaluation metrics is presented. Table 4.7 summarizes the best performing model parameters. While Table 4.4, shows slightly elevated metrics when training with bounding boxes and utilizing only horizontal flipping, a comprehensive evaluation of the test set, shown in Table 4.5, reveals that employing horizontal flipping along with color jitter transformations

provides superior results. Additionally, this setup demonstrates improved outcomes in SHAP saliency maps (for example see 4.6). The original images show similar performance to training with bounding boxes, as observed in Table 4.5, however when computing SHAP we observe that most of the pixels contributing to the model prediction are outside the tumor area (for example see 4.16). As observed in Table 4.8, the model shows decent performance on the test set, accurately classifying 74% of instances. The AUC-ROC indicates a moderate discriminative performance between the two classes and the F1-Score suggests a well-balanced trade-off between FP and FN. Furthermore, the model excels in capturing positive instances (Enchondroma), as indicated by high sensitivity. However, its ability to capture negative instances (ACT) is moderate, as indicated by the specificity. This can be attributed to the higher number of Enchondroma cases in our dataset. Figure 4.21 provides further insights through the confusion matrix derived from the test set. Out of the 28 Enchondroma cases, the model correctly predicts 24. For the 11 ACT cases, the model accurately predicts 5 instances.

Table 4.7.: Best performing model parameters.

Parameter	Best Configuration
Dataset	Bounding Boxes (original X-ray images resized to 224 x 224 provided by tumor experts)
Input Size	224 x 224
Model Architecture	vit-base-patch16-224-in21k (Vision Transformer Base variant with fixed patch size 16 x 16)
Loss Function	WeightedCELoss (weight ACT = 2.90, weight Ench = 0.60)
Optimizer	AdamW ($\beta_1 = 0.9$, $\beta_2 = 0.9999$, $\epsilon = 1e-8$)
Learning Rate	0.001
Batch size	4
Num Epochs	100
Transfer Learning	Training only newly added classifier layers
Pretrained Weights	ImageNet-21k
Data Augmentation	Horizontal flipping p = 0.5, Color Jitter (contrast = 0.5, brightness = 0.2, saturation = 0.2)

Table 4.8.: Best performing model results on the test set.

Dataset	Data Augmentation	Accuracy	AUC-ROC	F1-Score	Sensitivity	Specificity
Bounding Boxes	Horizontal flip + Color Jitter	0.744	0.656	0.828	0.857	0.455

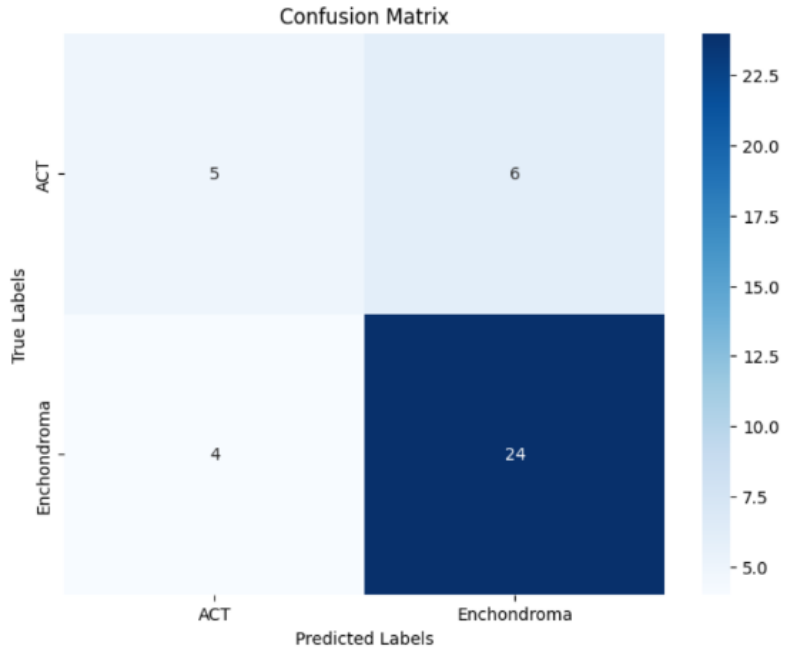


Figure 4.21.: Confusion matrix obtained on the test set with the best performing model.

5. Discussion

In this chapter the results are interpreted, breaking down the analysis into two key components: a quantitative interpretation that discusses the metrics obtained across all experiments, and a clinical evaluation, that sheds light on the SHAP results through insights provided from a panel of bone tumor experts. Then, we address the limitations and consider the generalizability of our approach.

5.1. Interpretation of Results

5.1.1. Quantitative Evaluation

The main finding of this study is that we were able to build a classification model capable of distinguishing between Enchondroma and ACT from conventional radiographs, obtaining good results. We first started experimenting with the original X-ray images and evaluated the performance of the model with different training strategies.

Impact of Hyperparameter Optimization

Through hyperparameter optimization, we saw an improvement in accuracy, AUC-ROC, F1-Score, and sensitivity, as shown in Table 4.1. The highest improvement is in the sensitivity, which increases by 3.5%. This suggests a better ability of the model to correctly identify Enchondroma cases. However, we saw a decrease of 2.8% in specificity after optimization. This is because sensitivity and specificity often have an inverse relationship, so improving sensitivity might lead to a decrease in specificity.

Impact of Transfer Learning

Fine-tuning solely the newly added classifier layers led to an increase of 9.1% in specificity, as seen in Table 4.2. This suggests that the model becomes more adept at capturing instances of ACT, leading to a significant reduction in false positives. Consequently, this approach is more effective for our task since it provides a good balance between sensitivity and specificity. By freezing the weights of the pre-trained model during fine-tuning, we harnessed its learned hierarchical features from a larger dataset,

adapting its understanding to our specific data characteristics, a crucial advantage with limited datasets.

