國立臺灣大學電機資訊學院資訊工程學系

碩士論文

Department of Computer Science and Information Engineering

College of Electrical Engineering and Computer Science

National Taiwan University

Master Thesis

利用雙能量 CT 之 ConvNeXt 肺癌存活率預測

Lung Cancer Survival Prediction using Dual Energy CT

楊朝勛

Chao-Hsun Yang

指導教授：張瑞峰 博士

Advisor: Ruey-Feng Chang, Ph.D.

中華民國 112 年 7 月

July 2023

口試委員會審定書

國立臺灣大學碩士學位論文
口試委員會審定書
MASTER'S THESIS ACCEPTANCE CERTIFICATE
NATIONAL TAIWAN UNIVERSITY

利用雙能量 CT 之 ConvNeXt 肺癌存活率預測

Lung Cancer Survival Prediction Using Dual Energy CT

本論文係楊朝勛君（學號 R10922148）在國立臺灣大學資訊工程學系完成之碩士學位論文，於民國 112 年 7 月 26 日承下列考試委員審查通過及口試及格，特此證明。

The undersigned, appointed by the Department of Computer Science and Information Engineering on 26 July 2023 have examined a Master's thesis entitled above presented by YANG, CHAO-HSUN (student ID: R10922148) candidate and hereby certify that it is worthy of acceptance.

口試委員 Oral examination committee:

張瑞峰　羅崇銘　陳炳宇

（指導教授 Advisor）

系主任/所長 Director: 洪士灝

致謝

時光匆匆，倒數的日子歷歷在目，看著日漸厚實的論文，除了肯定自己的執著不懈外，也要感謝許多人的協助與支持。

感謝指導老師張瑞峰教授，在每次的 group meeting 中給我建議，也保持開會的高效，提供良好的實驗設備，讓我能夠將時間與精力專注在論文，每次遇到問題需要聯絡老師時，老師總是第一時間回覆，讓我感受到老師的認真與用心，非常幸運進入老師的實驗室。

接著要感謝宥崴學長，陪我到醫院蒐集資料，了解我實驗中遇到的這種狀況並一一給予方向與建議，也會時常關心這陣子有沒有壓力或遭遇到任何不順心的事。開始撰寫論文後，也幫助我更改不正式的寫法，使我了解論文寫作的獨特之處。

也感謝在實驗室遇到的每位同學，透過修課、聚餐時分享課業、工作、生活、旅遊等，讓我收穫甚多。很慶幸有一群在同樣環境下努力同一件事的朋友，使我不會那麼孤單，學業上的困擾也有人聽，有人懂。

感謝家人支持我完成碩士學位，雖然還沒正式打響尾聲，不過對未來的期待之餘還是有些感傷，回首兩年的回憶慶幸自己過得充實，在台大認識許多厲害、有趣、特別的人，帶著與大家交流激盪後嶄新的自己，在未來繼續努力向陽！

摘要

肺癌是全世界最致命的癌症，手術後的生存率非常低，主要原因是即使不同患者存在相同的臨床與病理資訊，他們的三年、五年存活率也不一定相同。目前肺癌分期是醫生診斷及給予後續治療（如腫瘤切除或放射治療）的主要指標，若缺乏其他輔助資訊，則可能影響治療決策的品質，因此引入存活率預測作為診斷的支持性指標非常重要。本研究中，我們提出了一種結合注意力機制（Attention block）的深度學習模型，並整合雙能量電腦斷層掃描（dual energy CT）和臨床、病理資料進行 3 年後存活結果預測。

我們將 ConvNeXt[1]作為基底模型，並利用 Attention block 從腫瘤紋理中篩選特徵，再透過 Damper block[2]結合影像特徵與腫瘤大小資訊，最終和臨床與病理資料合併訓練，得到 3 年後存活預測結果。通過實驗，結合 ConvNeXt[1]、Attention block 及 Damper block[2]的準確率為 86.03%，靈敏度為 82.86%，特異度為 86.69%。Attention block 採用了 Squeeze-and-Excitation （SE）[3]和 Gated Channel Transformation（GCT）[4]，以實現有效的特徵選擇和融合。我們的研究結果表明，提出的模型能夠從有限的資料集中生成豐富的影像特徵，我們也進一步證明將 dual Energy CT、臨床和病理特徵融合對於存活結果預測相當重要，因為不同的患者受到各種身體狀況的影響，dual Energy CT 無法代表患者全部的身體狀況。

關鍵字：肺癌、存活率、雙能量電腦斷層掃描、深度學習、卷積神經網路

Abstract

Lung cancer has a high mortality rate and poses significant challenges for post-surgery survival. The survival periods of patients with the same stage of lung cancer can vary significantly, which makes accurate treatment decisions difficult. While lung cancer staging is a primary indicator for determining treatment options like tumor resection or radiation therapy, relying solely on staging can result in inaccurate decisions. Therefore, incorporating survival prediction as a supportive indicator in preoperative diagnosis is crucial to improve the quality of treatment decisions. In this work, we proposed a novel deep learning model with attention mechanisms for 3-year survival prediction by integrating dual energy Computer Tomography (dual energy CT) and clinical characteristics. This ConvNeXt[1] based model leverages channel attention to filter features from tumor textures, along with a damper block[2] for tumor size learning. Through the experiments, we achieved promising results, with an accuracy of 86.03, sensitivity of 82.86, and specificity of 86.69 by combining ConvNeXt[1] with channel attention. The channel attention incorporates the Squeeze-and-Excitation (SE)[3] and Gated Channel Transformation (GCT)[4] for effective feature fusion and selection. Some experiments show that the model effectively generate informative features from CT with small dataset. We

further demonstrate the fusion between features from CT and clinical characteristics are essential for predicting individual survival outcomes, which are influenced by various bodily conditions. Overall, the proposed model allows for a rapid and accurate survival prediction.

Keywords: Lung cancer, Survival, Dual energy CT, Deep learning, Convolution neural network

Table of Contents

List of Figures

List of Tables

Chapter 1 Introduction

Lung cancer remains the leading cause of cancer-related deaths, accounting for approximately 1.79 million fatalities (18%) in 2020. The overall 5-year survival rate for individuals diagnosed with lung cancer ranges from 10% to 20%[5, 6]. Between 2009 and 2015, a significant portion of patients (57%) received a diagnosis of metastatic disease, which corresponds to a 5-year relative survival rate of only 5%. Conversely, patients diagnosed with localized-stage disease had a higher survival rate of 57%, but the unfortunate reality was that only 16% of patients fell into this category[7, 8]. Notably, between 2015 and 2019, there was a sudden annual increase of 4.5% in rates of localized-stage diagnoses. This resulted in a rise in the proportion of localized-stage cases to 26%, accompanied by a corresponding improvement in the 3-year relative survival rate, which reached 31%. These statistics emphasize the critical role of timely treatment in lung cancer survival rates. Delayed treatment can have a profound impact on overall survival outcomes[9, 10].

Optimal treatment planning for lung cancer patients presents challenges due to the heterogeneity observed among individuals, despite the effectiveness of early interventions in enhancing survival rates[11, 12]. Previous studies have indicated that patients with the same TNM staging can exhibit varying survival times[13, 14].

Additionally, Howlader et al.[15] has shown that small-cell lung cancer (SCLC) generally has a lower chance of survival. Another study has concluded that the presence of liver and bone metastases is associated with poorer survival outcomes[16]. Various attributes influence the survival period, and even patients with the same clinical and pathological status may experience different survival times. Therefore, considering survival as a new and distinct marker is essential in making informed decisions regarding further treatment for patients.

Several studies have attempted to predict survival outcome of lung cancer patients using various technique. Agrawal, A., et al. [17] has employed different machine learning methods and a voting technique to predict the survival of lung cancer patients at different time points, such as 6 months, 9 months, 1 year, 2 years and 5 years of diagnosis, based on the Surveillance, Epidemiology, and End Results (SEER) data[18]. Lai, Y.-H., et al. [19] has utilized a deep neural network (DNN) to predict the 5-year survival outcome of lung cancer patients. They incorporated 15 prognostic biomarkers and clinical data into their model. Additionally, He, B., et al. [20] have performed radiomic feature extraction on computed tomography (CT) images of lung cancer patients and developed a random forest classifier to predict the survival status. However, these approaches have their limitations. Some rely on predefined radiomic features, which may lack flexibility and variability. On the other

hand, some studies focus solely on clinical data, neglecting the valuable information provided by CT images. Indeed, CT images provide a wealth of tumor-related information, encompassing factors like tumor location and size. On the other hand, clinical data offers essential information about patients, such as their medical history and overall health. Thus, the integration of features extracted from CT images and clinical data is vital in bolstering the predictive capacity of survival outcome models. By combining these two sources of information, a more comprehensive and accurate understanding of the condition of a patient can be obtained, leading to improved prediction of survival outcomes.

Convolutional neural networks (CNNs) is a powerful type of deep learning model that has been specifically developed for the analysis and processing of visual data. They have gained widespread adoption across various domains, including computer vision and pattern recognition tasks[21-23]. In the field of medical imaging, CNNs have made remarkable progress, revolutionizing the interpretation and analysis of medical images. Recent advancements in deep learning have greatly contributed to the identification, classification, and quantification of patterns in medical images[24-27]. A key factor in these breakthroughs is the ability of CNNs to automatically learn hierarchical representations of features directly from the data, eliminating the need for handcrafted features based on domain-specific

knowledge[28]. Furthermore, 3D CNNs can process the entire volume of a medical scan, capturing spatial relationships and contextual information across multiple slices[25]. This enables more comprehensive and accurate analysis of complex anatomical structures, such as organs or tumors, in three-dimensional space. 3D CNNs have been successfully utilized in various tasks, including organ segmentation, lesion detection, disease classification, and treatment planning[29-31]. These networks can exploit the volumetric nature of medical images, leveraging their ability to learn spatial features and capture 3D patterns and structures.

In this study, we present a novel model designed to predict the 3-year survival outcomes of lung cancer patients after treatment. The approach involves leveraging CNNs to extract features from Dual Energy CT scans. These scans provide valuable lung tumor characteristics. Then we integrate the extracted features with clinical data encompassing clinical information and pathological features of the patients. By combining these two sources of information, the model gains a comprehensive understanding that enhances the predictive capability for survival outcomes.

The organization of this study is structured into several chapters. In Chapter 2, we delve into the details of the dataset, which includes Dual Energy CT scans and clinical data. Chapter 3 introduces the proposed method, carefully outlining each stage involved. We present the feature extraction process from Dual Energy CT scans

using CNNs and describe the integration of clinical data with the extracted features. Chapter 4 showcases the experimental results. We present the performance metrics and perform a thorough analysis of ablation study. Finally, chapter 5 draws conclusions based on the experimental results. We also discuss the implications of the study and suggest potential avenues for future research and improvements in the proposed method.

Chapter 2 Materials

This paper utilizes a dataset provided by the Department of Medical Imaging at National Taiwan University Hospital. The primary imaging technique employed is dual energy CT, an advanced medical imaging technology used to enhance the diagnosis of traditional CT scans. By utilizing two different X-ray energy levels and synthesizing images based on different keV values, dual energy CT provides more detailed and accurate images of tissues and organs.

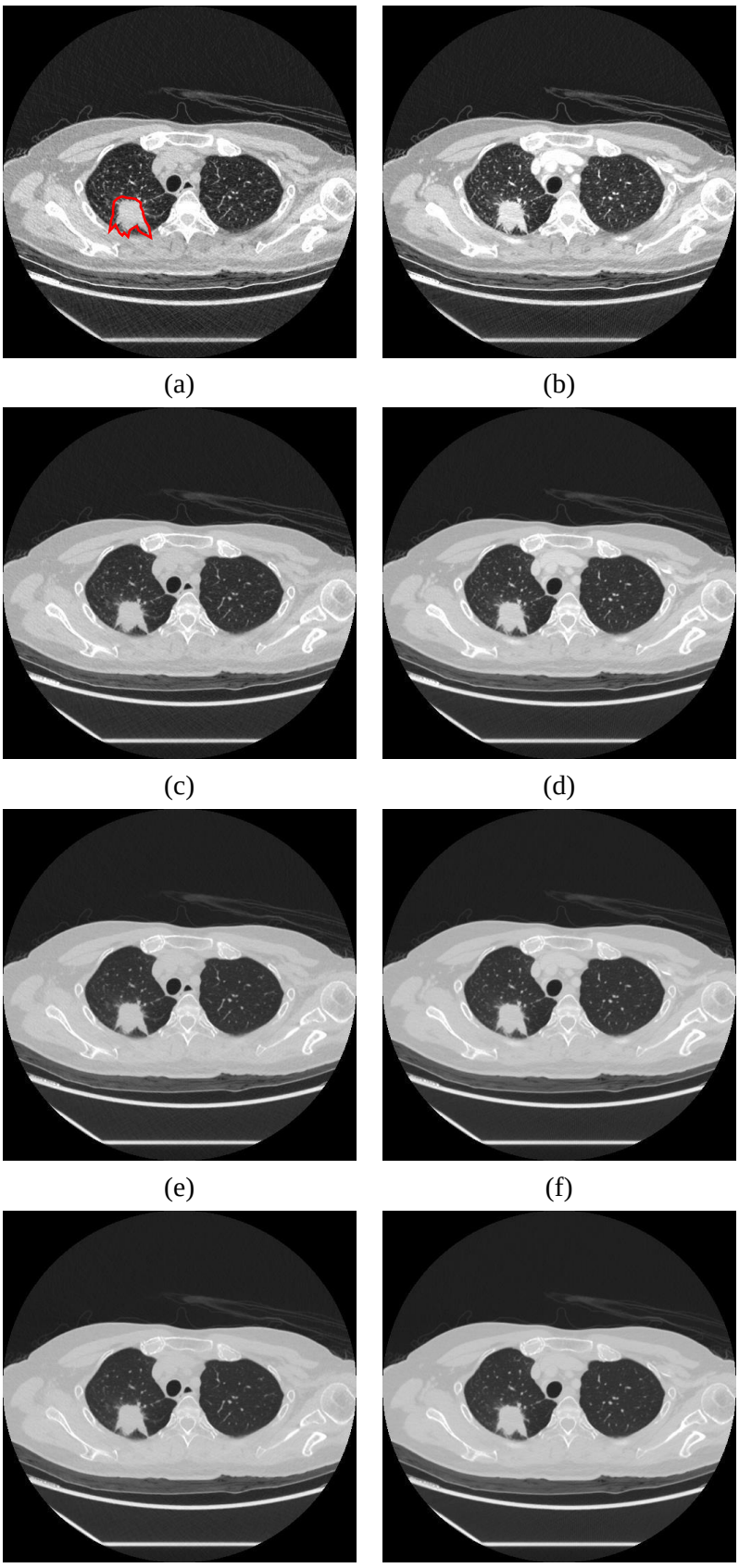
The dataset comprises records from 306 patients, collected between July 2018 and October 2022. Each patient is associated with 11 cases, utilizing dual energy CT imaging, spanning a range of 40-140 keV with a 10 keV interval. These cases include contrast-enhanced images (C+) and images without contrast enhancement (C-). Within each patient, 11 cases of different keV were obtained using the same scanning procedure. Fig. 2 - 1 illustrates different energy level images for the same slice across four distinct keV units. Notably, lower energy level images exhibit higher contrast, while higher energy level images appear smoother and blurrier in nature[32]. The CT images in this study underwent reconstruction to a size of 512×512 pixels and were stored in the digital imaging and communications in medicine (DICOM) format. The pixel intensities in the images were represented as 16-bit grayscale, using Hounsfield

Units (HU)[33]. HU provides a quantitative scale for describing the X-ray energy absorption in tissues, enabling the characterization of different structures and densities within the scanned anatomy, aiding in the interpretation and analysis of the CT images.

The collection of clinical information for this study encompassed a comprehensive set of 26 indicators that were meticulously recorded. These indicators encompassed various aspects related to the medical profile of the patient, including gender, smoking status, hypertension (HTN), diabetes mellitus (DM), family history of lung cancer, and FEV1/FVC ratio that represents the forced expiratory volume in one second (FEV1) divided by the forced vital capacity (FVC). Furthermore, an extensive range of lung cancer-specific clinical and pathological data was also obtained. The data encompassed crucial parameters such as tumor size, tumor location, differentiation, lymphovascular invasion (LVI), anaplastic lymphoma kinase (ALK) status, ROS oncogene 1 (ROS1) gene fusions, epidermal growth factor receptor (EGFR) status, presence of EGFR mutations, and the clinical staging and pathological staging system regarding the TNM staging of the disease. Additionally, the presence or absence of metastasis and recurrence were also documented.

The dataset used for this analysis comprised a total of 236 patient records, representing a three-year span of data. To ensure the validity of the data, only patients

who had undergone surgery at least three years prior to the study are included. For those patients who unfortunately did not survive beyond this three-year period are classified as deceased. Conversely, patients who survived beyond the three-year mark are categorized as alive. The clinical records are listed in Table 2 - 1.



