



**REPUBLIC OF TÜRKİYE
ADANA ALPARSLAN TÜRKES SCIENCE AND TECHNOLOGY
UNIVERSITY**

**GRADUATE SCHOOL
SOFTWARE ENGINEERING**

**SEGMENTATION OF MAGNETIC RESONANCE IMAGES WITH
DEEP LEARNING FOR THE DIAGNOSIS OF HEPATOCELLULAR
CARCINOMA**

Ömer IŞIK

M.Sc.

ADANA 2025



**REPUBLIC OF TÜRKİYE
ADANA ALPARSLAN TÜRKŞ SCIENCE AND TECHNOLOGY
UNIVERSITY**

**GRADUATE SCHOOL
SOFTWARE ENGINEERING**

**SEGMENTATION OF MAGNETIC RESONANCE IMAGES WITH
DEEP LEARNING FOR THE DIAGNOSIS OF HEPATOCELLULAR
CARCINOMA**

Ömer IŞIK

M.Sc.

**THESIS ADVISOR
Asst.Prof.Dr. Çağatay Neftali TÜLÜ**

**SECOND ADVISOR
Prof.Dr. Turgay İBRİKÇİ**

ADANA, 2025

DECLARATION OF CONFORMITY

In thesis, which was prepared following the thesis writing rules of Adana Alparslan Türkeş Science and Technology University Institute of Graduate School, I declare that I provided all the information, documents, evaluations, and results in accordance with scientific ethics and moral codes, without resorting to any means or assistance that would be contrary to scientific ethics and traditions. I also declare that I referred to all of the articles I used in this study with appropriate references and accept all moral and legal consequences if a situation is found contrary to my statement regarding my work.

25/06/2025

Ömer IŞIK

ÖZET

HEPATOSELÜLER KARSİNOMUN TANISI İÇİN DERİN ÖĞRENME İLE MANYETİK REZONANS GÖRÜNTÜLERİNİN SEGMENTASYONU

Ömer IŞIK

Yüksek Lisans, Yazılım Mühendisliği

Danışman: Dr.Öğr.Üyesi Çağatay Neftali TÜLÜ

II. Danışman: Prof.Dr. Turgay İBRİKÇİ

Haziran 2025, 143 sayfa

Amaç: Bu çalışmanın amacı, hepatoselüler karsinoma (HCC) tanısında manyetik rezonans görüntü (MRI)'ler üzerinde gerçekleştirilen segmentasyon işlemlerinin doğruluğunu artırmak üzere, farklı görüntü sekansları (T1 ve T2), çözünürlük düzeyleri ve ön işleme tekniklerinin (Gaussian filtreleme, histogram eşitleme) makine öğrenmesi temelli modellerle sistematik olarak karşılaştırmalı analizini yapmaktır.

Yöntem: Çalışmada, ATLAS veri setine ait T1 sekanslı görüntüler ve kuruma ait T2 sekanslı MR görüntüleri kullanılmış; her iki veri seti üç farklı çözünürlükte (128×128 , 256×256 , 512×512) işlenmiştir. Görüntülere Gaussian filtreleme (sigma 1–5) ve iki farklı histogram eşitleme yöntemi (Histogram Equalization (HE) ve Contrast Limited Adaptive Histogram Equalization (CLAHE)) uygulanmıştır. Segmentasyon modellerinin başarımı Dice Similarity Coefficient (DICE), Intersection over Union (IoU), Pixel Accuracy (PA), Recall (REC), Sensitivity (SEN) ve Specificity (SPE) metrikleriyle değerlendirilmiş, elde edilen sonuçlar çok değişkenli istatistiksel testlerle karşılaştırılmıştır.