Impact of Data Augmentation

Diverse data augmentation techniques were explored, revealing that horizontal flipping notably enhanced the model's performance across various metrics, as seen in Table 4.3. This technique enables the model to capture both positive and negative instances effectively, as shown by the high values of F1-Score, sensitivity, and specificity. The efficacy of horizontal flipping may be attributed to the symmetry in anatomy, providing diverse perspectives crucial for better generalization. A similar performance boost was observed with color jitter transformations, which enhance the visibility of anatomical structures, making it easier for the model to detect important features. In contrast, with random rotation, while enhancing diversity, it might not always lead to improved performance. The decrease of specificity by 27.9% suggests that the model may be prone to false positives, potentially leading to misclassifications.

Impact of training with Original Images and Bounding Boxes

When comparing performance between models trained on original images and bounding boxes, the results indicate a slightly superior performance for the model trained on original images, potentially attributed to the increased dataset size. Results can be observed in Table 4.4. With more samples, there is a potential for increased diversity. This diversity might allow the model to learn a broader range of features and patterns, contributing to its overall performance. However, despite having fewer samples, the model trained on bounding boxes performs well, especially with high sensitivity. Overall, the model demonstrates similar performance when trained with both datasets. When evaluating the model on the test set for both original and bounding boxes, employing the two most effective data augmentation techniques, horizontal flipping + color jitter transformations showed improved performance, especially in terms of specificity. These results are shown in Table 4.5. Furthermore Table 4.6 shows that assessing the model on the misclassified test set, compared to a random split, enhances model performance, especially in specificity and AUC-ROC, when trained exclusively on correctly classified samples. This improvement in specificity suggests an increased accuracy in identifying true negative cases. The observed differences in model performance can be attributed to the nature of the data used for training. In the second scenario, where correct radiologist annotations are used, the training data may be more consistent and reliable. Consequently, the model might be exposed to fewer challenging and confusing samples, leading to a better generalization.

5.1.2. Clinical Evaluation

To obtain clinical insights from our SHAP results, a panel of bone tumor experts, comprising a radiologist and two orthopedic surgeons, conducted a thorough clinical evaluation of selected interesting cases. The panel assessed 29 patient cases, each presenting an X-ray image alongside corresponding SHAP saliency maps overlaid onto the image. The showcased information included the biopsy label (our ground-truth), radiologist label, and the model prediction. To simplify SHAP visualization and focus on pixels with the highest contribution to the model's prediction, we applied the previously explained threshold mechanism (see Subsection 3.2.2) to the SHAP saliency maps. We presented 22 cases where the model correctly predicted the class and 7 cases where predictions were inaccurate. Additionally, 2 cases were misclassified by the radiologist, aligning with the model's incorrect prediction. With this evaluation we wanted to see if the model is looking at clinically relevant tumor regions. Among the cases presented, the experts identified few instances where the model highlighted pixels in the tumor's calcification, a significant indicator of Enchondroma. However, in some cases, no discernible clinical findings could be extracted. The experts concluded that no consistent pattern aiding the differentiation between Enchondroma and ACT was evident. Nevertheless, they observed that, in most instances, our model focused on areas within the tumor that typically do not capture the attention from radiologists. While no definitive patterns emerged from this evaluation, the experts expressed the notion that with an expanded dataset, more precise conclusions could potentially be drawn. They speculated that the model might provide novel insights or information that could assist clinicians in effectively distinguishing between these two entities.

5.2. Limitations

In the following section, the limitations of our approach are elaborated. One of the main limitations is the limited dataset available. With only 499 images in the original X-ray images dataset and 354 in the bounding boxes dataset, diversity is limited. A small dataset may not fully capture the diversity of clinical cases, especially with challenging cases where the X-ray images are very similar for both entities. With a small dataset, there is an increased risk of overfitting, where the model may memorize specific details of the training examples rather than learning generalizable patterns. Moreover, our inability to test the model with datasets from different clinics poses another significant limitation. Privacy concerns and the rarity of bone tumors limit the sharing of medical datasets. Another limitation is the issue of class imbalance, which affects the model's ability to learn from both classes equally. While the model performs well in identifying Enchondroma cases, its efficacy decreases in detecting ACT cases. A third limitation

arises from our exclusive reliance on conventional radiographic images, which are heterogeneous and may contain noise. Incorporating additional imaging modalities such as CT and MRI could enhance performance and detect entity-specific features. The experts also suggested integrating clinical data, particularly tumor location, as a potential performance enhancer. However, with this approach, there is still difficulties in differentiating these entities when the tumor is located in the long bones. With our best performing model, we achieved an accuracy of 0.744 which is not good enough for clinical usage, however, there is room for improvement. Perhaps the model architecture chosen may not be suited to our task. ViT often requires a large amount of data to perform optimally and training them with a small dataset may not allow the model to fully exploit the potential benefits of their architecture. Therefore, it is recommended to explore alternative CNN architectures which could potentially yield better results. In the interpretability domain, SHAP values, while revealing valuable insights, pose challenges due to their complexity and difficulty in interpretation. Notably, the presence of red and blue pixels adjacent to each other in many cases suggests that blue pixels contribute to the other class, introducing confusion. Given the noise and heterogeneity inherent in X-ray images, SHAP values may not be the optimal method for this task. Consequently, applying alternative Explainable AI techniques might offer a more suitable solution for enhancing interpretability in this specific context.