(a) (b) (c) (d) (e) (f) (g) (h)

Fig. 2 - 1 Comparison of different energy level images with or without contrast enhancement, (a) C- 40 keV, (b) C+ 40 keV, (c) C- 70 keV, (d) C+70 keV, (e) C- 100 lev, (f) C+ 100 lev, (g) C- 140 lev, and (e) C+ 140 lev, the red circle is a tumor.

Table 2 - 1 Clinical and pathological characteristics in 3-year survival. For quantitative variables, data are presented as mean ± standard deviation. For qualitative variables, data are represented as the number of observations. The values in parentheses represent the ratio of the data belonging to the same category.

	3-year survival (n=236)	
	Deceased (n=34)	Survivor (n=202)
Male	21	92
Female	13	110
Age (year)	68±12	64±12
Size (cm)	5.3±3.2	2.9±2.2
Smoking	16	48
Predicted FEV1 (L)	2.0±0.7	2.1±0.7
Predicted FCV (L)	2.6±0.8	2.7±0.7
FEV1/FVC (%)	76.9±17.4	80.9±8.2
PORT-A	10 (29%)	9 (4%)
Resection	6 (18%)	142 (70%)
Tumor Location		
RUL	9 (26%)	53 (26%)
RML	6 (18%)	39 (19%)
RLL	4 (12%)	51 (25%)
LUL	9 (26%)	40 (20%)
LLL	3 (9%)	12 (6%)
Right pulmonary hilar lung	0 (0%)	1 (<1%)
Left side	0 (0%)	1 (<1%)
Resection Location		
RUL	0 (0%)	48 (24%)

RML	1 (3%)	8 (4%)
RLL	2 (6%)	29 (14%)
LUL	1 (3%)	36 (18%)
LLL	1 (3%)	21 (10%)
RFL/LFL	1 (3%)	0 (0%)
LL	0 (0%)	1 (<1%)
Differentiation		
Undifferentiated	0 (0%)	1 (<1%)
Well	0 (0%)	22 (11%)
Moderately	2 (6%)	76 (38%)
Poorly	4 (12%)	23 (11%)
Lymphovascular Invasion (LVI)		
Present	2 (6%)	19 (9%)
Not Present	2 (6%)	97 (48%)
Indeterminate	0 (0%)	3 (1%)
ALK		
ALK(-)	22 (65%)	74 (37%)
ALK(+)	0 (0%)	6 (3%)
ROS-1		
ROS-1(-)	15 (44%)	58 (29%)
ROS-1(+)	2 (6%)	19 (9%)
EGFR		
EFGR(-)	15 (44%)	14 (7%)
EFGR(+)	9 (26%)	36 (18%)
EGFR Mutation		
L858R	5 (15%)	26 (13%)
T790M	0 (0%)	3 (1%)
Exon 19	5 (15%)	12 (6%)
Exon 20	0 (0%)	1 (<1%)
Exon 21	0 (0%)	1 (<1%)
L861Q	0 (0%)	1 (<1%)
HER2 Exon 20 insert	0 (0%)	1 (<1%)
Clinical Staging		

T stage		
T1	2 (6%)	48 (24%)
T2	6 (18%)	32 (16%)
T3	4 (12%)	13 (6%)
T4	19 (56%)	21 (10%)
N Stage		
N0	3 (9%)	75 (37%)
N1	5 (15%)	5 (2%)
N2	12 (35%)	17 (8%)
N3	11 (32%)	16 (8%)
M Stage		
M0	11 (32%)	82 (41%)
M1	20 (59%)	31 (15%)
Pathological Staging		
T Stage		
T1	0 (0%)	72 (36%)
T2	2 (6%)	36 (18%)
T3	3 (9%)	5 (2%)
T4	0 (0%)	1 (<1%)
N Stage		
N0	3 (9%)	95 (47%)
N1	1 (3%)	8 (4%)
N2	1 (3%)	9 (4%)
N3	0 (0%)	2 (1%)
M Stage		
M0	4 (12%)	106 (52%)
M1	1 (3%)	8 (4%)
Metastasis		
No	4 (12%)	147 (73%)
Yes	30 (88%)	53 (26%)
Lymph Node	6	9
Brain	14	15
Bone	14	12

Liver	7	4
Adrenal Gland	4	3
Others	16	35
Recurrence		
No	26 (76%)	186 (92%)
Yes	8 (24%)	13 (6%)
Lung	3	6
Lymph Node	2	1
Brain	3	1
Bone	1	0
Liver	1	0
Others	2	3
PPD		
>3	0 (0%)	3 (1%)
2 – 3	2 (6%)	4 (2%)
1 - 2	9 (26%)	19 (9%)
0 - 1	13 (38%)	148 (73%)
HTN		
No	21 (62%)	159 (79%)
Yes	13 (38%)	42 (21%)
DM		
No	27 (79%)	180 (89%)
Yes	7 (21%)	21 (10%)
Family Lung Cancer History		
No	33 (97%)	145 (72%)
Yes	1 (3%)	54 (27%)
Complication		
No	10 (29%)	149 (74%)
Yes	24 (71%)	53 (26%)

Chapter 3 Methods

In this study, we proposed a model for predicting the 3-year survival outcomes of lung cancer patients after treatment. We utilize CNNs to extract features from dual energy CT scans and combine them with clinical data for the final prediction. Besides, we add some attention mechanisms for the model to gain condensed features relative to survival outcomes. In CNNs, the previous stages of the model look over dual energy CT scans and convert them into fine-grained features. These features are then passed to the subsequent stages of the model, which include attention mechanisms that examine and condense the features to extract the relevant ones for the 3-year survival prediction.

We employ various preprocessing techniques for both the dual energy CT scans and clinical data. For the dual energy CT scans, we use window settings for the dual Energy CT scans to enhance the visibility of the relevant organs in the images. We also crop the Volume of Interest (VOI) because tumors are much smaller compared to surrounding organs and tissue. During the training iterations, we randomly apply rotations and flips to mitigate model overfitting[34]. As for the clinical data, we transform the numerical values into binary features (0 and 1). This not only reduces the complexity of the data but also improves the training efficiency of the model. The

workflow is depicted in Fig. 3 - 1.

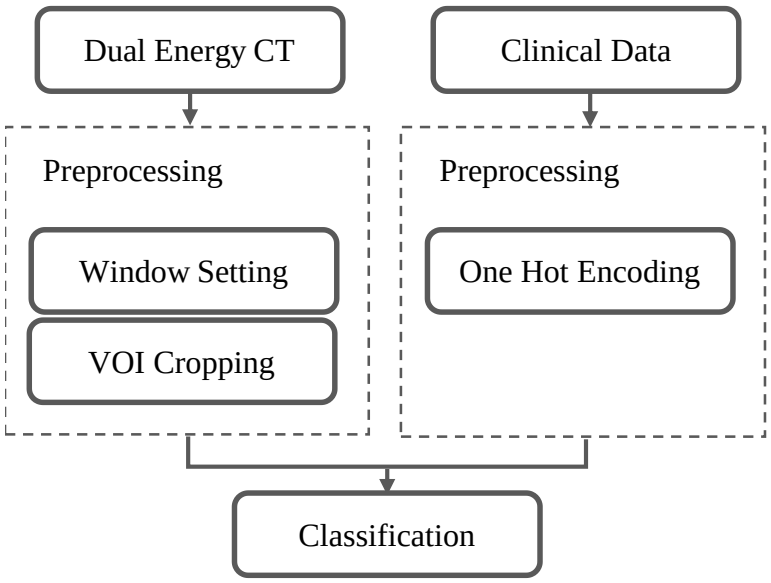


Fig. 3 - 1 The workflow of survival prediction method.

3.1. Preprocessing

3.1.1. Dual Energy CT

The original images are CT scans, which are three-dimensional, capturing cross-sectional slices of the body. Each scan comprises multiple slices, and each slice is represented by a matrix size of 512×512 pixels. However, tumors are typically much smaller than the matrix size of a single slice, resulting in unclear tumor representation in the image. This not only makes the identification of the tumor difficult for the model, but other organs and tissues in the image may also affect the training of model.

The images allow us to observe different organs, tissues, and substances,

including air with a Hounsfield Unit (HU) value of -1000, water with a HU value of 0, and bones with HU values ranging from 400 to 1000. Within this broad range, not all organs are of interest to us. We focus on the lungs and their structures, as well as the location and shape of tumors. Therefore, we need to adjust the HU range to enhance and highlight the specific areas we are concerned about

To tackle above problems, we set the window center to -300 and the window width to 1400. This allows us to focus on the relevant organ structures like lung. Afterward, we perform a Volume of Interest (VOI) cropping, where we center the images around the VOI and cut them into 128×224×224 dimensions from the original size[35].

During the training process, we employ data augmentation techniques[36, 37] to enhance the diversity of the image dataset. Each image undergoes random flips along the x, y, and z-axis, with a 50% probability, to introduce variations in spatial orientation. Furthermore, the image is rotated by 15 degrees in three distinct directions, with a 50%, further augmenting the dataset. By applying these transformations, we effectively expand the available data, providing the model with a broader range of examples to learn from. The workflow is depicted in Fig. 3 - 2.

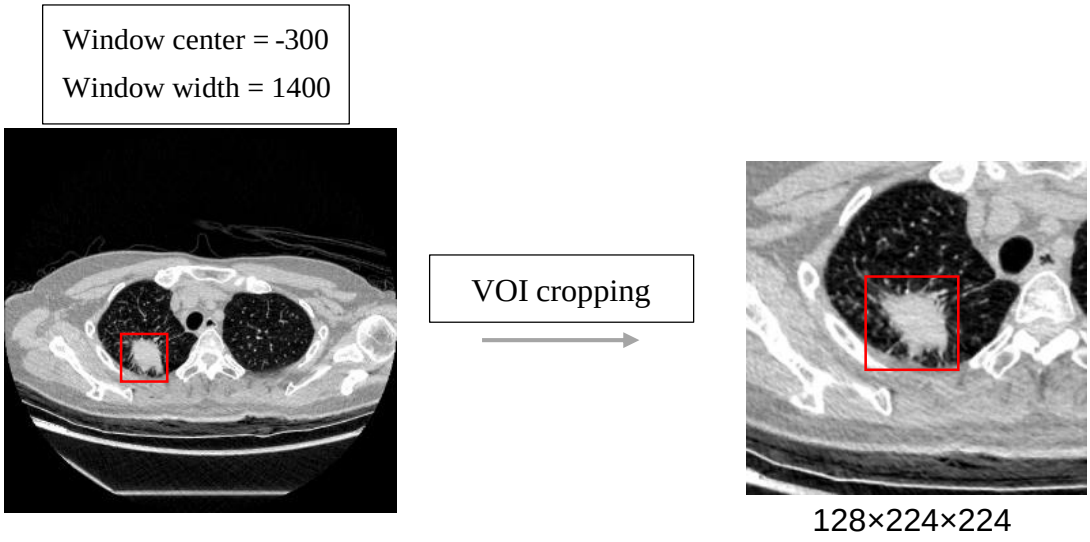


Fig. 3 - 2 The overview of image preprocessing.

3.1.2 Clinical Data

We utilized one hot encoding to convert the clinical data of patients into binary features, as illustrated in Fig. 3 - 3. This approach effectively addresses challenges associated with categorical variables in deep learning tasks. One hot encoding is a technique used to convert categorical variables, such as text or non-ordinal data, into numerical representations. Dummy variables are created for each unique category within the original variable that can be easily understood and processed by CNN models.

One hot encoding offers several benefits in data preprocessing for CNN models. First, enabling models to comprehend categorical data. CNNs, for example, operate on numerical data. Using one hot encoding, we can represent categorical variables in

a way that CNNs can interpret. This conversion allows the models to incorporate categorical information effectively into their learning process. Second, providing a precise representation for non-ordinal data. One hot encoding is particularly useful for non-ordinal data, with no inherent ordering among the categories. For example, when encoding gender (male/female), using binary columns allows the model to differentiate between the two categories accurately without assuming any specific ordinal relationship. Third, one hot encoding provides flexibility in customizing the transformation process based on domain knowledge or specific requirements. For instance, if we are specifically interested in whether a tumor size exceeds 3 cm, we can create a single column indicating whether the tumor size is greater than 3 cm instead of creating separate columns for each centimeter increment[38]. This approach reduces the number of columns and leverages domain knowledge to assist the model in diagnosis.

By employing one hot encoding, we eliminate any inherent order or hierarchy among the categories, treating them as independent and of equal importance. This enables the algorithms to capture the relationships between the categories without assuming any numerical correlation between the values. Through this transformation process, we obtained 150 clinical features, which can be readily utilized in the deep learning analyses.

Size (cm)	Size<1	Size<2	Size<3	Size>=3
1.8	0	1	0	0
2.4	0	0	1	0
11.2	0	0	0	1
2.5	0	0	1	0

Fig. 3 - 3 The example of one-hot encoding.

3.2 Classification

After preprocessing, the dual energy CT data are classified using a ConvNeXt[1] model as the backbone. To enhance the classification ability of the model, attention blocks are incorporated, enabling the model to identify the importance of each channel information[39]. Additionally, we include the size of the tumor as an additional input to the model. This is because tumor size is one of the decisive factors for lung cancer prognosis, and the fixed size of the dual energy CT data does not inherently include tumor size features. We utilize a damper block[2] to combine the features extracted from the dual energy CT after convolution with the tumor size feature.

Once the operations are performed on the damper block[2], the features from the 3D images are merged with preprocessed clinical data. This merged data is then fed into fully connected layers for training. The outcome of this process is the generation

of classification results. The workflow is depicted in Fig. 3 - 4.

=

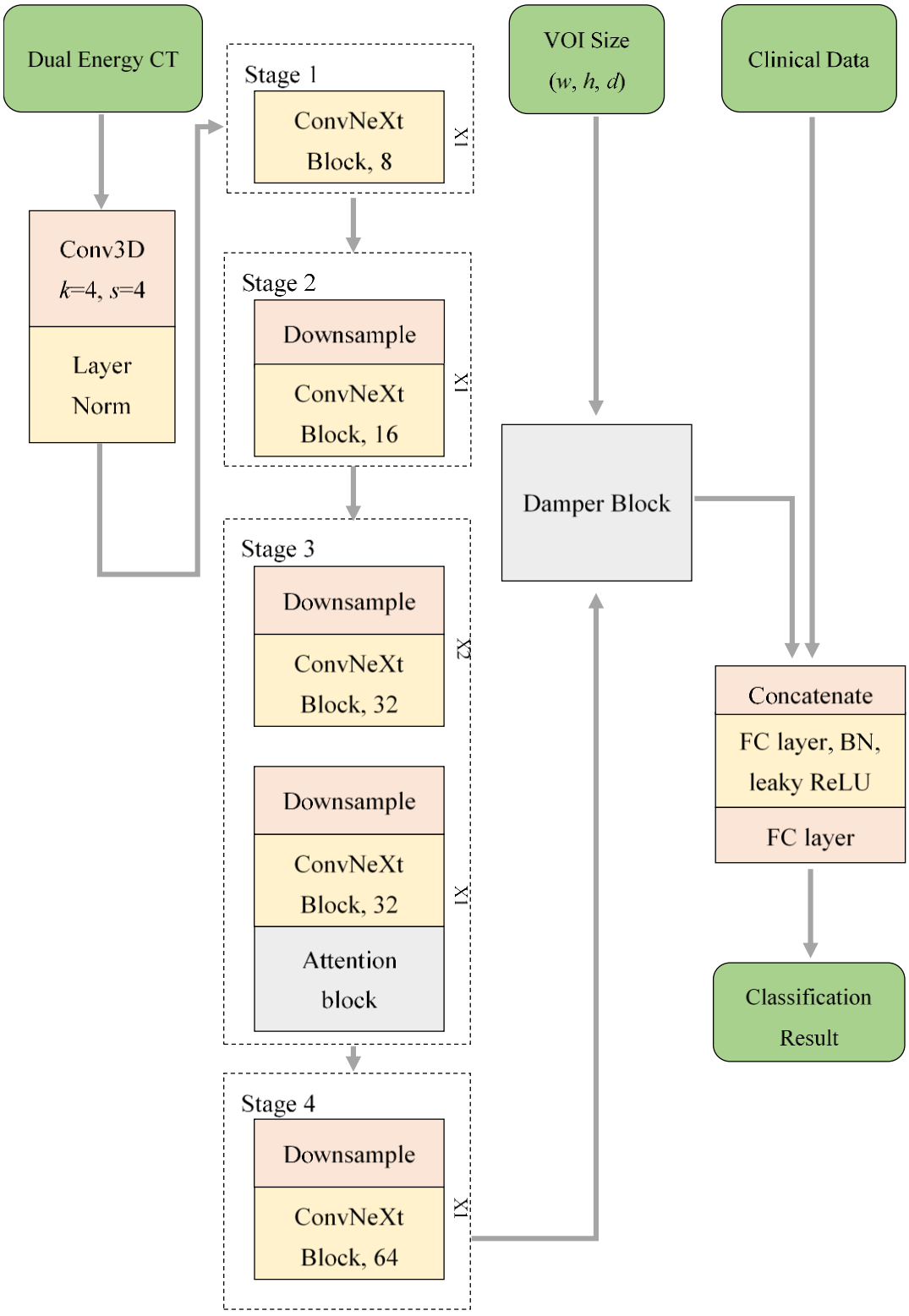


Fig. 3 - 4 The framework of ConvNeXt[1] based model. The numbers in ConvNeXt[1] Block and FC layer indicate the output dimensions.