Bulgular: T1 sekanslı ATLAS veri seti, genel segmentasyon metriklerinde daha istikrarlı ve yüksek başarı gösterirken; T2 sekanslı kendi veri setimiz, bazı metriklerde rekabetçi hatta üstün sonuçlar sunmuştur. Özellikle T2 sekanslı görüntülerde 128×128 çözünürlükte yüksek Recall (0.969), 512×512 çözünürlükte ise yüksek Specificity (0.9936) değerleri elde edilmiştir. Gaussian filtreleme sonucunda sigma 1–2 aralığı en başarılı performansı sunarken, sigma 5 düzeyinde performans düşüşleri gözlenmiştir. Ayrıca, çözünürlük ve filtre düzeyinin Specificity gibi metrikler üzerindeki etkisinin istatistiksel olarak anlamlı olduğu belirlenmiştir ($p < 0.05$).

Sonu: Bu alıřma, karaciğer tmr segmentasyonunda kullanılan MR grntler iin znrlk, sekans tipi ve n iřleme tekniklerinin birlikte deęerlendirilmesi gerektięini vurgulamaktadır. Literatrde sıklıkla yalnızca DICE ve IoU gibi metrikler raporlanırken, bu alıřmada metrikler gruplar arasında istatistiksel olarak da karřılařtırılmıřtır. Bulgular, parametrik olarak optimize edilmiř n iřleme adımlarının yalnızca doęruluęu deęil, iřlem verimlilięini de artırabileceęini gstermektedir. Bu ynyle alıřma, tıbbi grnt segmentasyonu alanına hem yntemsel hem de uygulamalı dzeyde zgn katkılar sunmaktadır.

Anahtar Kelimeler: Hepatoseller Karsinoma, Makine ęrenmesi Manyetik Rezonans Grntleme, T1 Sekansı, T2 Sekansı, Segmentasyon.

ABSTRACT

SEGMENTATION OF MAGNETIC RESONANCE IMAGES WITH DEEP LEARNING FOR THE DIAGNOSIS OF HEPATOCELLULAR CARCINOMA

Ömer IŞIK

M.Sc., Software Engineering

Supervisor: Asst. Prof. Dr. Çağatay Neftali TÜLÜ

Co-Supervisor: Prof. Dr. Turgay İBRİKÇİ

June 2025, 143 pages

Aim: The aim of this study is to systematically perform a comparative analysis of deep learning-based segmentation models to enhance accuracy in the diagnosis of hepatocellular carcinoma (HCC) using magnetic resonance imaging (MRI). The analysis specifically evaluates the impact of different imaging sequences (T1- and T2-weighted), resolution levels, and preprocessing techniques (Gaussian filtering and histogram equalization) on segmentation performance.

Methods: In this study, T1-weighted MR images from the ATLAS dataset and T2-weighted MR images obtained from the institution were used. Both datasets were processed at three different resolutions (128×128 , 256×256 , and 512×512). The images were subjected to Gaussian filtering (sigma 1–5) and two distinct histogram equalization techniques: standard Histogram Equalization (HE) and Contrast-Limited Adaptive Histogram Equalization (CLAHE). The performance of the segmentation models was evaluated using several metrics, including the Dice Similarity Coefficient (DICE), Intersection over Union (IoU), Pixel Accuracy (PA), Recall (REC), Sensitivity (SEN), and Specificity (SPE). The results were compared using multivariate statistical tests.

Results: The T1-weighted ATLAS dataset demonstrated more consistent and higher overall segmentation performance across standard evaluation metrics. However, the institution's T2-weighted dataset yielded competitive and in some cases superior results for specific metrics. Notably, the T2-weighted images achieved a high Recall score of 0.969 at 128×128 resolution and a high Specificity score of 0.9936 at 512×512 resolution. Gaussian filtering produced

optimal results in the sigma 1–2 range, while a noticeable decline in performance was observed at sigma 5. Furthermore, the influence of resolution and filter level on certain metrics particularly Specificity was found to be statistically significant ($p < 0.05$).

Conclusion: The study demonstrates that resolution, MR sequence type, and preprocessing methods must be jointly considered in liver tumor segmentation. Unlike most studies that report only DICE and IoU, this work includes statistical comparisons of multiple metrics. Results show that optimized preprocessing improves both segmentation accuracy and computational efficiency. Thus, the study provides methodological and applied contributions to medical image segmentation.

Keywords: Hepatocellular Carcinoma, Deep Learning, Magnetic Resonance Imaging, T1-Weighted Sequence, T2-Weighted Sequence, Segmentation.