5.3. Generalizability

Our approach can be applicable to many scenarios since we are only using conventional X-ray images. The primary requirement for its deployment is an annotated dataset and images rescaled to 224 x 224 pixels. In the cases where the images have a lower resolution, an alternative network should be considered. Our approach is well-suited for similar classification tasks in orthopedics and radiology. Furthermore, its adaptability extends to multi-class classification tasks, requiring a simple adjustment in the number of classes within the newly added classifier layers. The only step needed is fine-tuning the model to the specific task by training the added classifier layers. This is very fast in terms of training time. Another advantage of using a pre-trained model is that a good performance can be achieved even with small datasets, a crucial attribute in the context of the medical domain, where datasets tend to be inherently limited. The versatility of our approach extends beyond bone tumor classification, encompassing a spectrum of medical imaging tasks. This includes the identification of abnormalities in X-rays, CT scans, or MR images. The interpretability aspect, facilitated by SHAP values and the model-agnostic Gradient Explainer, makes it applicable to any kind of ML model. Our approach is therefore applicable to many medical scenarios, especially in addressing

the growing demand for understandable and transparent AI systems in the medical domain.

6. Conclusion

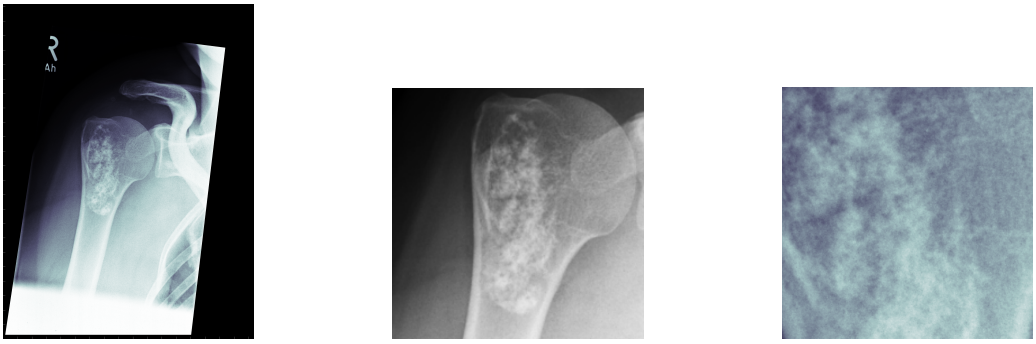
With this study, we have established the first DL-approach which addresses the complex task of distinguishing Enchondroma and ACT from X-ray images. We were able to achieve notable metrics, including an accuracy of 0.744, sensitivity of 0.857, and specificity of 0.455 with the Vision Transformer model, on the random train/validation/test split when training with the bounding boxes. Similar performance was observed with the original images, with several instances where the model accurately identified the tumor in the image. However, for this task it is better to use the bounding boxes since as we have seen in the SHAP results, the pixels that contribute to the model's prediction appear inside the tumor area. Conversely, when training with original images in most cases the model drives its attention to non-diagnostic regions like the background. In the evaluation using misclassified samples by the radiologist as the test set, our model correctly predicted 7 out of 10 misclassified Enchondroma cases and 3 out of 14 misclassified ACT cases. This is explained by the imbalance present in our dataset. While consistent metrics are obtained across various experiments, the incorporation of color jitter transformations stands out, showcasing significant improvement in SHAP results. Following this line, we suggest exploring further image processing techniques to enhance the visualization of X-ray images. This could potentially improve the performance of the model. Additionally, bone tumor experts confirmed that SHAP saliency maps are complex to understand and that they might not be the optimal explainable AI method when dealing with limited and heterogeneous data. However, in some cases, the model focuses on interesting features of the tumor such as calcifications or borders of the tumor which are good indicators of Enchondroma. Careful clinical evaluation is needed for our interpretable results, especially in the cases where the model's prediction aligns with the ground truth while diverging from the radiologist's annotation. This could help in understanding the uncertain features. With further improvements and refinements commented through this work, our model could evolve into a precise tool for the initial assessment and management of patients with cartilaginous tumors.

This work shows multiple paths for future research. One dataset left unexplored in this study is the 'only_tumor' images dataset. The decision to exclude these images stems from the challenge of rescaling them to 224 x 224 without compromising resolution. Also, for our task, it was crucial to observe the borders of the tumor which were

not observed in most of these images. Nevertheless, an alternative model which accepts smaller input sizes, could be trained on this dataset. Applying SHAP to such a model holds the potential to extract more features inside the specific tumor area, unveiling characteristics that might have been previously overlooked. Another promising area for exploration involves an in-depth analysis of image processing techniques to enhance X-ray image visualization. Our observations indicated improved performance with color jitter transformations. While time constraints limited our exploration of hyperparameter tuning for bounding boxes, it remains an aspect demanding further investigation. The parameters derived from the original images may not be the most optimal for bounding boxes, necessitating dedicated optimization. Furthermore, we recommend exploring other model architectures, especially considering the efficacy of CNNs in handling smaller datasets. The incorporation of additional imaging modalities, such as CT and MRI, presents another promising path for future research, holding promise for unraveling more specific clinical features. To enrich the model with additional information, a multi-modal approach, as proposed by Günther et al. [29], could enhance performance. By integrating clinical patient information and demographics like age, sex, and tumor localization, the model could achieve greater accuracy in its predictions. Finally, to assess the robustness of our approach, external validation on diverse datasets is paramount, ensuring its generalizability across various clinical contexts.

A. Patient Cases Examples

The subsequent section showcases additional patient cases from our dataset. Each case is presented with the three sizes provided: the original X-ray image, the bounding box, and the image showing only the tumor.