3.2.1 ConvNeXt[1]

ConvNeXt[1] is widely recognized as one of the leading classification models among current CNNs. Although incorporating conventional structures such as convolution, depth-wise convolution, normalization, and activation functions[40], ConvNeXt[1] has achieved superior performance compared to Vision Transformers[41, 42] in terms of both accuracy and computation time.

The key to the success of ConvNeXt[1] lies in the ability to mimic certain aspects of Vision Transformers[41, 42] while still leveraging convolution as the primary feature extractor. This unique design allows ConvNeXt[1] to surpass Vision Transformers[41, 42], which do not rely on convolution at all. On the other hand, ConvNeXt[1] is derived from ResNet family[43-45] but gets better performance in various aspects. Unlike ResNet family[43-45], ConvNeXt[1] adopts (3, 3, 9, 3) blocks similar to Vision Transformers[41, 42] and utilizes depth-wise convolution, which shares similarities with the group convolution in ResNeXt[45] and the weighted sum operation in self-attention. Self-attention is a key component in Vision Transformers[41, 42] that derived from Natural Language Processing (NLP). Additionally, ConvNeXt[1] increases the kernel size to 7x7, further enhancing the performance. ConvNeXt[1] also reduces the number of normalization steps and activation functions[40] compared to ResNet family [43-45] and replaces batch

normalization (BN)[46] with layer normalization (LN)[47], a technique commonly employed in NLP. These adjustments contribute to the overall improvement of the performance of ConvNeXt[1].

In summary, by making specific modifications to the design, ConvNeXt[1] surpasses Vision Transformers[41, 42] and ResNeSt[43] by a significant margin. ConvNeXt[1] successfully combines the strengths of convolutional operations and incorporating insights from Vision Transformers[41, 42] to achieve outstanding classification performance. The architecture of ConvNeXt[1] is depicted in Fig. 3 - 5.

Due to the relatively small proportion of tumors in the dual energy CT images, using a large number of channels and a high number of blocks is unsuitable. The original ConvNeXt[1] architecture consists of (3, 3, 9, 3) blocks, with corresponding channel numbers of (96, 192, 384, 768). However, for the specific dataset, this configuration is not appropriate. Additionally, the number of available data samples is limited, recommending fewer channels and less blocks. We reduce the number of executions for the ConvNeXt[1] block from (3, 3, 9, 3) to (1, 1, 3, 1). This means the block is executed fewer times during forward propagation. Furthermore, the number of channels for each stage within the ConvNeXt[1] block has been changed from (96, 192, 384, 768) to (8, 16, 32, 64). This modification reduces the capacity of the model to capture complex features, but lower the chance for the model to be overfitting[34].

ConvNeXt[1] utilizes a 4x4 convolutional kernel with a stride of 4. The aim is to roughly organize the information in the image and compress the size of the image by a factor of 4. Each subsequent Downsample Layer uses a 2x2 convolutional kernel with a stride of 2 to capture finer details while compressing the image by a factor of 2. The implementation of the ConvNeXt[1] block mimics the Vision Transformers[41, 42], and the output size of the image remains consistent with the input size. That is, image compression is only performed within the Downsample Layer.

Within the ConvNeXt[1] block, the image undergoes a series of operations that are repeated one or three times. These operations include Depth-wise Convolution, Layer Normalization (LN)[47], Convolution, the Gaussian Error Linear Unit (GELU) [48], and Layer Scaling[49]. ConvNeXt[1] shares the same number of convolutional kernels as the Vision Transformers[41, 42], and LN[47] is directly inherited. Additionally, GELU[48] is chosen over Rectified Linear Units (ReLU) [50, 51] because GELU[48] offers advantages by providing smoothness, improved representation power, regularization, and better gradient propagation. GELU[48] enhances deep learning models by stabilizing gradients compared to the popular ReLU activation function[50, 51]. The architecture of ConvNeXt block and Downsample layer is depicted in Fig. 3 - 6.

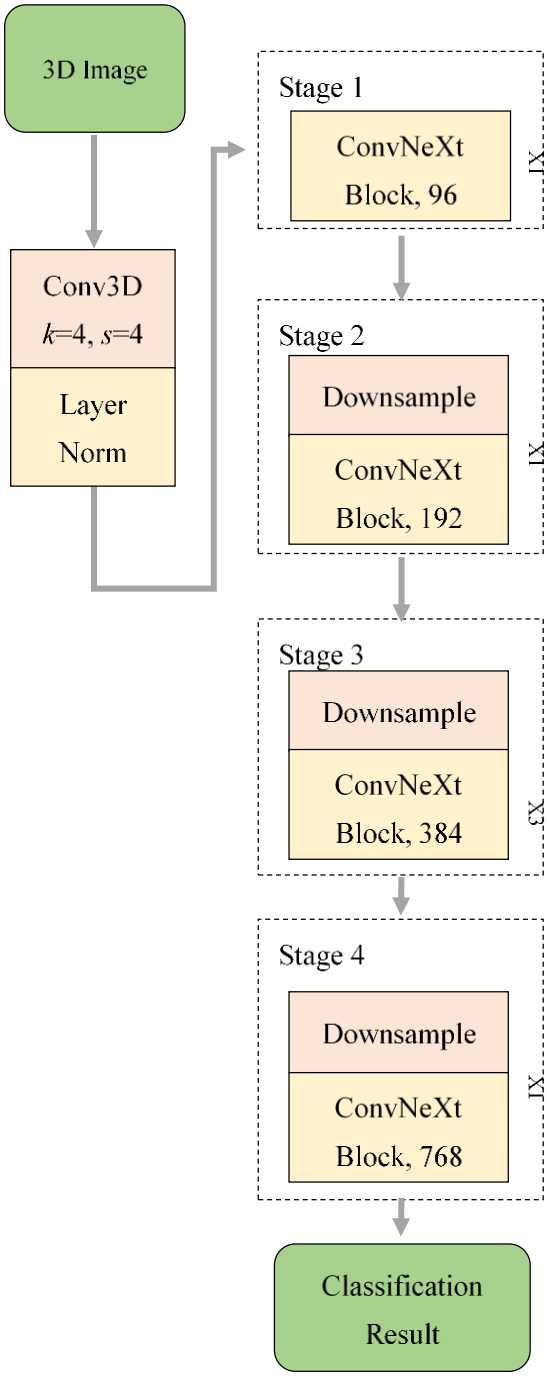


Fig. 3 - 5 The architecture of ConvNeXt[1]. The numbers in ConvNeXt[1] block indicate both the input and the output dimensions.

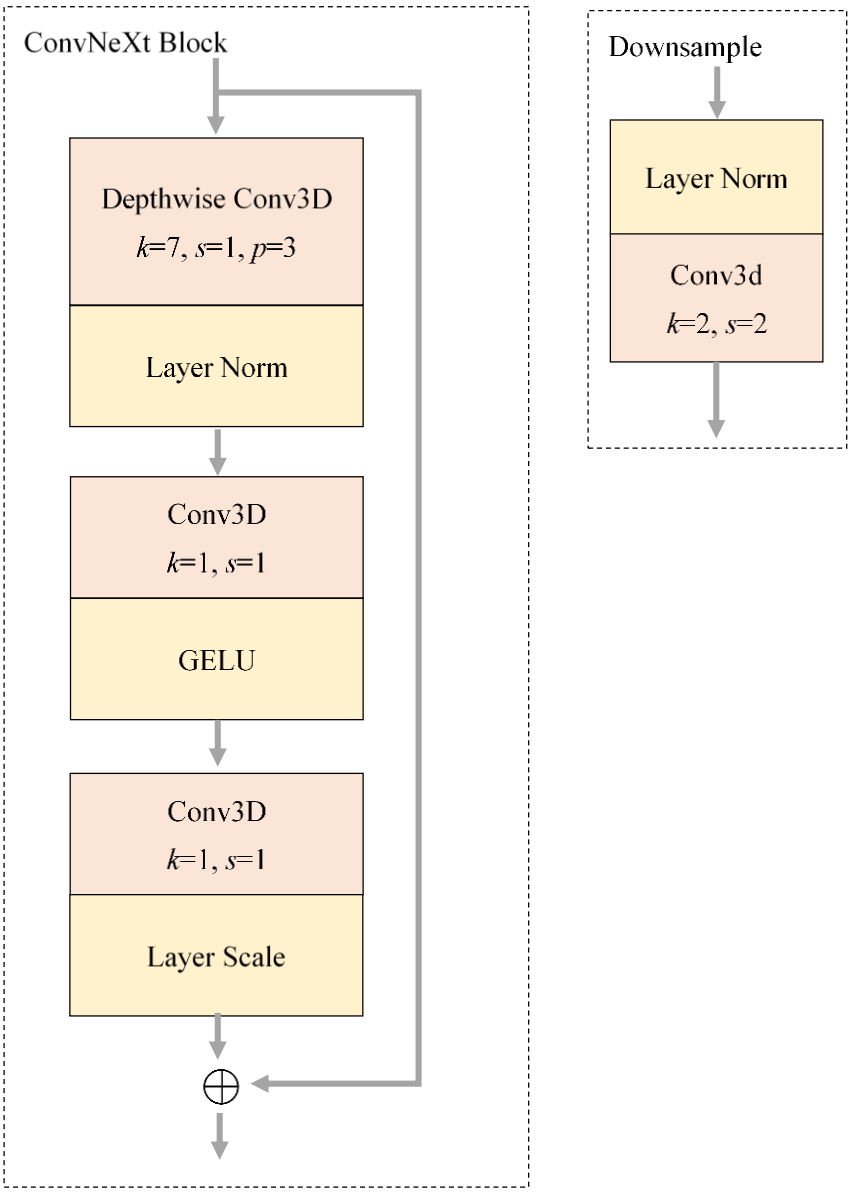


Fig. 3 - 6 The implementation of ConvNeXt block and Downsample layer.

3.2.2 Attention Block

To optimize the ability of the model to predict survival outcomes based on relevant features, we have incorporated an attention block into the architecture of the model. Specifically, we have employed a channel attention block to address the challenge of distinguishing features with varying levels of importance within the channels, enhancing the performance of the model in capturing relative features for

survival outcome prediction. The attention block is depicted in Fig. 3 - 7.

A channel attention block is a component widely employed in deep learning models for computer vision tasks, especially in image recognition. The primary goal of this block is to capture and emphasize the most important features within each channel of a feature map, thereby enhancing the capacity of the model to extract relevant information from the input data. By giving more weight to channels that contain crucial information and less weight to less informative channels, the channel attention block enables the model to focus on the most discriminative features, leading to improved performance and better generalization.

Moreover, a channel attention block is convenient and flexible in integration within CNNs. This block can be easily inserted at various stages of the model architecture, allowing for efficient feature recalibration and enhancing the performance of the model. Channel attention is convenient by exclusively focusing on the channel dimension of the feature map, enabling the independence from spatial dimensions and compatibility with different network architectures. Whether a convolutional layer, a residual block, or a fully connected layer, channel attention can be seamlessly integrated into any part of the model, providing an additional level of adaptability and boosting the ability of the model to capture essential features. This flexibility makes channel attention a versatile tool for improving the discriminative

power of models in a wide range of computer vision tasks without requiring significant architectural modifications.

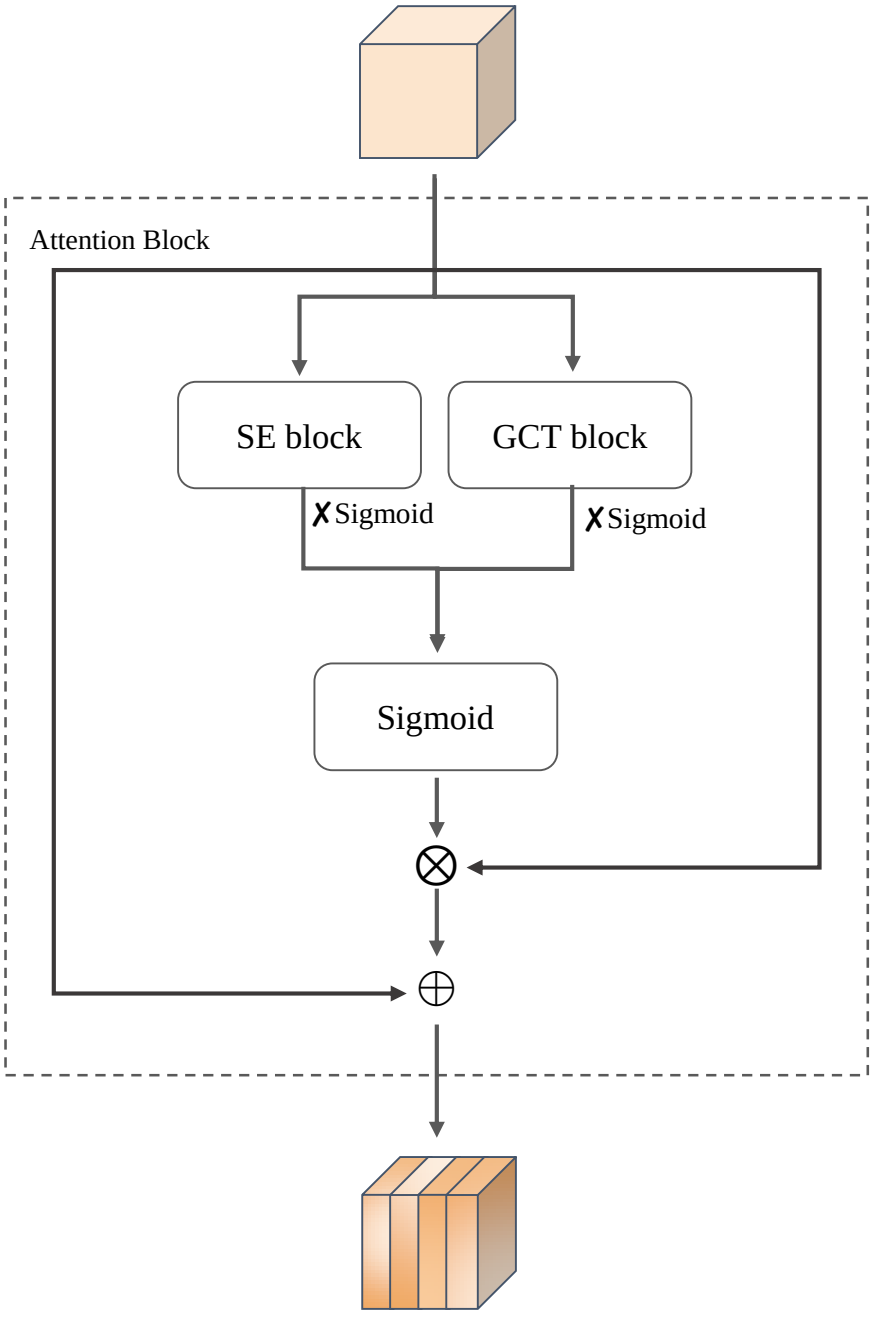


Fig. 3 - 7 The attention block.

3.2.2.1 Squeeze and Excitation (SE)[3]

The squeeze-and-excitation (SE) block[3] is a powerful building block in deep

neural networks, designed to enhance the representation and adaptively recalibrate channel-wise feature responses. The SE block[3] consists of two main steps: squeezing and exciting, with the details depicted in Fig. 3 - 8.

In the squeezing step, global information is captured by performing global average pooling over the spatial dimensions of the input feature maps. This reduces the spatial dimensions to a single-channel feature vector. We perform computations on the original tensor x, which has dimensions $C \times D \times H \times W$. By taking the average across the $D \times H \times W$ dimensions, we obtain the output $\mathrm{y} \in \mathbb{R}^C$, the c-th element of y is given by:

$$y_c = \mathrm{F}_{sq}(\mathrm{x}) = \frac{1}{D \times H \times W} \sum_{i=1}^{D} \sum_{j=1}^{H} \sum_{k=1}^{W} x_c(i,j,k) \ \#(1)$$

In the exciting step, the squeezed feature vector is fed into two fully connected (FC) layers, followed by activation functions such as Rectified Linear Units (ReLU)[50, 51] and sigmoid. These layers act as a gating mechanism, allowing the network to learn channel-wise feature dependencies. The output of the SE block[3] is a channel-wise scaling vector that weights the importance of each channel in the feature maps. The first FC layer, denoted by $W_1 \in \mathbb{R}^{C \times \frac{C}{2}}$, is followed by the ReLU[50, 51] function. ReLU[50, 51] introduces non-linearity into the functions, thus increasing the robustness of the model. In addition, ReLU[50, 51] ensures that the output values are non-negative and introduces non-linear transformations to the input.