Dedication

This thesis is dedicated to the two most cherished corners of my heart İpek and Eylül.

Our beloved daughter İpek, you are the most precious gift of our hearts, our pride, and the light of our hope. May the light of science illuminate your path in life; may your curiosity always lead you to new horizons. With every step you take, know that we will be watching you with endless love.

And to my dear wife, Eylül... You are the one who has brought beauty into my life with your presence, carried me forward with your patience, and become the peace that lives within my heart. Without your love and support, this journey would have been far more difficult. With you, every challenge became more bearable, and every achievement more meaningful. This thesis would not have been possible without you.

ACKNOWLEDGEMENTS

This thesis study, titled "Prediction of Hepatocellular Carcinoma Disease Prognosis from Radiological, Pathological and Clinical Data Obtained from Patients Using Artificial Intelligence Algorithms," has been conducted as part of the The Scientific and Technological Research Council of Turkey (TUBITAK) 1001 project numbered 122S588. I would like to thank TUBITAK and the Graduate School of Adana Alparslan Türkeş University for their support.



TABLE OF CONTENTS

CERTIFICATION OF APPROVAL	i
ÖZET	iii
ABSTRACT	v
<i>Dedication</i>	vii
TABLE OF CONTENTS	ix
LIST OF FIGURES	xi
LIST OF TABLES	xiv
LIST OF ABBREVIATIONS	xv
1.INTRODUCTION	1
2.LITERATURE REVIEW	9
3.MATERIAL AND METHODS	17
3.1.Datasets	18
3.2.Medical Image Segmentation	20
3.3.Segmentation Using Artificial Intelligence Algorithms	21
3.3.1. Convolutional Neural Network	22
3.3.2. U-Net	29
3.3.3. Digital Image Processing	34
3.4.Performance Evaluation	40
3.5.Model Parameters	45
4.RESULTS AND DISCUSSION	47
4.1.ATLAS Dataset Results	47
4.1.1. Segmentation Performance Evaluation According to Different Resolutions (Different Dimensions)	47
4.1.2. Effect of Gaussian Filter Parameter (Sigma) on Segmentation Performance at Different Resolutions	52
4.1.3. Effect of Histogram Equalization and CLAHE Methods on Segmentation Performance at Different Resolutions	57
4.2.Results of CU-HCC Dataset	62
4.2.1. Segmentation Performance Evaluation According to Different Resolutions (Different Dimensions)	62

4.2.2. Effect of Gaussian Filter Parameter (Sigma) on Segmentation Performance at Different Resolutions	64
4.2.3. Effect of Histogram Equalization and CLAHE Methods on Segmentation Performance at Different Resolutions	70
4.3.Statistical Comparison of the Obtained Results	75
4.4.Discussion	109
5.CONCLUSIONS AND FUTURE WORKS	113
6. REFERENCES	117
APPENDIX	124
VITA	125



LIST OF FIGURES

Figure 1. Anatomical location of the liver in the human body.	1
Figure 2. Workflow of the MRI-Based Liver Tumor Segmentation Using U-Net	
Architecture and Preprocessing Techniques	18
Figure 3. Sample images an masks in the Atlas Dataset and CU-HCC Dataset	20
Figure 4. Schematic of Convolutional Neural Network.....	23
Figure 5. Convolutional Layer.	24
Figure 6. Convolution Operation with a Kernel Size.....	24
Figure 7. Pooling Layer	25
Figure 8. Fully Connected Layer	26
Figure 9. U-Net architecture for image segmentation. Adapted from U-Net	30
Figure 10. Different size images, masks and result predictions of the same patient	47
Figure 11. Framed form of mask and prediction dimensions in the input image of the same patient in different sizes	48
Figure 12. Segmentation masks prediction result	49
Figure 13. Mask and prediction results for the same patient's image given as input in different sizes	50
Figure 14. Segmentation performance of different sized images of ATLAS Dataset	51
Figure 15. Segmentation Performance of Different Gaussian Sigma Values on 128x128 ATLAS Dataset Images	53
Figure 16. Segmentation Performance of Different Gaussian Sigma Values on 256x256 ATLAS Dataset Images	55
Figure 17. Segmentation Performance of Different Gaussian Sigma Values on 512x512 ATLAS Dataset Images	57
Figure 18. Segmentation Performance of Different Histogram Equalization Methods on 128x128 ATLAS Dataset Images	59
Figure 19. Segmentation Performance of Different Histogram Equalization Methods on 256x256 ATLAS Dataset Images	60
Figure 20. Segmentation Performance of Different Histogram Equalization Methods on 512x512 ATLAS Dataset Images	62
Figure 21. Segmentation performance of images of different sizes in CU-HCC Dataset ..	64