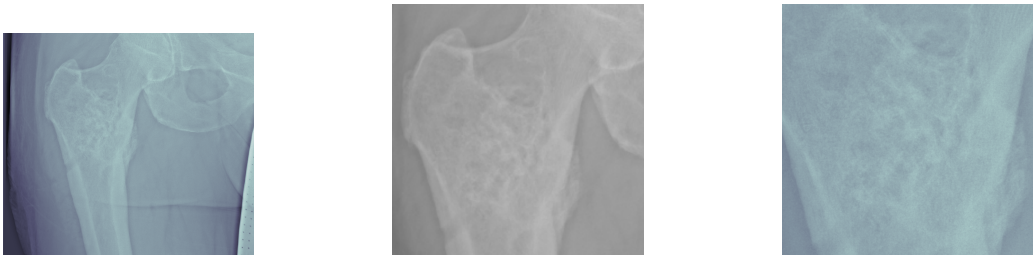


(a) Original X-ray image. (b) Bounding Box. (c) Only tumor.

Figure A.1.: Patient case example 1 labeled as Enchondroma by both the radiologist and the biopsy.

(a) Original X-ray image. (b) Bounding Box. (c) Only tumor.

Figure A.2.: Patient case example 2 labeled as ACT by both the radiologist and the biopsy.



(a) Original X-ray image. (b) Bounding Box. (c) Only tumor.

Figure A.3.: Patient case example 3 labeled as ACT by the radiologist but the biopsy confirms is an Enchondroma.

(a) Original X-ray image. (b) Bounding Box. (c) Only tumor.

Figure A.4.: Patient case example 4 labeled as Enchondroma by the radiologist but the biopsy confirms is an ACT.

B. Further Experiments

B.1. Experiment 1: Loss Functions

In our experiments, we employed different loss functions, starting with CELoss and later exploring WeightedCELoss to address class imbalance. For WeightedCELoss we assigned a weight of 2.94 for the minority class (ACT) and 0.61 for the majority class (Enchondroma). While CELoss yielded superior results in some metrics, it is recommended to use WeightedCELoss. This recommendation is based on the observation that CELoss tends to skew model predictions toward the most frequent class. The parameters utilized for these experiments are detailed below:

- Num Epochs = 100
- Adam Optimizer
- Learning Rate = 1e-6
- Scheduler (step_size = 10, gamma = 0.5)
- Batch Size = 4
- Data Split 80/10/10
- No data augmentation
- Transfer Learning: fine-tuning only the classifier layers

Table B.1.: Model's performance when training with CELoss, and WeightedCELoss. The metrics are computed based on a single train/validation/test split. These metrics represent the vaildation results.

Loss Function	Accuracy	AUC-ROC	F1-Score	Sensitivity	Specificity
CELoss	0.698	0.504	0.795	0.907	0.101
WeightedCELoss	0.693	0.478	0.817	0.925	0.03

B.2. Experiment 2: Data Split

We carried out additional experiments with two distinct data splits to expand the validation set, achieving improved results with a split ratio of 70/20/10. The parameters employed for these experiments are as follows:

- Num Epochs = 100
- Adam Optimizer
- Learning Rate = 1e-6
- Scheduler (step_size = 10, gamma = 0.5)
- WeightedCELoss (2.94, 0.61)
- Batch Size = 4
- No data augmentation
- Transfer Learning: fine-tuning only the classifier layers

Table B.2.: Model's performance when training with different data splits. The metrics are computed based on a single train/validation/test split. These metrics represent the vaildation results.

Data Split	Accuracy	AUC-ROC	F1-Score	Sensitivity	Specificity
80/10/10	0.693	0.478	0.817	0.925	0.03
70/20/10	0.803	0.525	0.889	0.971	0.079

B.3. Experiment 3: Data Augmentation

In this experiment, we explored diverse data augmentation techniques to expand the dataset size, employing both single and multiple augmentation strategies. Notably, our findings reveal that incorporating a vertical flip in our case led to a decrease in performance across all metrics, except for specificity. This unusual specificity result suggests that the model may not be effectively learning from specific features associated with this entity. Consequently, we do not recommend using vertical flipping as data augmentation for X-ray images. The parameters utilized for these strategies are detailed below:

- Num Epochs = 100
- Adam Optimizer
- Learning Rate = 1e-6
- Scheduler (step_size = 10, gamma = 0.5)
- WeightedCELoss (2.94, 0.61)
- Batch Size = 4
- Data Split 70/20/10
- Transfer Learning: fine-tuning only the classifier layers

Table B.3.: Model's performance when training with different data augmentation techniques. The metrics are computed based on a single train/validation/test split. These metrics represent the vaildation results.

Data Augmentation	Probability	Accuracy	AUC-ROC	F1-Score	Sensitivity	Specificity
Horizontal flip	p = 0.2	0.825	0.595	0.900	0.964	0.226
	p = 0.3	0.799	0.512	0.887	0.972	0.053
Horizontal + vertical flip	p = 0.2	0.222	0.501	0.100	0.054	0.947

B.4. Experiment 4: Learning Rate

We conducted experiments involving grid search to assess various learning rates. Table B.4 presents consistent evaluation metrics across all the learning rates tested. Prioritizing a balanced trade-off between sensitivity and specificity, we selected a Learning Rate of 1e-3 for our final framework. The parameters employed are as follows:

- Num Epochs = 200
- Adam Optimizer
- WeightedCELoss (2.94, 0.61)
- Batch Size = 4

- Data Split 70/20/10
- Data Augmentation: horizontal flip p = 0.2
- Transfer Learning: fine-tuning only the classifier layers

Table B.4.: Model's performance when training with different learning rates. The metrics are computed based on a single train/validation/test split. These metrics represent the vaildation results.