On the other hand, the second FC layer, denoted by $W_2 \in \mathbb{R}^{\frac{C}{2}\times C}$, is followed by the sigmoid function. Sigmoid acts as a gating mechanism that restricts the range of values between 0 and 1. By applying sigmoid to the output of the second FC layer, we obtain an activation value, denoted as $\mathrm{z} \in \mathbb{R}^{C}$, which represents the final output of the model:

$$\mathrm{z} = \mathrm{F}_{ex}(\mathrm{y}) = \sigma\big(\mathrm{W}_2\delta(\mathrm{W}_1\mathrm{y})\big)\#(2)$$

Where δ represents the ReLU[50, 51] function, and σ represents the sigmoid function.

Finally, we multiply the activation value obtained from the excitation step with the original input x. This operation results in an output denoted as $\tilde{\mathrm{x}}$, which represents a reweighted version of the input x, adjusting the importance of each channel.

$$\tilde{\mathrm{x}} = \mathrm{F}_{scale}(\mathrm{x},\mathrm{z}) = z_c x_c\#(3)$$

Where $z_c \in \mathbb{R}^{1\times1\times1}$ is a scalar value and represents the c-th channel of the activation value z, and $x_c \in \mathbb{R}^{D\times H\times W}$ is a tensor and represents the spatial information of the input x in the c-th channel.

By adaptively recalibrating the feature responses, the SE block[3] allows the network to focus on informative channels and suppress less useful ones. This improves ability of the model to capture fine-grained details and enhances the representation power. The SE block[3] can be inserted into various positions in a

network, such as within residual blocks or after convolutional layers, enabling CNNs to capture and refine channel dependencies at different scales and depths. Overall, the lightweight nature and effectiveness make the SE block[3] an attractive component in designing neural network architectures across different computer vision tasks.

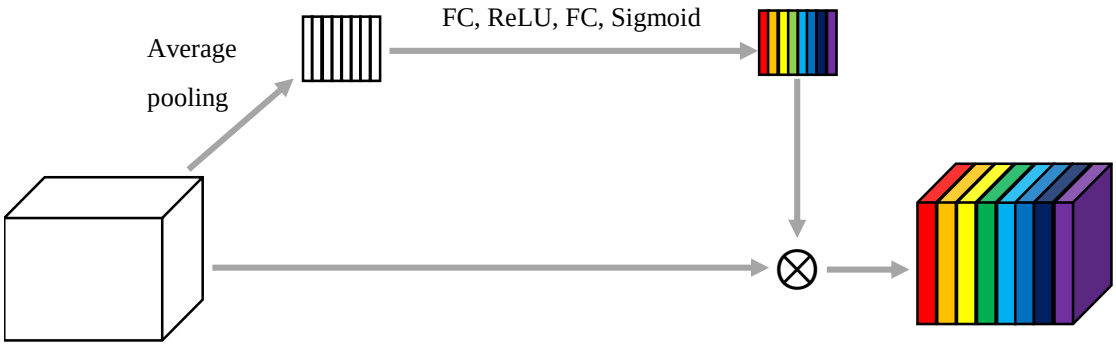


Fig. 3 - 8 A squeeze and excitation block[3].

3.2.2.2 Gated Channel Transformation (GCT)[4]

The Gated Channel Transformation (GCT) block[4] is a powerful building block in deep neural networks that enhance the channel-wise feature representation. The GCT block[4] resembles the squeeze-and-excitation (SE) block[3], but unlike the SE block[3], the GCT block[4] does not utilize fully connected layers. Therefore, the GCT block[4] is a lighter-more block and could be implemented with fewer parameters. The GCT block[4] consists of three main steps: embedding, normalization and gating, with the details depicted in Fig. 3 - 9.

In the embedding step, a global context embedding module is designed to

accumulate global context information in each channel. Instead of global average pooling, the GCT block[4] utilizes L_2-norm to obtain a single-channel feature vector. This choice is motivated by the higher level of robustness offered by L_2-norm compared to global average pooling. L_2-norm considers the magnitude of the feature vector elements, penalizing larger values more. This property helps mitigate sensitivity to spatial variations within the feature maps, enhancing robustness against small changes in the spatial locations of features. Additionally, a learnable parameter α is introduced to control the significance of each channel, allowing the model to adaptively adjust the importance during training. Formally, we perform computations on the original tensor x, which has dimensions $C \times D \times H \times \mathrm{W}$. By taking the L_2-norm across the $D \times H \times W$ dimensions, we obtain the output $\mathrm{y} \in \mathbb{R}^C$, the c-th element of y is given by:

$$y_c = \mathrm{F}_{embedding}(\mathrm{x}) = \alpha_c \|x_c\|_2 = \alpha_c \left\{ \left[\sum_{i=1}^{D} \sum_{j=1}^{H} \sum_{k=1}^{W} x_c(i,j,k)^2 \right] + \epsilon \right\}^{\frac{1}{2}} \#(4)$$

Where ϵ is a very small value used to avoid the derivation on the left-hand side equals zero.

In the normalization step, L_2-norm is applied again to operate over the channel-wise features due to the lightweight nature and robustness. Furthermore, the GCT block[4] introduces a scalar $\sqrt{C}$ to strengthen the scale of the normalization output, where C represents the number of channels. Adding this scalar factor

considers the discrepancy in the scale of the channel numbers and helps avoid very small values in the normalization output. By applying L_2-norm to the output of the embedding step and multiplying the result with $\sqrt{C}$, we obtain a channel normalization value, denoted as $\mathrm{z} \in \mathbb{R}^C$:

$$\mathrm{z} = \mathrm{F}_{normalization}(\mathrm{y}) = \frac{\sqrt{C}y_c}{\|\mathrm{y}\|_2} = \frac{\sqrt{C}y_c}{[(\sum_{c=1}^{C} {y_c}^2) + \epsilon]^{\frac{1}{2}}} \#(5)$$

Where ϵ is a very small value.

In the gating step, the output features from the normalization step are combined with the gating weights, $\gamma \in \mathbb{R}^C$ and $\beta \in \mathbb{R}^C$, which activate each channel with different proportions. This allows for fine-grained control over the contribution of each channel to the final feature representation. Subsequently, the sigmoid function is used to adjust the scale of the features, mapping them to the range of -1 to 1. The GCT block[4] eventually obtains a channel-wise scaling vector, denoted as $\tilde{\mathrm{x}}$, that represents the importance of each channel in the feature maps:

$$\tilde{\mathrm{x}} = \mathrm{F}_{gating}(\mathrm{x},\mathrm{z}) = x_c[1 + \sigma(\gamma_c z_c + \beta_c)] \#(6)$$

Where σ represents the sigmoid function, $z_c \in \mathbb{R}^{1\times1\times1}$ is a scalar value and represents the c-th channel of the channel normalization value z, and $x_c \in \mathbb{R}^{D\times H\times W}$ is a tensor and represents the spatial information of the input x in the c-th channel.

In contrast to the original setup in the GCT block[4], we modified by replacing

the tanh activation function with sigmoid. The experiments, detailed in Chapter 4, have demonstrated that this modification leads to further performance enhancements in the proposed model.

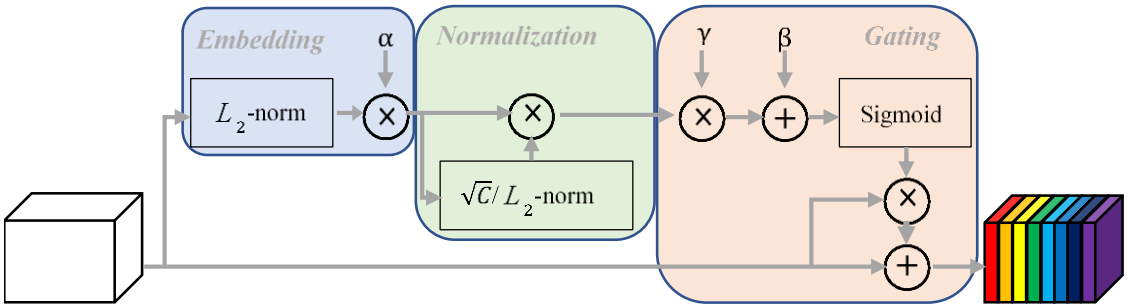


Fig. 3 - 9 A gated channel transformation block[4].

3.2.2.3 SE[3] + GCT[4]

We combine Squeeze and excitation (SE)[3] and Gated Channel Transformation (GCT)[4] as a channel attention. Both serve to find relative information within channels, but differ in their approach and the way to capture channel-wise dependencies. SE[3] gets weights via global average pooling, resulting in a channel descriptor vector; the channel descriptor vector is passed through a small neural network consisting of fully connected layers. On the other hand, GCT[4] introduces gating units to learn channel-wise transformations that adaptively adjust the feature map channels, aiming to exploit local dependencies within the feature map. The main difference between SE[3] and GCT[4] lies in their approach to capturing channel

dependencies. SE[3] focuses on global information and learns channel-wise attention weights, while GCT[4] exploits normalizations and learns channel-wise transformations through gating units. Both mechanisms aim to enhance the discriminative power of the model by emphasizing important channels while suppressing less relevant ones. Combining SE[3] and GCT[4] as a comprehensive channel attention mechanism, we harness the advantages of both approaches, resulting in a more powerful system.

In the study, we aim to obtain richer features by combining the features obtained from the SE block[3] and GCT block[4]. The input map from the ConvNeXt block[1] is separately processed through the SE block[3] and GCT block[4]. However, neither of them enters the Sigmoid function directly. Instead, the output features from both blocks are channel-wise added before entering the Sigmoid function, which transforms the values of each channel to the range of 0 to 1. At this stage, the obtained channel weights can be multiplied channel-wise with the input map. We remove the addition of GCT block[4] with the input map and perform the addition only after the channel-wise multiplication is completed. A detailed flowchart is provided in Fig. 3 - 7.

3.2.3 Damper Block[2]

In the proposed CNNs, a challenge arises due to the fixed size requirement of

input images in CNNs. As a result, the output features of the model do not capture information about the size of tumor. To address this limitation, we bring in a damper block[2] that incorporates the original VOI size as an effect parameter into the model. The damper block[2] integrates the output of CNNs with the width, height, and depth parameters of the original VOI. The purpose of the damper block[2] is to normalize the output of the previous CNNs by considering the size of the original VOI, as illustrated in Fig. 3 - 10. We employ three FC layers for the original VOI size. These FC layers are responsible for adjusting the dimensions of the width, height, depth to match the desired size for compatibility with the CNNs.

In the damper operation, we perform the following calculations:

$$I - \frac{Sum(I.*V)}{n(I.*V)} \#(7)$$

Where I represent 3D image features, V represent VOI features obtained from the VOI size after 3 fully connected layers. The expression $Sum(I.*V)$ represents the sum of I and V after element-wise multiplication, while $n(I.*V)$ represents the total number of pixels after element-wise multiplication. By incorporating the damper block[2], we enable the model to capture and leverage information about the size of the original VOI during the learning process. This allows for a more comprehensive and context-aware representation, enhancing the ability of the model to handle variations in input image sizes.

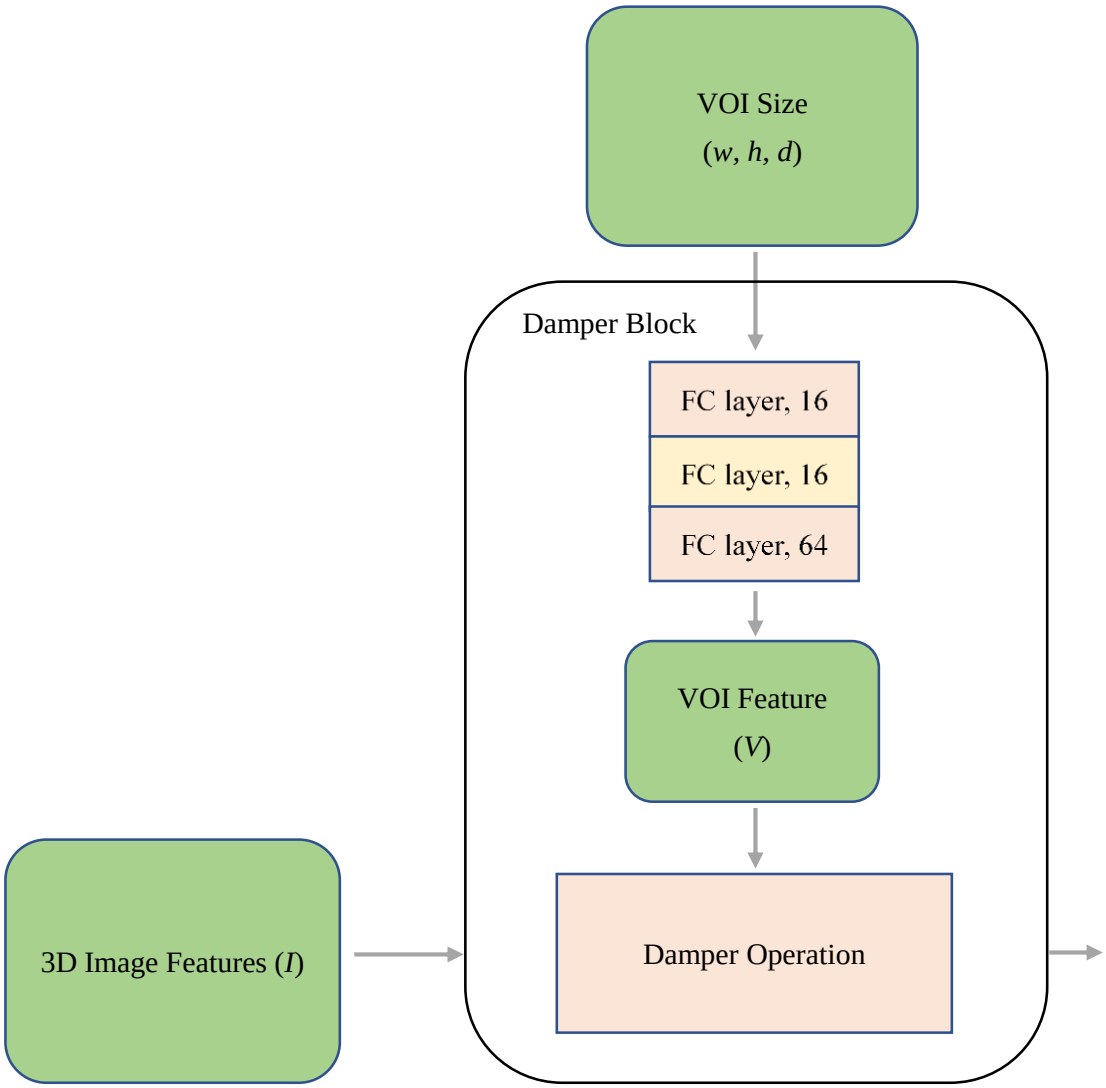


Fig. 3 - 10 The damper block[2].

3.2.4 Focal Loss[52]

Focal Loss[52] and Cross Entropy Loss are both loss functions commonly used in deep learning for classification tasks. However, we select Focal Loss[52] due to the distinct advantages over Cross Entropy Loss, particularly when dealing with imbalanced datasets.

Focal Loss, first introduced by Lin et al., aims to address the limitations of Cross Entropy Loss in imbalanced scenarios[52]. This loss function introduces a modulating factor that downweighs the loss for well-classified examples. By focusing more on hard and misclassified examples, Focal Loss[52] effectively helps the model pay greater attention to challenging samples, thereby improving the ability to learn from them. This modulation reduces the impact of easy examples, which are typically the majority class samples, and enhances the learning process for minority classes.

We define p_t as below, when the predicted target y is 1, the output of the model prediction p is given, otherwise, the output is $1-p$:

$$p_t = f(p, y) = \begin{cases} p, & y = 1 \\ 1 - p, & y = 0 \end{cases} \#(8)$$

Focal Loss[52] introduces an additional parameter γ as a modulating factor. γ allows further reduction of the loss for easily predictable data ($p_t \approx 1$). In such cases, even if there are many easily predictable data points, the accumulated loss will not be too large. This allows the model to focus on predicting challenging data points ($p_t \approx 0.5$) and primarily update the weights based on difficult data, reducing the impact of easy data. Besides, α assigns different weights to each class, reducing the loss for easily predictable data and amplifying the loss for difficult data. This further addresses the issue of data imbalance:

$$\text{Loss} = \text{FL}(p_t) = -\alpha(1 - p_t)^{\gamma}\log(p_t) \#(9)$$

By dynamically adjusting the loss contribution, Focal Loss[52] places greater emphasis on samples that are harder to classify, including those from the minority classes. As a result, Focal Loss[52] helps the model achieve better discrimination and generalization, particularly for underrepresented classes. This is crucial in various real-world applications, such as medical diagnosis, where the minority class samples are often of greater importance.