Learning Rate	Accuracy	AUC-ROC	F1-Score	Sensitivity	Specificity
1e-3	0.788	0.609	0.872	0.896	0.322
1e-4	0.787	0.592	0.873	0.905	0.279
1e-5	0.807	0.522	0.892	0.979	0.064
1e-6	0.807	0.500	0.893	0.993	0.007

B.5. Experiment 5: Batch Size

We conducted experiments employing grid search to evaluate different batch sizes. As illustrated in Table B.5, using a batch size of 4 demonstrated performance improvement across almost all metrics, achieving a favorable balance between sensitivity and specificity. The parameters utilized for this configuration are detailed below:

- Num Epochs = 100
- Adam Optimizer (weight decay = 1e-2)
- Learning Rate = 1e-3
- WeightedCELoss (2.94, 0.61)
- Data Split 70/20/10
- Data Augmentation: horizontal flip p = 0.2
- Transfer Learning: fine-tuning only the classifier layers

Table B.5.: Model's performance when training with different batch sizes. The metrics are computed based on a single train/validation/test split. These metrics represent the vaildation results.

Batch Size	Accuracy	AUC-ROC	F1-Score	Sensitivity	Specificity
4	0.792	0.616	0.875	0.899	0.333
8	0.762	0.607	0.853	0.856	0.358
16	0.736	0.597	0.834	0.820	0.373

C. More SHAP Results

C.1. Original X-ray Images

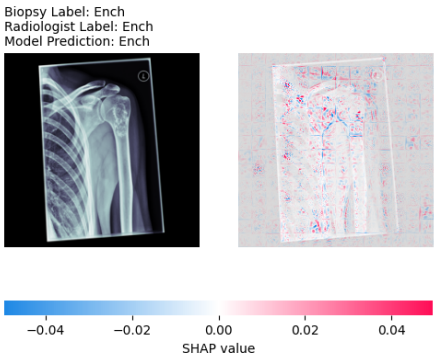


Figure C.1.: SHAP results sample 25.

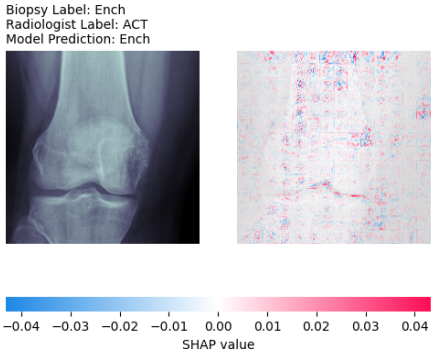


Figure C.2.: SHAP results sample 31.

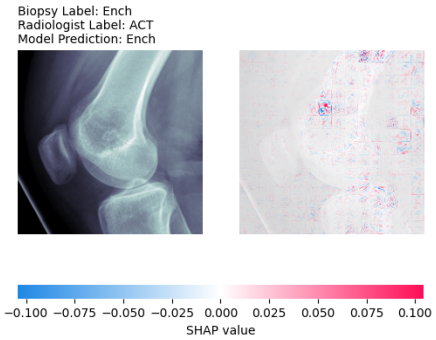


Figure C.3.: SHAP results sample 32.

C.2. Bounding Boxes

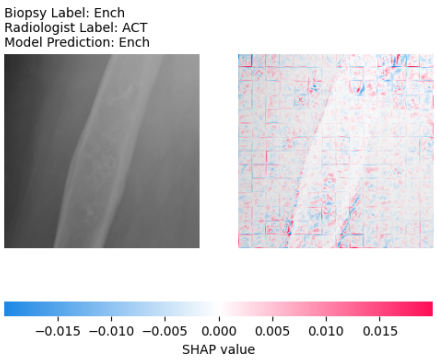


Figure C.4.: SHAP results sample 0.

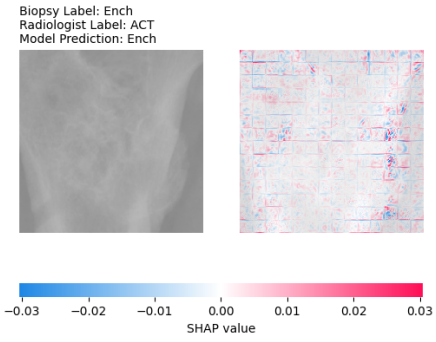


Figure C.5.: SHAP results sample 4.

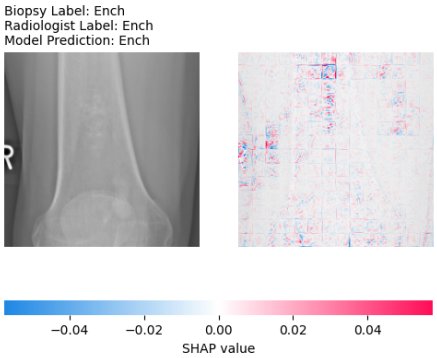


Figure C.6.: SHAP results sample 12.

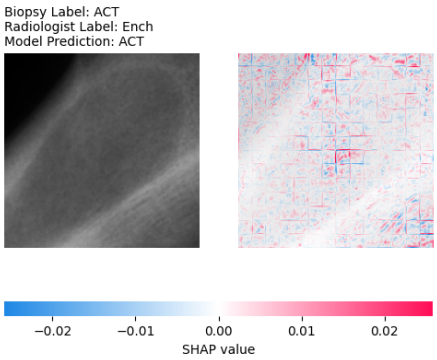


Figure C.7.: SHAP results sample 16.

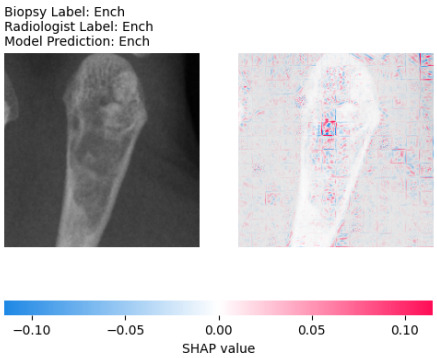


Figure C.8.: SHAP results sample 25.

C.3. Threshold

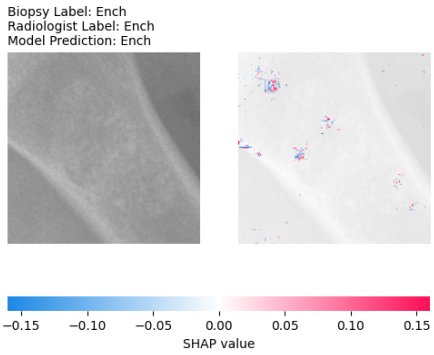


Figure C.9.: SHAP results sample 8.

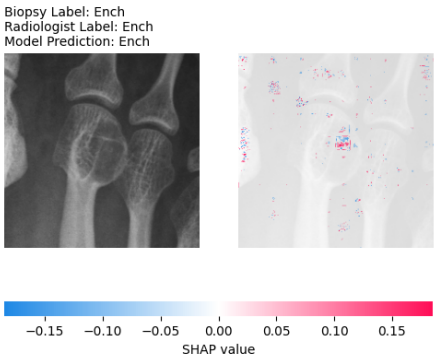


Figure C.10.: SHAP results sample 14.

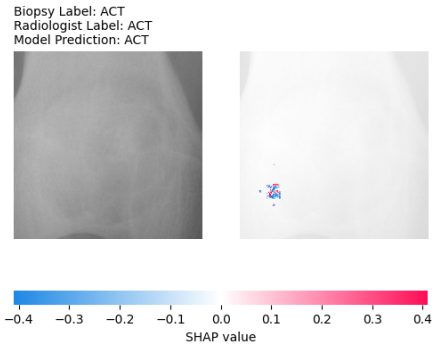


Figure C.11.: SHAP results sample 33.

List of Figures

List of Tables

Bibliography

[1] N. AI. *24 Evaluation Metrics for Binary Classification (And When to Use Them)*. Available online at https://neptune.ai/blog/evaluation-metrics-binary-classification. Accessed: 2023-10-26.

[2] J. Alammar. *The Illustrated Transformer*. Available online at https://jalammar.github.io/illustrated-transformer/. Accessed: 2023-10-18.

[3] American Academy of Orthopaedic Surgeons (AAOS). *Bone Tumor*. Available online at https://orthoinfo.aaos.org/en/diseases--conditions/bone-tumor/. Accessed: 2023-09-26.

[4] American Academy of Orthopaedic Surgeons (AAOS). *Enchondroma*. Available online at https://orthoinfo.aaos.org/en/diseases--conditions/enchondroma/. Accessed: 2023-09-30.

[5] F. Amin and M. Mahmoud. "Confusion Matrix in Binary Classification Problems: A Step-by-Step Tutorial." In: *Journal of Engineering Research* 6.5 (2022), pp. 0–0.

[6] M. Bhandari, T. B. Shahi, B. Siku, and A. Neupane. "Explanatory classification of CXR images into COVID-19, Pneumonia and Tuberculosis using deep learning and XAI." In: *Computers in Biology and Medicine* 150 (2022), p. 106156.

[7] B. Bhinder, C. Gilvary, N. S. Madhukar, and O. Elemento. "Artificial intelligence in cancer research and precision medicine." In: *Cancer discovery* 11.4 (2021), pp. 900–915.

[8] M. Bloier. *Segmentation of Bone Tumors with Methods of Deep Learning*. 2022.

[9] T. J. Bradshaw, Z. Huemann, J. Hu, and A. Rahmim. "A Guide to Cross-Validation for Artificial Intelligence in Medical Imaging." In: *Radiology: Artificial Intelligence* (2023), e220232.

[10] A. Bria, C. Marrocco, and F. Tortorella. "Addressing class imbalance in deep learning for small lesion detection on medical images." In: *Computers in biology and medicine* 120 (2020), p. 103735.

[11] F. Chollet. *Deep learning with Python*. Simon and Schuster, 2021.

[12] A. Choulwar. *The Art of Convolutional Neural Network*. Available online at `https://medium.com/@achoulwar901/the-art-of-convolutional-neural-network-abda56dba55c`. Accessed: 2023-11-21.

[13] C. M. Costelloe and J. E. Madewell. "Radiography in the initial diagnosis of primary bone tumors." In: *American Journal of Roentgenology* 200.1 (2013), pp. 3–7.

[14] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. "Bert: Pre-training of deep bidirectional transformers for language understanding." In: *arXiv preprint arXiv:1810.04805* (2018).

[15] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. "An image is worth 16x16 words: Transformers for image recognition at scale." In: *arXiv preprint arXiv:2010.11929* (2020).

[16] H. Engel, G. W. Herget, H. Füllgraf, R. Sutter, M. Benndorf, F. Bamberg, and P. M. Jungmann. "Chondrogenic bone tumors: the importance of imaging characteristics." In: *RöFo-Fortschritte auf dem Gebiet der Röntgenstrahlen und der bildgebenden Verfahren*. Vol. 193. 03. Georg Thieme Verlag KG. 2021, pp. 262–275.