Chapter 4 Experiments

In the experiment, we use hyperparameters included a batch size of 16, 100 epochs, and the implementation of early stopping to expedite the training process. Furthermore, we implement AdamW optimizer[53], along with Cosine Annealing[54] and warmup strategy[55] to adjust the learning rate, ranging from 0.000001 to 0.001, with a cycle of 20 epochs.

Furthermore, we implement 5-fold cross validation to strengthen the experimental results[56-58]. Cross validation[56-58] is a widely used technique in deep learning to evaluate the performance and generalization ability of a predictive model. In this experiment, the available dataset is divided into five equal-sized subsets. The model is then trained and evaluated five times, with one-fold serving as the validation set and one-fold as the test set, while the remaining folds are used for training. This process is repeated iteratively until each fold has been used as the test set. By averaging the results from these iterations, we obtain a more robust estimation of the performance of the model.

4.1 Dual Energy CT with 40, 70, 100, 140 keV

In this experiment, we trained the proposed model on datasets of different keV because low-energy images of tumors have higher contrast, while high-energy images

have smoother edges. Additionally, in images with contrast agents, certain regions of tumor appear brighter, further enhancing tumor visibility. In the experiment, we selected 40, 70, 100, and 140 keV as the experimental settings and compared the results with and without contrast agent enhancement, as shown in Table 4 - 1. The ROC curve is depicted in Fig. 4 - 1. The model performs best on the dataset of 140 keV with contrast agent enhancement, achieving an accuracy of 86.03%, a sensitivity of 82.86%, and an AUC of 0.8908, whereas the dataset of 70 keV with contrast agent enhancement obtains the highest specificity (87.64%).

Furthermore, when comparing datasets with and without contrast agent enhancement, we can observe that the dataset with contrast agent enhancement achieves higher accuracy and specificity overall, while the dataset without contrast agent enhancement has higher sensitivity.

Table 4 - 1 The performances on different cases. The "-" symbol indicates the dataset without contrast agent enhancement, while the "+" symbol indicates the dataset with contrast agent enhancement.

keV	ACC (%)	SEN (%)	SPEC (%)	PPV (%)	NPV (%)	AUC
C-40	85.60±6.42	**82.86±11.95**	86.19±7.85	54.38±18.51	96.70±2.32	0.8871±0.0523
C-70	85.18±6.17	76.19±8.91	86.64±7.17	52.72±18.70	95.63±1.48	0.8867±0.0571
C-100	85.18±6.17	**82.86±11.95**	85.69±7.63	53.29±18.25	96.69±2.32	0.8887±0.0495
C-140	84.75±6.25	73.33±7.22	86.64±7.17	51.81±18.97	95.09±1.27	0.8853±0.0571
C+40	85.59±6.09	73.33±12.42	87.62±8.11	56.94±25.07	95.24±1.89	0.8900±0.0467
C+70	85.59±6.09	73.81±11.42	**87.64±8.24**	57.34±24.93	95.23±1.92	0.8742±0.0654

C+100	85.60±6.42	80.00±16.29	86.69±8.78	**57.71±25.29**	96.28±2.82	0.8858±0.0486
C+140	**86.03±6.45**	**82.86±11.95**	86.69±7.84	55.29±18.35	**96.72±2.33**	**0.8908±0.0470**

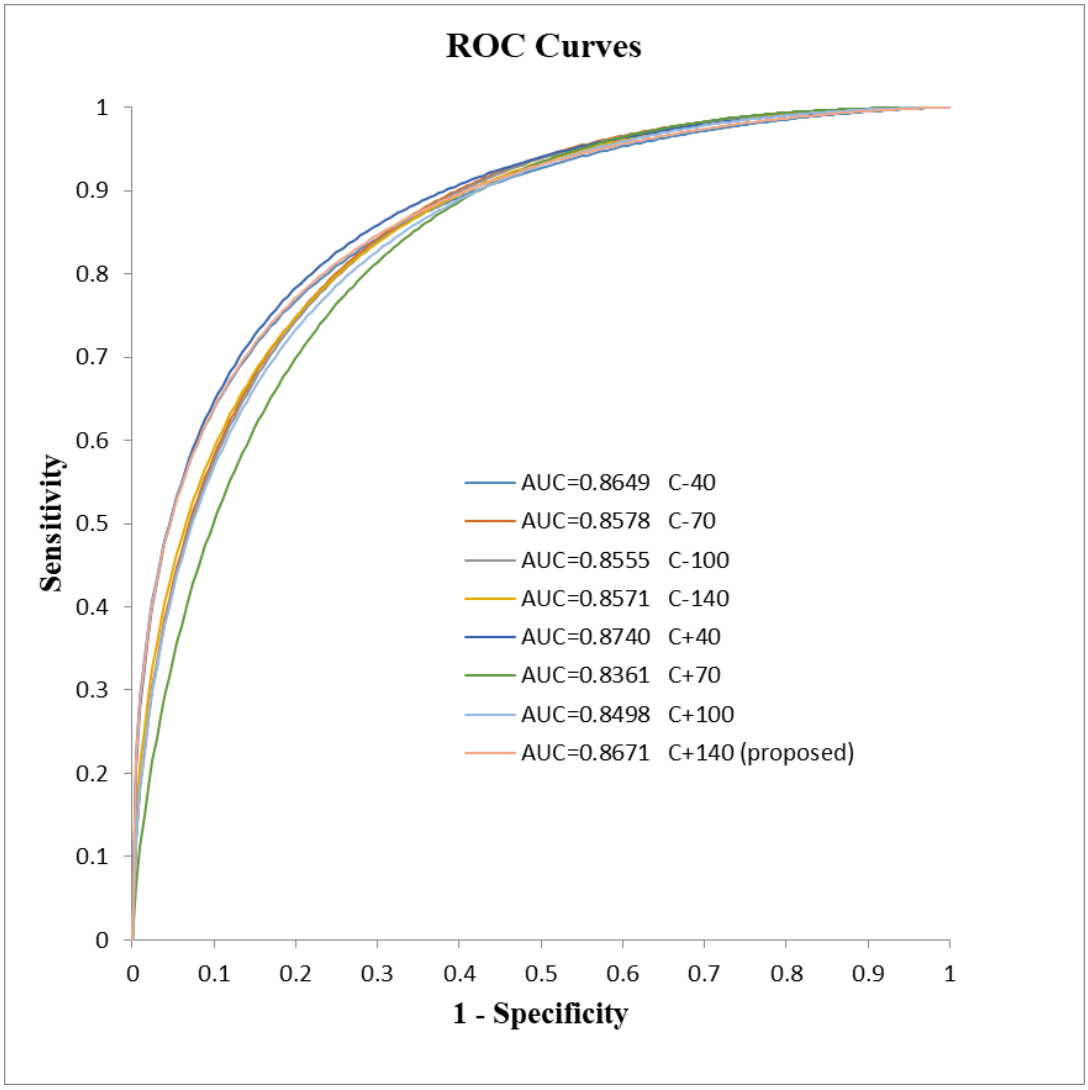


Fig. 4 - 1 The ROC curve for different keV with or without the contrast agent enhancement. The "-" symbol indicates the dataset without contrast agent enhancement, while the "+" symbol indicates the dataset with contrast agent enhancement.

4.2 Different Attention Blocks

In this experiment, we compared the performances obtained using different attention blocks. In addition to the SE block[3] and GCT block[4] already used in the

model, we also compared the performance of the Bottleneck Attention Module (BAM) block[59] and Convolutional Block Attention Module (CBAM) Block[60], which simultaneously incorporate channel attention and spatial attention.

Both the BAM block[59] and the CBAM block[60] introduce bottleneck structures that combine channel attention and spatial attention mechanisms, enabling the network to selectively emphasize informative features while suppressing less relevant ones. However, there is a difference in how the BAM block[59] and the CBAM block[60] handle these attention mechanisms. In the BAM block[59], the channel-wise and spatial-wise features are obtained simultaneously and then merged. On the other hand, the CBAM block[60] first obtains the channel-wise features and then uses them as input to the spatial attention module to obtain the final features. We noticed that the channel attention in both the BAM block[59] and the CBAM block[60] is similar to the SE block[3]. Therefore, we directly replaced their channel attention mechanism with the SE block[3].

We compared the performance of four attention blocks individually, and also examined the performance of different combinations. Since the BAM block[59] and the CBAM block[60] employ channel attention architectures similar to the SE block[3], we mainly conducted additional experiments combining them with the GCT block[4], such as SE[3] + GCT[4] (the proposed method in this paper), parallelizing

the Channel part of BAM[59] with GCT[4] (named BAM[59] + GCT[4]), parallelizing the Channel part of CBAM[60] with GCT[4] (named CBAM[60] + GCT[4]).

From Table 4 - 2 and Fig. 4 - 2, we observe that the SE[3] + GCT[4] combination achieves the best results with an accuracy of 86.03% and sensitivity of 82.86%, and BAM[59] obtains the highest sensitivity (88.19%). Furthermore, GCT[4] exhibits a sensitivity that was closest to SE[3] + GCT[4] among all the combinations. This indicates that GCT[4] is helpful in predicting survival outcomes for mortality. However, when combined with BAM[59] or CBAM[60], the performance of GCT[4] decreases. Therefore, in the end, we decide not to use BAM[59] and CBAM[60] in the proposed model.

Table 4 - 2 The performances using different attention blocks.

	ACC (%)	SEN (%)	SPEC (%)	PPV (%)	NPV (%)	AUC
SE	85.59	73.33	87.62	53.81	95.16	0.8889
	±6.09	±7.22	±7.29	±18.06	±1.19	±0.0499
GCT	84.32	82.38	84.67	51.98	96.66	0.8788
	±6.47	±6.21	±8.51	±18.63	±0.88	±0.0583
BAM	85.62	70.48	**88.19**	**59.72**	94.79	0.9002
	±7.48	±17.63	**±9.39**	**±28.09**	±2.68	±0.0381
CBAM	84.75	76.19	86.14	51.99	95.78	0.8712
	±5.48	±13.47	±8.01	±17.96	±2.37	±0.0550
SE + GCT	**86.03**	**82.86**	86.69	55.29	**96.72**	0.8908
	±6.45	**±11.95**	±7.84	±18.35	**±2.33**	±0.0470
BAM + GCT	84.77	76.19	86.19	54.72	95.60	**0.9037**
	±7.50	±8.91	±8.61	±26.88	±1.52	**±0.0410**
CBAM + GCT	84.76	70.48	87.17	51.21	94.61	0.8740
	±5.01	±2.13	±5.82	±18.38	±0.29	±0.0574

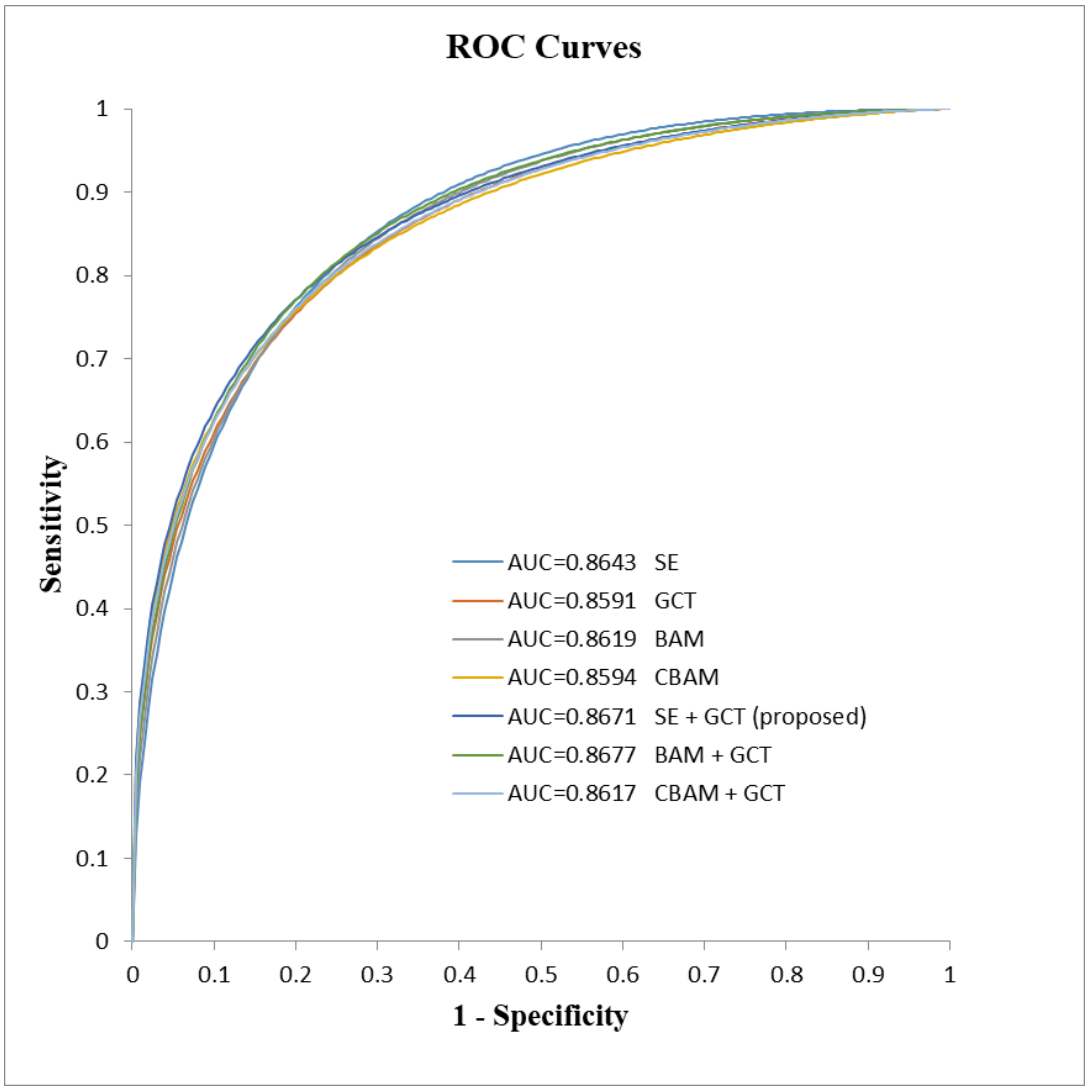


Fig. 4 - 2 The ROC curve for different attention blocks.

4.3 Positions of the Attention Block

In the experiment, we placed the attention block at different stages of the ConvNeXt[1] blocks. The proposed model consists of 4 stages, with each stage containing (1, 1, 3, 1) blocks, respectively. Since the attention block is used to update the weights of each channel, we primarily placed the attention block in the third and fourth stages because these stages capture more detailed features compared to the first

two stages.

When the attention block is placed after the third ConvNeXt[1] block in the third stage, the best performance is achieved with an accuracy of 86.03% and sensitivity of 82.86%, as shown in Table 4 - 3 and Fig. 4 - 3. Adding the attention block to each stage results in the best specificity, but the accuracy and sensitivity decreased.

Table 4 - 3 The performances for different positions of the attention block. "✓" represents the presence of the attention block after the corresponding ConvNeXt[1] block.