[17] F. Erdem, İ. Tamsel, and G. Demirpolat. "The use of radiomics and machine learning for the differentiation of chondrosarcoma from enchondroma." In: *Journal of Clinical Ultrasound* (2023).

[18] A. Esteva, A. Robicquet, B. Ramsundar, V. Kuleshov, M. DePristo, K. Chou, C. Cui, G. Corrado, S. Thrun, and J. Dean. "A guide to deep learning in healthcare." In: *Nature medicine* 25.1 (2019), pp. 24–29.

[19] J. Feger and T. Foster. *Central atypical cartilaginous tumor low-grade chondrosarcoma*. Available online at `https://radiopaedia.org/articles/central-atypical-cartilaginous-tumour-low-grade-chondrosarcoma`. Accessed: 2023-10-3.

[20] C. D. M. Fletcher et al. *WHO Classification of Tumours: Soft Tissue and Bone Tumours*. 5th. Vol. 3. World Health Organization Classification of Tumours. International Agency for Research on Cancer, 2020.

[21] F. Gaillard and J. Feger. *Enchondroma vs low grade chondrosarcoma*. Available online at `https://radiopaedia.org/articles/enchondroma-vs-low-grade-chondrosarcoma-3`. Accessed: 2023-09-26.

[22] C. Garbin. *Exploring SHAP explanations for image classification*. Available online at `https://cgarbin.github.io/shap-experiments-image-classification/`. Accessed: 2023-10-26.

[23] F. G. Gassert, S. Breden, J. Neumann, F. T. Gassert, C. Bollwein, C. Knebel, U. Lenze, R. von Eisenhart-Rothe, C. Mogler, M. R. Makowski, et al. "Differentiating Enchondromas and Atypical Cartilaginous Tumors in Long Bones with Computed Tomography and Magnetic Resonance Imaging." In: *Diagnostics* 12.9 (2022), p. 2186.

[24] Y. Gavrilova. *Convolutional Neural Networks for Beginners*. Available online at `https://serokell.io/blog/introduction-to-convolutional-neural-networks`. Accessed: 2023-10-17.

[25] A. Géron. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow*. " O'Reilly Media, Inc.", 2022.

[26] S. Ghosh. *The Ultimate Guide to Evaluation and Selection of Models in Machine Learning*. Available online at `https://neptune.ai/blog/ml-model-evaluation-and-selection`. Accessed: 2023-11-21.

[27] R. J. Gillies, P. E. Kinahan, and H. Hricak. "Radiomics: images are more than pictures, they are data." In: *Radiology* 278.2 (2016), pp. 563–577.

[28] J. C. Gore. *Artificial intelligence in medical imaging*. 2020.

[29] M. Günther. *Multimodal Deep Learning Classification of Bone Tumors*. 2023.

[30] M. Hakama, A. Pokhrel, N. Malila, and T. Hakulinen. "Sensitivity, effect and overdiagnosis in screening for cancers with detectable pre-invasive phase." In: *International Journal of Cancer* 136.4 (2015), pp. 928–935.

[31] Y. He, I. Pan, B. Bao, K. Halsey, M. Chang, H. Liu, S. Peng, R. A. Sebro, J. Guan, T. Yi, et al. "Deep learning-based classification of primary bone tumors on radiographs: A preliminary study." In: *EBioMedicine* 62 (2020).

[32] S. A. Hicks, I. Strümke, V. Thambawita, M. Hammou, M. A. Riegler, P. Halvorsen, and S. Parasa. "On evaluation metrics for medical applications of artificial intelligence." In: *Scientific reports* 12.1 (2022), p. 5979.

[33] S. Hwang, M. Hameed, and M. Kransdorf. "The 2020 World Health Organization classification of bone tumors: what radiologists should know." In: *Skeletal Radiology* (2022).

[34] IBM. *What are convolutional neural networks?* Available online at `https://www.ibm.com/topics/convolutional-neural-networks`. Accessed: 2023-10-17.

[35] U. Kamath and J. Liu. *Explainable artificial intelligence: An introduction to interpretable machine learning*. Springer, 2021.

[36] A. Kayid. "Explaining what learned models predict: In which cases can we trust machine learning models and when is caution required?" In: (2020).

[37] D. P. Kingma and J. Ba. "Adam: A method for stochastic optimization." In: *arXiv preprint arXiv:1412.6980* (2014).

[38] V. Kosar. *Transformer Embeddings and Tokenization*. Available online at `https://vaclavkosar.com/ml/transformer-embeddings-and-tokenization`. Accessed: 2023-10-18.

[39] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. "Gradient-based learning applied to document recognition." In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324.

[40] P. Lippe. *Tutorial 15: Vision Transformers*. Available online at `https://uvadlc-notebooks.readthedocs.io/en/latest/tutorial_notebooks/tutorial15/Vision_Transformer.html`. Accessed: 2023-10-19.

[41] R. Liu, D. Pan, Y. Xu, H. Zeng, Z. He, J. Lin, W. Zeng, Z. Wu, Z. Luo, G. Qin, et al. "A deep learning-machine learning fusion approach for the classification of benign, malignant, and intermediate bone tumors." In: *European Radiology* 32.2 (2022), pp. 1371–1383.

[42] S. M. Lundberg and S.-I. Lee. "A unified approach to interpreting model predictions." In: *Advances in neural information processing systems* 30 (2017).

[43] A. Luque, A. Carrasco, A. Martın, and A. de Las Heras. "The impact of class imbalance in classification performance metrics based on the binary confusion matrix." In: *Pattern Recognition* 91 (2019), pp. 216–231.

[44] S. Mahapatra. *Why Deep Learning over Traditional Machine Learning?* Available online at `https://towardsdatascience.com/why-deep-learning-is-needed-over-traditional-machine-learning-1b6a99177063`. Accessed: 2023-10-2.

[45] P. Marcelino. *Transfer Learning from Pre-Trained Models*. Available online at `https://towardsdatascience.com/transfer-learning-from-pre-trained-models-f2393f124751`. Accessed: 2023-10-20.

[46] J. Maurıcio, I. Domingues, and J. Bernardino. "Comparing Vision Transformers and Convolutional Neural Networks for Image Classification: A Literature Review." In: *Applied Sciences* 13.9 (2023), p. 5521.

[47] G. B. Medicine. *Measuring Performance: AUC (AUROC)*. Available online at `https://glassboxmedicine.com/2019/02/23/measuring-performance-auc-auroc/`. Accessed: 2023-10-26.

[48] M. Mishra. *Convolutional Neural Networks, Explained*. Available online at `https://towardsdatascience.com/convolutional-neural-networks-explained-9cc5188c4939`. Accessed: 2023-10-16.

[49] K. P. Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

[50] P. Picci, M. Manfrini, D. M. Donati, M. Gambarotti, A. Righi, D. Vanel, and A. P. D. Tos. *Diagnosis of Musculoskeletal Tumors and Tumor-like Conditions: Clinical, Radiological and Histological Correlations - The Razzoli Case Archive*. Springer, 2020.

[51] Pinecone. *Vision Transformers (ViT) Explained*. Available online at `https://www.pinecone.io/learn/series/image-search/vision-transformers/`. Accessed: 2023-10-18.

[52] D. Podareanu, V. Codreanu, G. C. van Leeuwen, and V. Weinberg. "Best practice guide-deep learning." In: *Partnership for Advanced Computing in Europe (PRACE), Tech. Rep* 2 (2019).

[53] T. Qin and T. Qin. "Deep Learning Basics." In: *Dual Learning* (2020), pp. 25–46.

[54] S. Ravi, S. Khoshrou, and M. Pechenizkiy. "ViDi: descriptive visual data clustering as radiologist assistant in COVID-19 streamline diagnostic." In: *arXiv preprint arXiv:2011.14871* (2020).

[55] M. R. Rezaei-Dastjerdehei, A. Mijani, and E. Fatemizadeh. "Addressing imbalance in multi-label classification using weighted cross entropy loss function." In: *2020 27th National and 5th International Iranian Conference on Biomedical Engineering (ICBME)*. IEEE. 2020, pp. 333–338.

[56] S. Saran and R. H. Phulware. "World health organization classification of bone tumors (fifth edition): What a radiologist needs to know?" In: *Indian Journal of Musculoskeletal Radiology* (2022).

[57] S. Serinelli and G. de la Roza. *Bone and joints Chondrosarcoma Atypical cartilaginous tumor chondrosarcoma, grade 1*. Available online at `https://pathologyoutlines.com/topic/boneatypicalcartilaginoustumor.html`. Accessed: 2023-09-30.

[58] A. W. Services. *What is Overfitting?* Available online at `https://aws.amazon.com/what-is/overfitting/?nc1=h_ls`. Accessed: 2023-10-20.

[59] F. Shamshad, S. Khan, S. W. Zamir, M. H. Khan, M. Hayat, F. S. Khan, and H. Fu. "Transformers in medical imaging: A survey." In: *Medical Image Analysis* (2023), p. 102802.

[60] SHAP. *SHAP*. Available online at `https://github.com/shap/shap`. Accessed: 2023-12-08.

[61] C. Shorten and T. M. Khoshgoftaar. "A survey on image data augmentation for deep learning." In: *Journal of big data* 6.1 (2019), pp. 1–48.

[62] A. Singh, S. Sengupta, and V. Lakshminarayanan. "Explainable deep learning models in medical image analysis." In: *Journal of imaging* 6.6 (2020), p. 52.

[63] E. W. Steyerberg, S. E. Bleeker, H. A. Moll, D. E. Grobbee, and K. G. Moons. "Internal and external validation of predictive models: a simulation study of bias and precision in small samples." In: *Journal of clinical epidemiology* 56.5 (2003), pp. 441–447.

[64] P. Sturmfels, S. Lundberg, and S.-I. Lee. *Visualizing the Impact of Feature Attribution Baselines*. Available online at `https://distill.pub/2020/attribution-baselines/`. Accessed: 2023-10-26.

[65] R. Szeliski. *Computer vision: algorithms and applications*. Springer Nature, 2022.

[66] Q. Teng, Z. Liu, Y. Song, K. Han, and Y. Lu. "A survey on the interpretability of deep learning in medical diagnosis." In: *Multimedia Systems* 28.6 (2022), pp. 2335–2355.

[67] TensorFlow. *Transfer learning and fine-tuning*. Available online at `https://www.tensorflow.org/tutorials/images/transfer_learning`. Accessed: 2023-10-20.

[68] S. University. *CS231n: Deep Learning for Computer Vision*. Available online at `http://cs231n.stanford.edu/`. Accessed: 2023-10-10.

[69] S. University. *Lecture 5: Convolutional Neural Networks*. Available online at `http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture5.pdf`. Accessed: 2023-10-17.

[70] S. University. *Transfer Learning*. Available online at `https://cs231n.github.io/transfer-learning/`. Accessed: 2023-10-20.

[71] G. Varoquaux and V. Cheplygina. "Machine learning for medical imaging: methodological failures and recommendations for the future." In: *NPJ digital medicine* 5.1 (2022), p. 48.

[72] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. "Attention is all you need." In: *Advances in neural information processing systems* 30 (2017).