ConvNeXt Block				ACC	SEN	SPEC	PPV	NPV	AUC
Stage 1	Stage 2	Stage 3	Stage 4	(%)	(%)	(%)	(%)	(%)	
				84.33	79.52	85.17	**55.31**	96.24	**0.8926**
				±6.47	±12.55	±9.47	**±25.77**	±1.81	**±0.0405**
				86.03	**82.86**	86.69	55.29	**96.72**	0.8908
		✓		**±6.45**	**±11.95**	±7.84	±18.35	**±2.33**	±0.0470
				85.18	77.14	86.69	56.83	95.77	0.8922
		✓	✓	±6.17	±16.29	±8.78	±25.26	±2.82	±0.0458
				85.59	76.19	87.12	53.61	95.66	0.8845
		✓	✓ ✓	±6.09	±8.91	±7.17	±18.04	±14.59	±0.0438
				85.59	76.67	87.14	54.23	95.71	0.8836
			✓	±5.71	±12.42	±7.33	±16.85	±2.14	±0.0458
				84.75	77.14	86.19	52.83	95.74	0.8858
		✓	✓	±5.48	±16.29	±7.85	±16.93	±2.86	±0.0507
				85.60	**82.86**	86.19	54.38	96.70	0.8952
		✓	✓ ✓	±6.42	**±11.95**	±7.85	±18.51	±2.32	±0.0385
				85.60	73.33	87.64	57.14	95.23	0.8802
		✓	✓ ✓ ✓	±6.42	±12.42	±8.24	±25.73	±1.92	±0.0403
				84.75	64.76	**88.14**	55.14	93.71	0.8830
✓	✓	✓	✓ ✓ ✓	±6.25	±7.22	**±8.12**	±26.14	±1.14	±0.0496

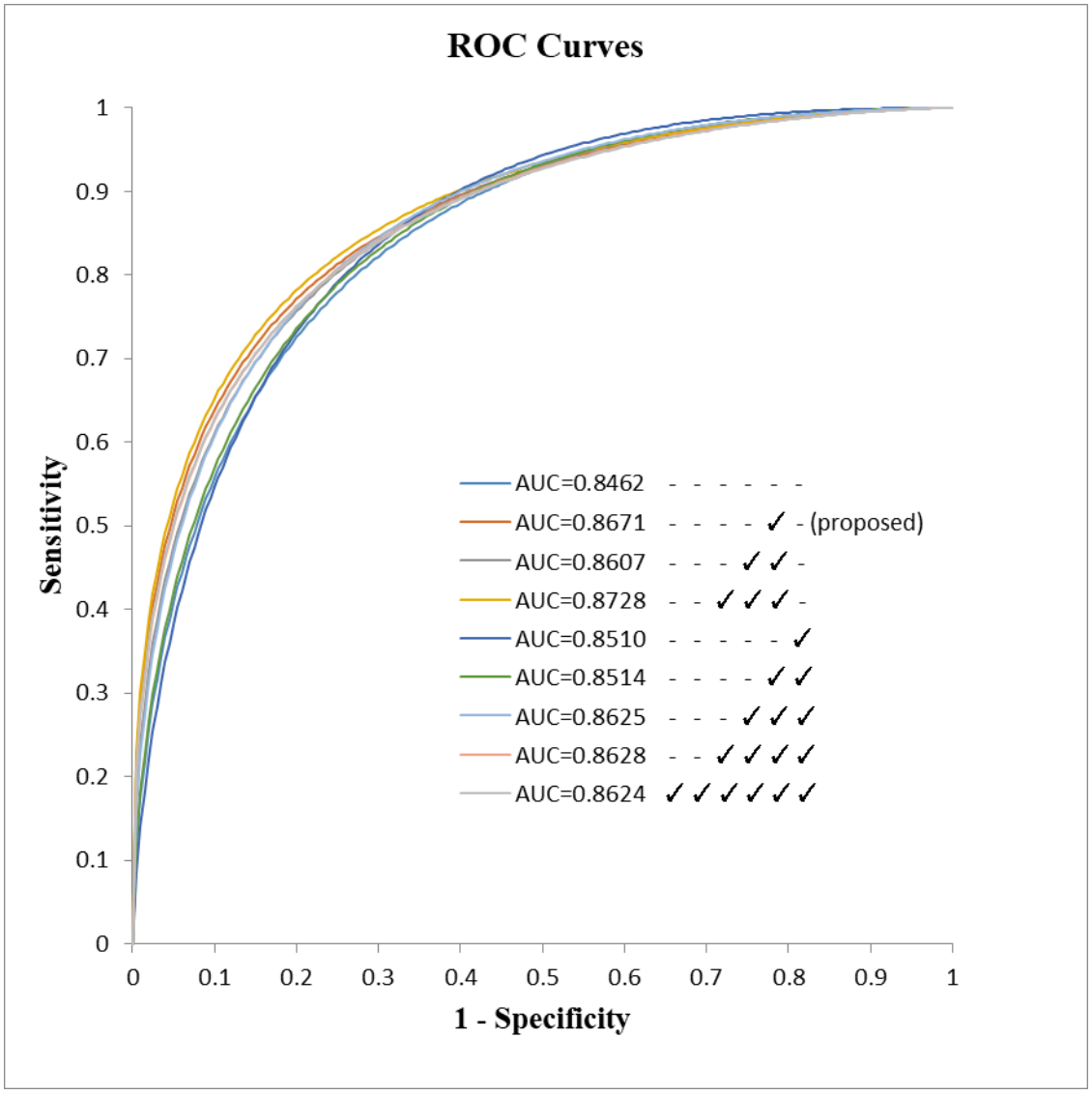


Fig. 4 - 3 The ROC curve for different position of the attention block.

4.4 With and Without the Damper Block

In this experiment, we compared the performance with and without damper block[2]. We find that incorporating the damper block[2] yielded better predictive results, as shown in Table 4 - 4 and Fig. 4 - 4. The accuracy is 86.03%, sensitivity is 82.86% and specificity is 86.69%. These results indicate a significant improvement compared to not using the damper block[2].

Table 4 - 4 The performances for the proposed model with or without damper block[2].

Damper Block	ACC (%)	SEN (%)	SPEC (%)	PPV (%)	NPV (%)	AUC
Without	85.15	76.67	86.60	50.75	95.60	**0.9163**
	±5.26	±7.22	±5.83	±9.85	±1.59	**±0.0404**
With	**86.03**	**82.86**	**86.69**	**55.29**	**96.72**	0.8908
	±6.45	**±11.95**	**±7.84**	**±18.35**	**±2.33**	±0.0470

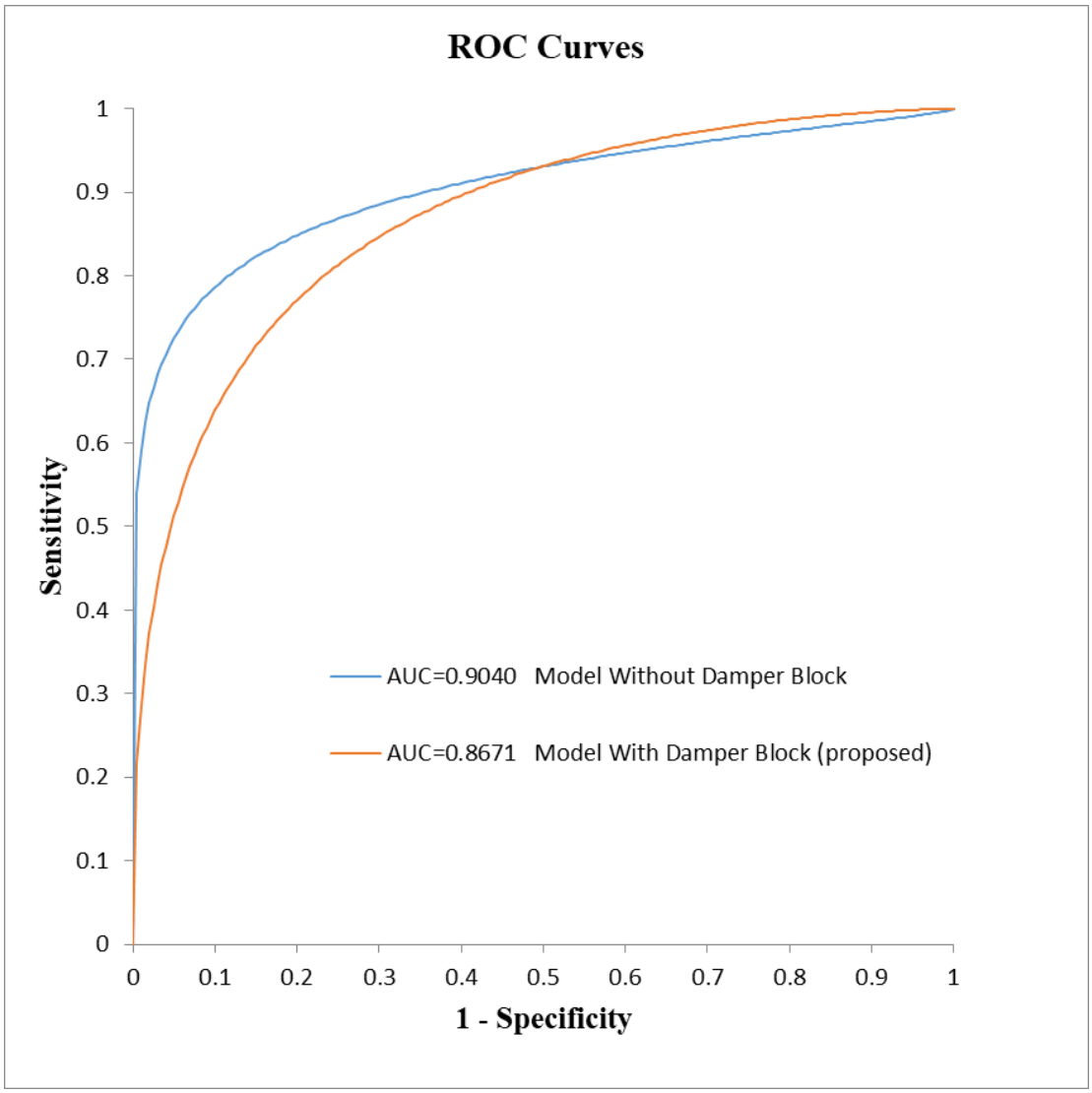


Fig. 4 - 4 The ROC curve for the proposed model with or without the damper block[2].

4.5 With and Without Clinical Data

In this experiment, we compared the performance with and without clinical data. We have found that incorporating clinical data significantly improves the performance of the model, as shown in Table 4 - 5 and Fig. 4 - 5. The accuracy is 86.03%, sensitivity is 82.86% and specificity is 86.69%.

In this experiment, we use DeLong's test[61] to evaluate the p-value, as shown in Table 4 - 6. There is a significant difference between the model with and without clinical data.

Table 4 - 5 The performances for the proposed model with and without clinical data.

Clinical Data	ACC (%)	SEN (%)	SPEC (%)	PPV (%)	NPV (%)	AUC
Without	68.68 ±13.69	47.62 ±24.97	72.40 ±18.79	24.99 ±14.04	89.36 ±3.43	0.6534 ±0.0978
With	**86.03 ±6.45**	**82.86 ±11.95**	**86.69 ±7.84**	**55.29 ±18.35**	**96.72 ±2.33**	**0.8908 ±0.0470**

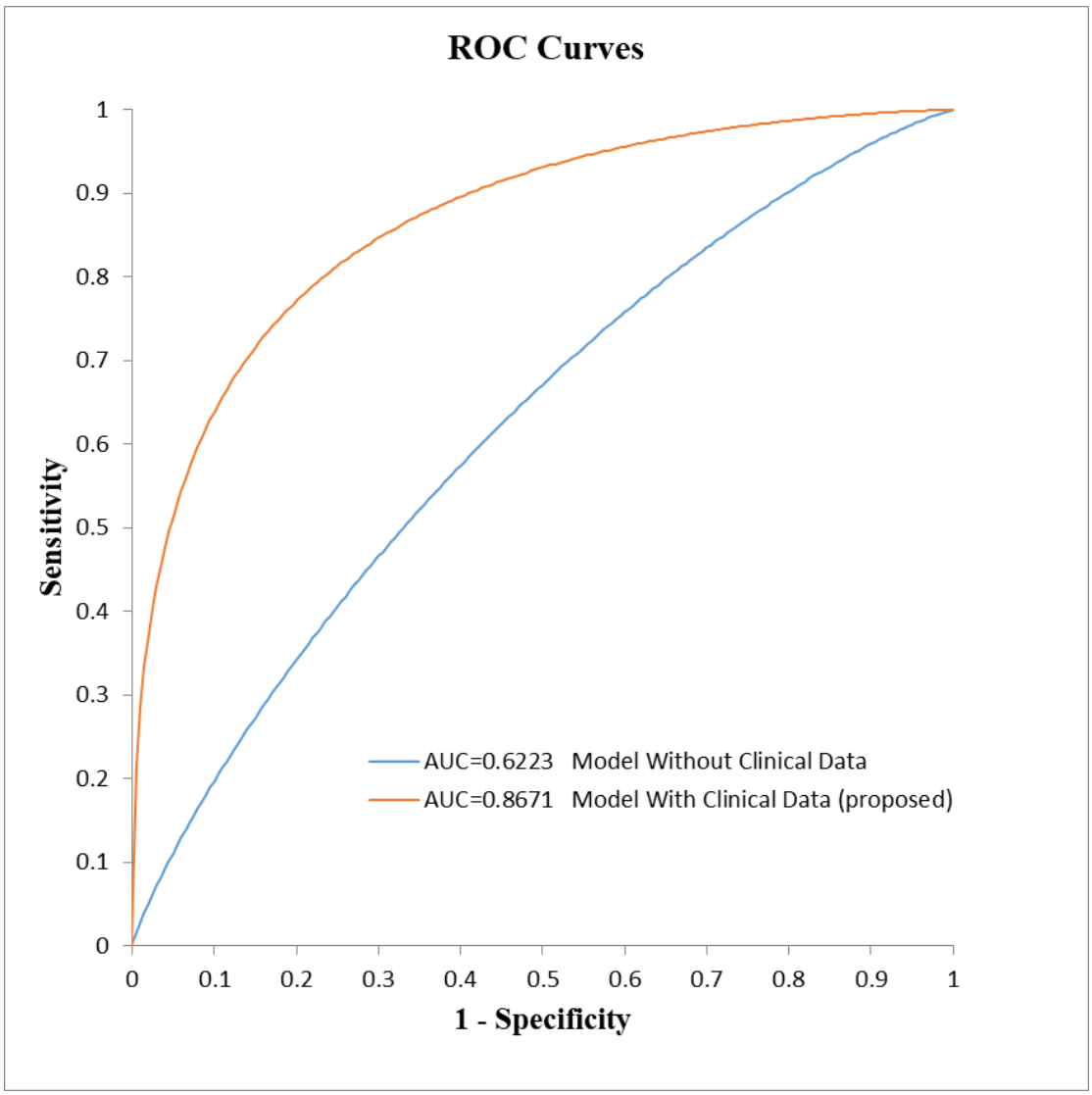


Fig. 4 - 5 The ROC curve for the proposed model with or without clinical data.

Table 4 - 6 The p-value between the model with or without clinical data. The "*" symbol represents a significant difference.

Clinical Data	DeLong's Test
Without	<0.0001*
With	-

4.6 Cross Entropy Loss and Focal Loss

In this experiment, we compared the performance of Focal Loss[52] and Cross Entropy Loss, as shown in Table 4 - 7 and Fig. 4 - 6. The model with Focal Loss[52]

gets the accuracy of 86.03%, sensitivity of 82.86% and specificity of 86.69%. We found that Focal Loss[52] achieves better classification results than Cross Entropy Loss in terms of accuracy, sensitivity and specificity.

Table 4 - 7 The performances for the model with Focal Loss[52] or Cross Entropy Loss.

	ACC (%)	SEN (%)	SPEC (%)	PPV (%)	NPV (%)	AUC
Cross Entropy	85.19	80.00	86.21	54.05	96.26	**0.8954**
Loss	±6.14	±16.29	±8.33	±17.63	±2.87	**±0.0358**
Focal Loss	**86.03**	**82.86**	**86.69**	**55.29**	**96.72**	0.8908
	±6.45	**±11.95**	**±7.84**	**±18.35**	**±2.33**	±0.0470

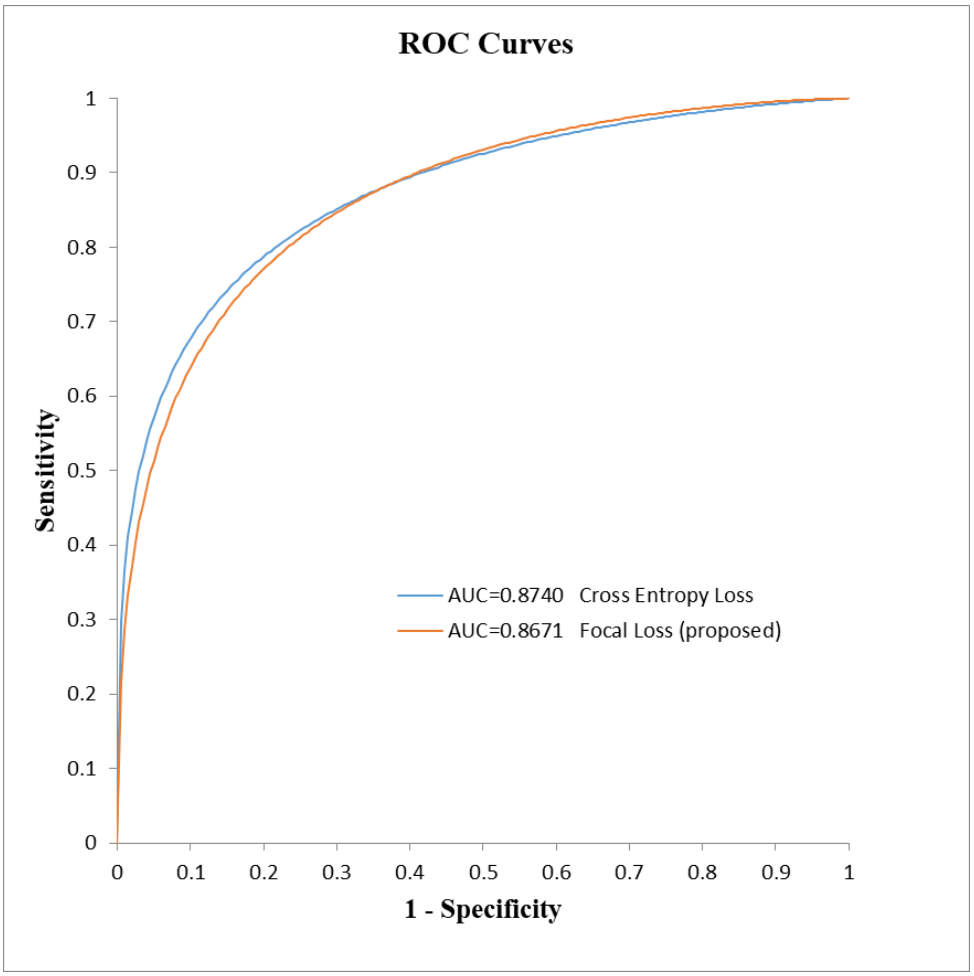


Fig. 4 - 6 The ROC curve for Focal Loss[52] and Cross Entropy Loss.

4.7 Comparison with Other Models

In this experiment, we compared different CNN models as shown in Table 4 - 8 and Fig. 4 - 7. The proposed model achieves the highest accuracy of 86.03%, sensitivity of 82.86% and specificity of 86.69%. On the other hand, Swin Transformer[42] obtains the highest sensitivity (82.86%), the same as the proposed model.

Among the ResNet family[43-45], we find that ResNeXt[45] has the highest accuracy (84.77%), and the lowest sensitivity (76.67%). ResNeSt[43] and ResNet[44] share similar result in terms of accuracy, sensitivity and specificity.

In this experiment, we use DeLong's test[61] to evaluate the p-value, as shown in Table 4 - 9. The ResNeXt[45] and ResNeSt[43] are significantly different from the proposed model.

Table 4 - 8 The performances for the different models.

	ACC (%)	SEN (%)	SPEC (%)	PPV (%)	NPV (%)	AUC
ResNet	84.32 ±8.06	79.52 ±7.45	85.14 ±9.19	51.53 ±16.94	96.04 ±1.70	0.8747 ±0.0428
ResNeXt	84.77 ±4.47	76.67 ±12.42	86.21 ±6.90	52.23 ±16.58	95.73 ±1.98	0.8787 ±0.0508
ResNeSt	84.30 ±6.18	79.52 ±16.11	85.10 ±6.47	49.05 ±11.99	96.12 ±3.20	0.8785 ±0.0319
Swin Transformer	83.48 ±5.03	82.86 ±11.95	83.71 ±6.99	48.00 ±10.41	96.66 ±2.21	0.8940 ±0.0495

Proposed Model	86.03 ±6.45	82.86 ±11.95	86.69 ±7.84	55.29 ±18.35	96.72 ±2.33	0.8908 ±0.0470

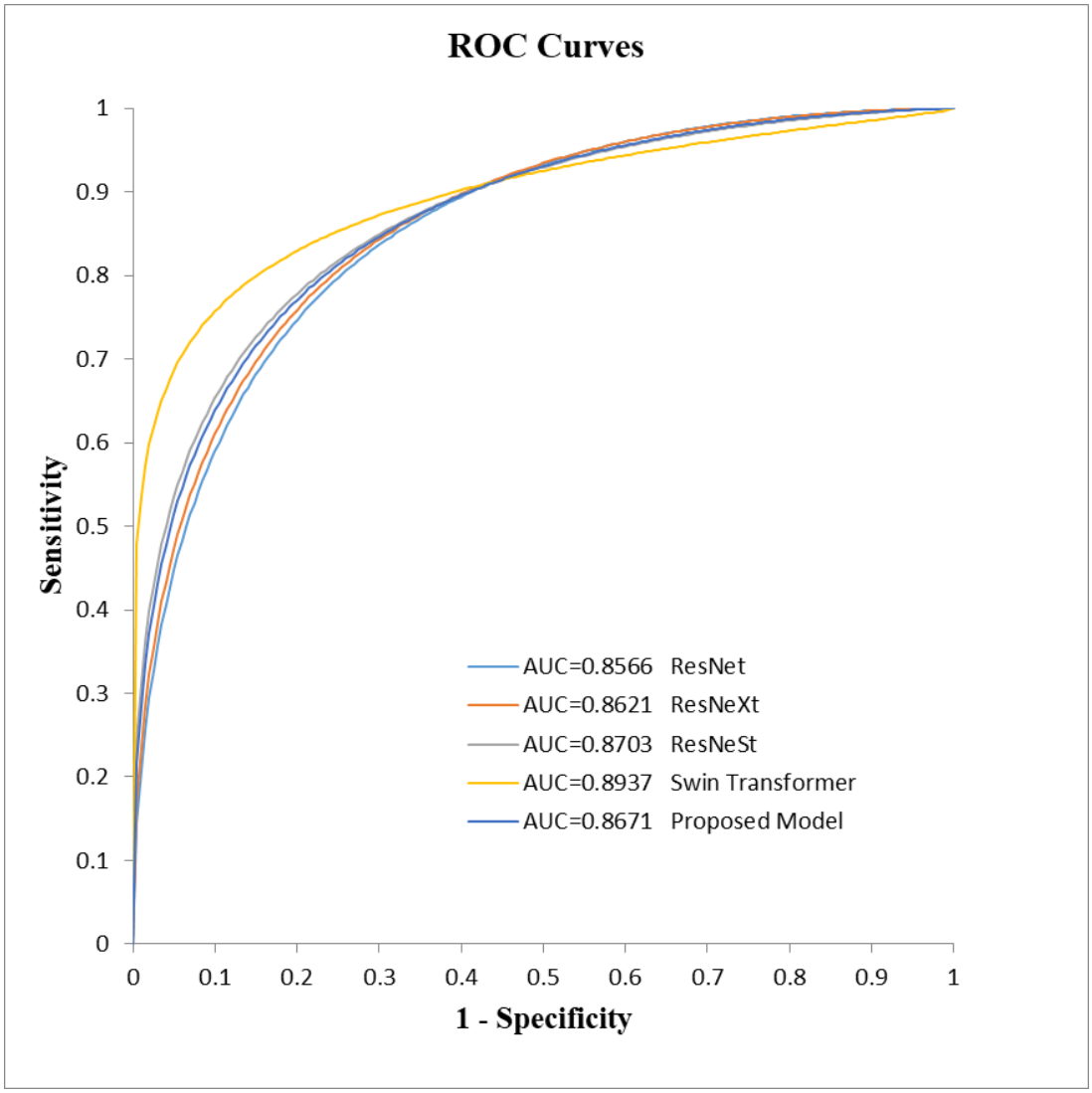


Fig. 4 - 7 The ROC curve for the proposed model and other models.

Table 4 - 9 The p-value between the proposed model and other models. The "*" symbol represents a significant difference.

CNN Models	DeLong's Test
ResNet	0.3258
ResNeXt	0.0390*
ResNeSt	0.0194*
Swin Transformer	0.1320
Proposed Model	-

Chapter 5 Discussion

In the study, we aimed to accurately predict the three-year survival outcomes of lung cancer patients using CNNs. We utilized ConvNeXt[1] as the backbone model and incorporated an attention block to assist the model in filtering and extracting image features relevant to survival prediction.

Additionally, in Table 5 - 1, we found that patients who did not survive tended to have larger tumor sizes, whereas those who survived had a smaller proportion of tumors exceeding 3 cm. This correlation suggests a clear association between tumor size and survival outcomes. However, CNNs require fixed image input sizes. To ensure that CNNs capture information about tumor size, we bring in a damper block[2] that combines detailed image features obtained from CNNs with tumor size features. Finally, since survival outcomes are not solely determined by tumor size, location, and other factors related to tumor, but also influenced by the clinical and pathological status of the patients, we merged the features obtained from the images with clinical and pathology information. These combined features were then used to train a neural network and obtain survival outcome predictions. The final accuracy was 86.03%, sensitivity was 82.86%, and AUC was 0.8908. These results demonstrate that the model successfully extracted relevant information related to survival outcomes from

both dual energy CT scans as well as clinical and pathology data, providing accurate predictions.

In the experiments across different keV, we found that the dataset captured at 140 keV with contrast agent enhancement yielded the best results. However, the datasets captured at 40 keV and 100 keV without contrast agent enhancement also achieved the best sensitivity results, and their accuracy was comparable to the 140 keV dataset with contrast agent enhancement. These results suggest that different energy levels of images offer distinct advantages. Both low-energy images with higher tumor contrast and high-energy images with smoother tumor surfaces enable the model to capture crucial features relevant to the survival prediction.

On the other hand, although the 40 keV dataset and 70 keV dataset with contrast agent enhancement exhibited lower sensitivities compared to the 140 keV dataset with contrast agent enhancement, this could be possibly attributed to label imbalance. Consequently, we cannot conclude that training the 140 keV dataset with contrast agent enhancement when more data is collected will consistently guarantee the most accurate predictions.

In the evaluation of different model architectures, we first explored the use of various attention blocks to enhance performance**.** We found that only the GCT block[4] achieved higher Sensitivity. Although the SE block[3], BAM block[59], and CBAM

block[60] performed well in terms of accuracy, their sensitivity remained low, and there was a notable gap between sensitivity and specificity. This indicates that they tend to overly predict one class, suggesting that they did not genuinely capture the necessary information for each label, especially under label imbalance conditions. Among the experiments of combinations, SE[3] + GCT[4] yielded the best results in terms of both accuracy and sensitivity. BAM[59] + GCT[4] improved sensitivity over BAM block[59] but led to a decrease in accuracy and specificity. On the other hand, CBAM[60] + GCT[4] drastically reduced sensitivity over CBAM block[60]. Therefore, BAM[59] + GCT[4] and CBAM[60] + GCT[4] are not suitable for this study.

Considering that BAM block[59] and CBAM block[60] include spatial attention and channel attention architectures, we concluded that incorporating spatial attention after the last ConvNeXt[1] block in the third stage of ConvNeXt[1] is not ideal. Therefore, we retained only the channel attention. However, we cannot imply that spatial attention should not be applied after other ConvNeXt[1] blocks. Further assessments with balanced data for each label are crucial to validate the generalization of this finding, and identify suitable positions for incorporating spatial attention.

We also designed the experiments regarding the placement of attention blocks. We hypothesized that features in the third and fourth stages of ConvNeXt[1] would be

more fine-grained. Thus, we primarily focused on incorporating attention blocks in these two stages. Through experimentation, we found that attaching the attention block after the last ConvNeXt[1] block in the third stage of ConvNeXt[1] resulted in the best predictive performance. Incorporating attention blocks at all positions could enhance specificity but led to a significant decrease in sensitivity. The phenomenon could be attributed to label imbalance, any slight modification to the model architecture could affect the sensitivity of labels with fewer image data representation. Thus, we cannot conclude that the current position of the attention block will consistently yield the best results. To make more comprehensive assessments, having roughly equal data for each label is essential. However, the experiments showed that without incorporating attention blocks, the results for accuracy, sensitivity, and specificity were inferior to the final model. This observation indicates that placing attention blocks in appropriate positions can help CNNs extract more relevant features for survival prediction.

Moreover, we explored the impact of the damper block[2]. We observed that incorporating the damper block[2] improved the performance of the model in terms of accuracy, sensitivity, and specificity. This indicates that the damper block[2] indeed helps the model capture more relevant features related to survival prediction, which aligns with the intuition that incorporating tumor size information enhances the

predictive capabilities of the model. For the deceased samples in Fig. 5 - 1, they were predicted incorrectly by the model when the damper block[2] was not incorporated. After adding the damper block[2], the predictions became accurate. Both the samples have large tumors; the tumor in the left sample is larger than 7 cm, and the tumor in the right sample is larger than 10 cm. Introducing the damper block[2] improved the ability of the model to accurately predict samples with large tumors and labeled outcomes of death. This demonstrates the beneficial impact of tumor size information for CNNs to process image features. Moreover, for the survived samples in Fig. 5 - 2, we observed that the tumors in these samples are generally small and mostly well-separated from their surroundings, resulting in distinct tumor contours. The addition of tumor size information likely enables the model to precisely identify the location of the tumor, leading to accurate predictions.

However, we noticed that some data that was correctly predicted without the damper block[2], resulted in incorrect predictions when the damper block[2] was added. In Fig. 5 - 3, the five samples were predicted correctly by the model without the damper block[2], and introducing the damper block[2] led to incorrect predictions. We found that in most of these samples, the tumors are connected to their surroundings, preventing the model from concisely identifying clear tumor contours. In such cases, the model has the ability to identify tumor features even if we do not

provide more information. The introduction of the damper block[2] could potentially constrain the capability of the model, causing the model to mistake non-tumor areas as part of the tumor, ultimately resulting in incorrect predictions.

Additionally, we compared the performance of the model with and without the incorporation of clinical data and found that without clinical data, the model struggled to accurately predict the survival status of patients three years later. Clinical data provides crucial information about tumors, such as genetic mutations, organ metastasis, sites of recurrence, as well as other essential factors like family history of lung cancer, presence of diabetes and hypertension. Therefore, incorporating clinical data helps the model capture more critical features, ultimately leading to accurate predictions of survival outcomes three years ahead.

We also designed the experiment comparing Focal Loss[52] with Cross Entropy Loss. We observed that using Focal Loss[52] as the loss function resulted in improved accuracy, sensitivity, and specificity. Notably, no data predicted correctly using Cross Entropy Loss were misclassified when using Focal Loss[52]. This indicates that Focal Loss[52] indeed outperforms Cross Entropy Loss in terms of prediction performance. One significant reason for the superior performance of Focal Loss[52] is that Cross Entropy Loss can be seen as a special case of Focal Loss[52]. Consequently, Focal Loss[52] is specifically designed to address the limitations of Cross Entropy Loss. By

carefully adjusting the modulating factors, Focal Loss[52] should be undoubtedly better for improving model performance.

The samples shown in Fig. 5 - 4 are the only two that were predicted incorrectly using Cross Entropy Loss but were predicted correctly after switching to Focal Loss[52]. On the left are the sample that died within three years, while on the right are the sample that survived after three years. We found that there is a small tumor in the left sample, yet the outcome was death, whereas on the right side, there is a large tumor exceeding 5 cm, but the labeled outcome is survival. For a model trained with Cross Entropy Loss, these two samples are considered challenging to predict. However, Cross Entropy Loss is easily influenced by the loss from numerous well-predicted samples, which prevents the model using Cross Entropy Loss from accurately predicting these two samples. Nonetheless, Focal Loss[52] could effectively reduce the impact of easily predictable samples, enabling the model to focus more on difficult samples. As a result, the model using Focal Loss[52] successfully predicts these challenging samples correctly.

After comparing the performances by modifying the architecture of the proposed model, we designed the comparative experiments with other models. We observed that the proposed model outperformed the ResNet family[43-45] in terms of accuracy, sensitivity, and specificity. This improvement can be attributed to that ConvNeXt[1] is

an enhancement of the ResNet family[43-45], incorporating the advantages of Vision Transformers, which naturally leads to better predictive results compared to the original ResNet family[43-45].

Furthermore, Swin Transformer[42] achieved the same sensitivity as the proposed model, and the difference between sensitivity and specificity is not significant. This indicates that Swin Transformer[42] was less affected by label imbalance. Overall, the proposed model achieved the highest accuracy, sensitivity, and specificity, demonstrating that ConvNeXt[1] effectively integrates the strengths of both the ResNet family[43-45] and Swin Transformer[42].

Table 5 - 1 The amount of data with tumor size greater than three centimeters.

	Deceased (n=34)	Survivor (n=202)
Tumor Size (≥ 3 cm)	33 (97%)	66 (33%)

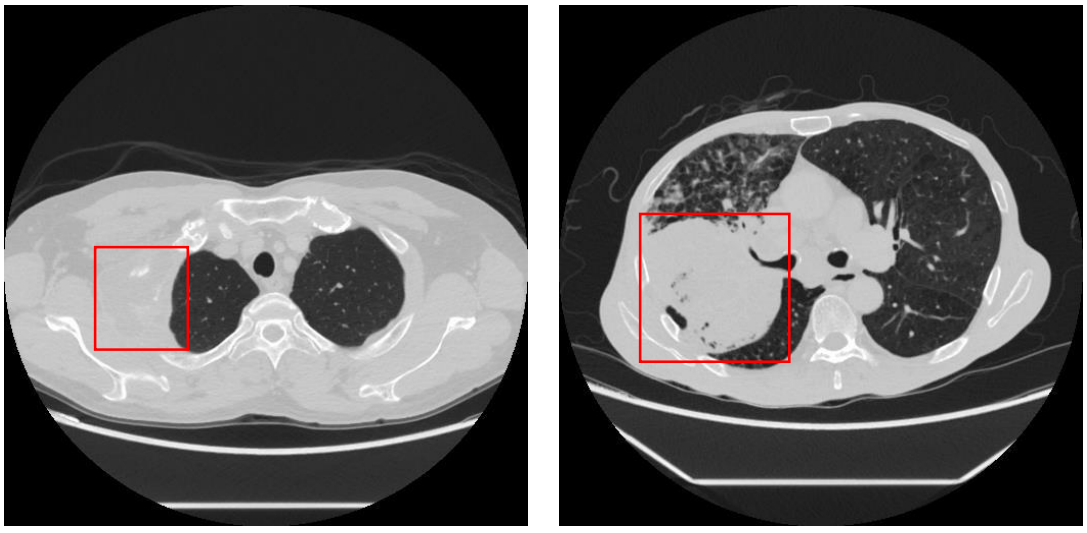


Fig. 5 - 1 The only two deceased samples that the model predicts wrongly without the damper block[2], and predicts correctly after adding the damper block[2].

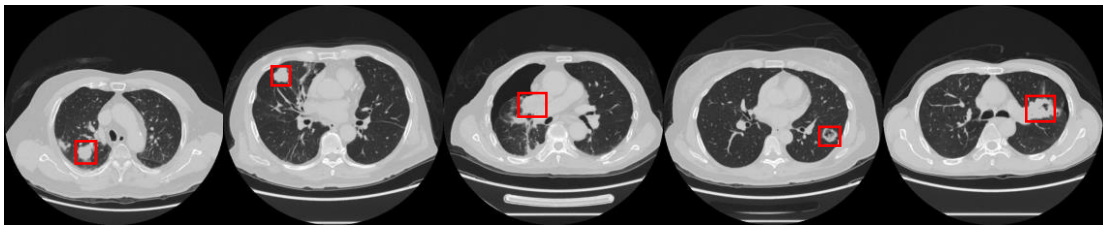


Fig. 5 - 2 The five survived samples that the model predicts wrongly without the damper block[2], and predicts correctly after adding the damper block[2].

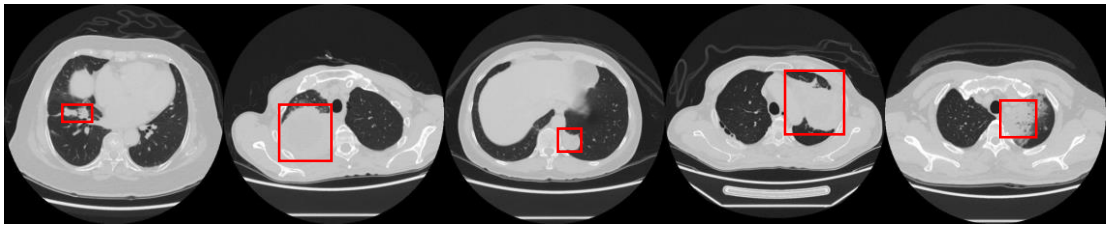


Fig. 5 - 3 The five survived samples that the model predicts correctly without the damper block[2], but predicts wrongly after adding the damper block[2].

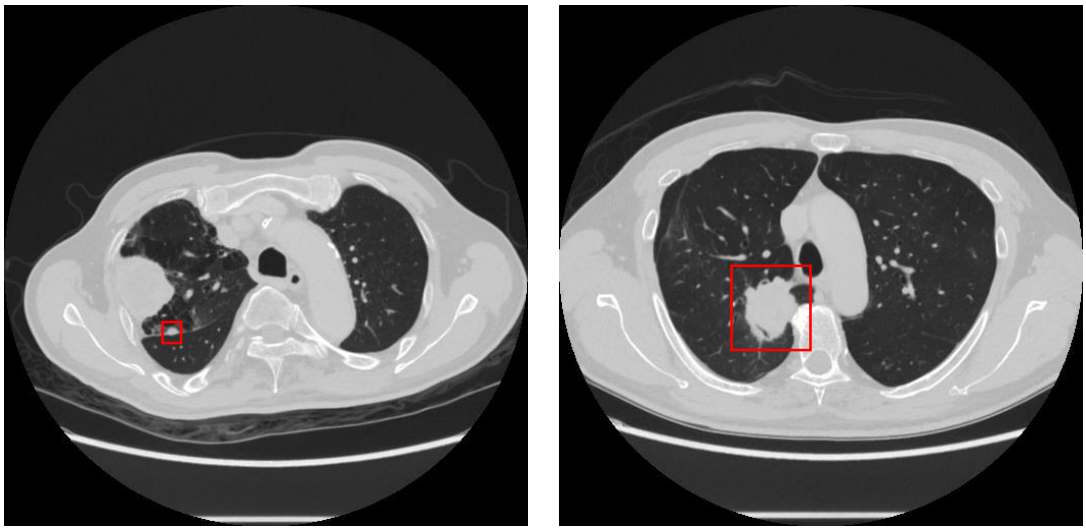


Fig. 5 - 4 The only two samples that the model predicts wrongly using Cross Entropy Loss, and predicts correctly after switching to Focal Loss[52].

Chapter 6 Conclusion

We designed a CNN to predict the 3-year survival outcome of lung cancer patients. By making certain architectural adjustments and integrating different datasets, we achieved favorable results in terms of accuracy, sensitivity, specificity, and AUC. Another advantage of the proposed model is the ability to directly transform dual energy CT into survival predictions, streamlining the prediction process and reducing human resource consumption through automation.

During this experiment, we encountered the issue of label imbalance in the dataset. We addressed this problem through adjustments in the model architecture and dataset handling. However, the combination of label imbalance and insufficient data may result in significant fluctuations in sensitivity and specificity. Additionally, the existing dataset might not fully reflect the real-world data distribution. Therefore, continuous data collection and ongoing improvements to the proposed model are necessary to achieve highly accurate and convincing survival predictions that align with real-world scenarios

References

[1] Z. Liu, H. Mao, C.-Y. Wu *et al.*, "A convnet for the 2020s." pp. 11976-11986.

[2] Y.-W. Wang, C.-J. Chen, H.-C. Huang *et al.*, “Dual energy CT image prediction on primary tumor of lung cancer for nodal metastasis using deep learning,” *Computerized Medical Imaging and Graphics,* vol. 91, pp. 101935, 2021.

[3] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks." pp. 7132-7141.

[4] Z. Yang, L. Zhu, Y. Wu *et al.*, "Gated channel transformation for visual recognition." pp. 11794-11803.

[5] H. Sung, J. Ferlay, R. L. Siegel *et al.*, “Global cancer statistics 2020: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries,” *CA: a cancer journal for clinicians,* vol. 71, no. 3, pp. 209-249, 2021.

[6] B. S. Chhikara, and K. Parang, “Global Cancer Statistics 2022: the trends projection analysis,” *Chemical Biology Letters,* vol. 10, no. 1, pp. 451-451, 2023.

[7] R. L. Siegel, K. D. Miller, and A. Jemal, “Cancer statistics, 2020,” *CA: A Cancer Journal for Clinicians,* vol. 70, no. 1, pp. 7-30, 2020.

[8] R. L. Siegel, K. D. Miller, H. E. Fuchs *et al.*, “Cancer Statistics, 2021,” *CA: A Cancer Journal for Clinicians,* vol. 71, no. 1, pp. 7-33, 2021.

[9] R. L. Siegel, K. D. Miller, H. E. Fuchs *et al.*, “Cancer statistics, 2022,” *CA: a cancer journal for clinicians,* vol. 72, no. 1, pp. 7-33, 2022.

[10] R. L. Siegel, K. D. Miller, N. S. Wagle *et al.*, “Cancer statistics, 2023,” *CA: a cancer journal for clinicians,* vol. 73, no. 1, pp. 17-48, 2023.

[11] S. Birring, and M. Peake, "Symptoms and the early diagnosis of lung cancer," 4, BMJ Publishing Group Ltd, 2005, pp. 268-269.

[12] J. Vansteenkiste, C. Dooms, C. Mascaux *et al.*, “Screening and early—detection of lung cancer,” *Annals of Oncology,* vol. 23, pp. x320-x327, 2012.

[13] Y. She, Z. Jin, J. Wu *et al.*, “Development and validation of a deep learning model for non–small cell lung cancer survival,” *JAMA network open,* vol. 3, no. 6, pp. e205842-e205842, 2020.

[14] K. A. Miles, “How to use CT texture analysis for prognostication of non-small cell lung cancer,” *Cancer Imaging,* vol. 16, pp. 1-6, 2016.

[15] N. Howlader, G. Forjaz, M. J. Mooradian *et al.*, “The effect of advances in

lung-cancer treatment on population mortality," *New England Journal of Medicine,* vol. 383, no. 7, pp. 640-649, 2020.

[16] M. Riihimäki, A. Hemminki, M. Fallah *et al.*, "Metastatic sites and survival in lung cancer," *Lung cancer,* vol. 86, no. 1, pp. 78-84, 2014.

[17] A. Agrawal, S. Misra, R. Narayanan *et al.*, "Lung cancer survival prediction using ensemble data mining on SEER data," *Scientific Programming,* vol. 20, no. 1, pp. 29-42, 2012.

[18] B. F. Hankey, L. A. Ries, and B. K. Edwards, "The surveillance, epidemiology, and end results program: a national resource," *Cancer Epidemiology Biomarkers & Prevention,* vol. 8, no. 12, pp. 1117-1121, 1999.

[19] Y.-H. Lai, W.-N. Chen, T.-C. Hsu *et al.*, "Overall survival prediction of non-small cell lung cancer by integrating microarray and clinical data with deep learning," *Scientific reports,* vol. 10, no. 1, pp. 4679, 2020.

[20] B. He, W. Zhao, J.-Y. Pi *et al.*, "A biomarker basing on radiomics for the prediction of overall survival in non–small cell lung cancer patients," *Respiratory research,* vol. 19, no. 1, pp. 1-8, 2018.

[21] J. Gu, Z. Wang, J. Kuen *et al.*, "Recent advances in convolutional neural networks," *Pattern recognition,* vol. 77, pp. 354-377, 2018.

[22] Z. Li, F. Liu, W. Yang *et al.*, "A survey of convolutional neural networks: analysis, applications, and prospects," *IEEE transactions on neural networks and learning systems,* 2021.

[23] S. Albawi, T. A. Mohammed, and S. Al-Zawi, "Understanding of a convolutional neural network." pp. 1-6.

[24] G. Litjens, T. Kooi, B. E. Bejnordi *et al.*, "A survey on deep learning in medical image analysis," *Medical image analysis,* vol. 42, pp. 60-88, 2017.

[25] S. P. Singh, L. Wang, S. Gupta *et al.*, "3D deep learning on medical images: a review," *Sensors,* vol. 20, no. 18, pp. 5097, 2020.

[26] M. Nishio, K. Fujimoto, H. Matsuo *et al.*, "Lung cancer segmentation with transfer learning: usefulness of a pretrained model constructed from an artificial dataset generated using a generative adversarial network," *Frontiers in artificial intelligence,* vol. 4, pp. 694815, 2021.

[27] S. P. Primakov, A. Ibrahim, J. E. van Timmeren *et al.*, "Automated detection and segmentation of non-small cell lung cancer computed tomography images," *Nature communications,* vol. 13, no. 1, pp. 3423, 2022.

[28] D. Shen, G. Wu, and H.-I. Suk, "Deep learning in medical image analysis," *Annual review of biomedical engineering,* vol. 19, pp. 221-248, 2017.

[29] U. Kamal, A. M. Rafi, R. Hoque *et al.*, "Lung cancer tumor region segmentation using recurrent 3d-denseunet." pp. 36-47.

[30] Z. Guo, J. Yang, L. Zhao *et al.*, “3D SAACNet with GBM for the classification of benign and malignant lung nodules,” *Computers in Biology and Medicine,* vol. 153, pp. 106532, 2023.

[31] E. S. Neal Joshua, D. Bhattacharyya, M. Chakkravarthy *et al.*, “3D CNN with visual insights for early detection of lung cancer using gradient-weighted class activation,” *Journal of Healthcare Engineering,* vol. 2021, pp. 1-11, 2021.

[32] M. Beeres, J. Trommer, C. Frellesen *et al.*, “Evaluation of different keV-settings in dual-energy CT angiography of the aorta using advanced image-based virtual monoenergetic imaging,” *The International Journal of Cardiovascular Imaging,* vol. 32, pp. 137-144, 2016.

[33] T. D. DenOtter, and J. Schubert, “Hounsfield unit,” 2019.

[34] X. Ying, "An overview of overfitting and its solutions." p. 022022.

[35] M. L. Richter, W. Byttner, U. Krumnack *et al.*, "(Input) size matters for CNN classifiers." pp. 133-144.

[36] S. Yang, W. Xiao, M. Zhang *et al.*, “Image data augmentation for deep learning: A survey,” *arXiv preprint arXiv:2204.08610,* 2022.

[37] L. Perez, and J. Wang, “The effectiveness of data augmentation in image classification using deep learning,” *arXiv preprint arXiv:1712.04621,* 2017.

[38] Á. López-Encuentra, J. L. Duque-Medina, R. Rami-Porta *et al.*, “Staging in lung cancer: is 3 cm a prognostic threshold in pathologic stage I non-small cell lung cancer?: a multicenter study of 1,020 patients,” *Chest,* vol. 121, no. 5, pp. 1515-1520, 2002.

[39] G. Brauwers, and F. Frasincar, “A general survey on attention mechanisms in deep learning,” *IEEE Transactions on Knowledge and Data Engineering,* 2021.

[40] S. R. Dubey, S. K. Singh, and B. B. Chaudhuri, “Activation functions in deep learning: A comprehensive survey and benchmark,” *Neurocomputing,* 2022.

[41] A. Dosovitskiy, L. Beyer, A. Kolesnikov *et al.*, “An image is worth 16x16 words: Transformers for image recognition at scale,” *arXiv preprint arXiv:2010.11929,* 2020.

[42] Z. Liu, Y. Lin, Y. Cao *et al.*, "Swin transformer: Hierarchical vision transformer using shifted windows." pp. 10012-10022.

[43] H. Zhang, C. Wu, Z. Zhang *et al.*, "Resnest: Split-attention networks." pp. 2736-2746.

[44] K. He, X. Zhang, S. Ren *et al.*, "Deep residual learning for image recognition." pp. 770-778.

[45] S. Xie, R. Girshick, P. Dollár *et al.*, "Aggregated residual transformations for deep neural networks." pp. 1492-1500.

[46] S. Ioffe, and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift." pp. 448-456.

[47] J. L. Ba, J. R. Kiros, and G. E. Hinton, “Layer normalization,” *arXiv preprint arXiv:1607.06450*, 2016.

[48] D. Hendrycks, and K. Gimpel, “Gaussian error linear units (gelus),” *arXiv preprint arXiv:1606.08415*, 2016.

[49] H. Touvron, M. Cord, A. Sablayrolles *et al.*, "Going deeper with image transformers." pp. 32-42.

[50] K. Fukushima, “Cognitron: A self-organizing multilayered neural network,” *Biological cybernetics*, vol. 20, no. 3-4, pp. 121-136, 1975.

[51] V. Nair, and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines." pp. 807-814.

[52] T.-Y. Lin, P. Goyal, R. Girshick *et al.*, "Focal loss for dense object detection." pp. 2980-2988.

[53] I. Loshchilov, and F. Hutter, “Decoupled weight decay regularization,” *arXiv preprint arXiv:1711.05101*, 2017.

[54] I. Loshchilov, and F. Hutter, “Sgdr: Stochastic gradient descent with warm restarts,” *arXiv preprint arXiv:1608.03983*, 2016.

[55] P. Goyal, P. Dollár, R. Girshick *et al.*, “Accurate, large minibatch sgd: Training imagenet in 1 hour,” *arXiv preprint arXiv:1706.02677*, 2017.

[56] F. Mosteller, and J. W. Tukey, “Data analysis, including statistics,” *Handbook of social psychology*, vol. 2, pp. 80-203, 1968.

[57] R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection." pp. 1137-1145.

[58] D. Berrar, "Cross-Validation," 2019.

[59] J. Park, S. Woo, J.-Y. Lee *et al.*, “Bam: Bottleneck attention module,” *arXiv preprint arXiv:1807.06514*, 2018.

[60] S. Woo, J. Park, J.-Y. Lee *et al.*, "Cbam: Convolutional block attention module." pp. 3-19.

[61] E. R. DeLong, D. M. DeLong, and D. L. Clarke-Pearson, “Comparing the areas under two or more correlated receiver operating characteristic curves: a nonparametric approach,” *Biometrics*, pp. 837-845, 1988.