Figure 22. Segmentation Performance of Different Gaussian Sigma Values on 128x128 Size Images of CU-HCC Dataset	66
Figure 23. Segmentation Performance of Different Gaussian Sigma Values on 256x256 Size Images of CU-HCC Dataset	67
Figure 24. Segmentation Performance of Different Gaussian Sigma Values on 512x512 Size Images of CU-HCC Dataset	70
Figure 25. Segmentation Performance of Different Histogram Equalization Methods on 128x128 Images of CU-HCC Dataset	71
Figure 26. Segmentation Performance of Different Histogram Equalization Methods on 256x256 Images of CU-HCC Dataset	73
Figure 27. Segmentation Performance of Different Histogram Equalization Methods on 512x512 Images of CU-HCC Dataset	75
Figure 28. Comparison of segmentation performance metrics obtained on the ATLAS Dataset and CU-HCC Dataset at different image sizes	76
Figure 29. Effect of different Gaussian sigma values on segmentation performance on 128x128 images, comparison of the ATLAS Dataset and the CU-HCC Dataset	78
Figure 30. Effect of different Gaussian sigma values on segmentation performance on 256x256 images, comparison of the ATLAS Dataset and the CU-HCC Dataset	80
Figure 31. Effect of different Gaussian sigma values on segmentation performance on 512x512 images, comparison of the ATLAS Dataset and the CU-HCC Dataset	82
Figure 32. Effect of different histogram equalization methods on segmentation performance on 128x128 images, comparison of the ATLAS Dataset with the images used in CU-HCC Dataset	84
Figure 33. Effect of different histogram equalization methods on segmentation performance on 256x256 images, comparison of ATLAS Dataset with the images used in CU-HCC Dataset	86
Figure 34. Effect of different histogram equalization methods on segmentation performance on 512x512 images, comparison of ATLAS Dataset with the images used in CU-HCC Dataset	87
Figure 35. Graph of DICE Score Average Values in Different Dimensions and Filters	91
Figure 36. Graph of IoU Average Values in Different Dimensions and Filters	94
Figure 37. Pixel Accuracy Average Values Graph for Different Sizes and Filters	97
Figure 38. Graph of Recall Values According to Gaussian Filter and Image Size Levels .	99

Figure 39. F-Score Average Values Graph in Different Dimensions and Filters	102
Figure 40. Graph of Precision Average Values in Different Size and Filters.....	104
Figure 41. Graph of Specificity Average Values in Different Dimensions and Filters	107



LIST OF TABLES

Table 1. Architecture Summary of the U-Net-Based Deep Learning Model for Medical Image Segmentation.....	32
Table 2. Segmentation performance of different sized images of ATLAS Dataset.....	51
Table 3. Segmentation Performance of Different Gaussian Sigma Values on 128x128 ATLAS Dataset Images	53
Table 4. Segmentation Performance of Different Gaussian Sigma Values on 256x256 ATLAS Dataset Images	54
Table 5. Segmentation Performance of Different Gaussian Sigma Values on 512x512 ATLAS Dataset Images	56
Table 6. Segmentation Performance of Different Histogram Equalization Methods on 128x128 ATLAS Dataset Images	58
Table 7. Segmentation Performance of Different Histogram Equalization Methods on 256x256 ATLAS Dataset Images	60
Table 8. Segmentation Performance of Different Histogram Equalization Methods on 512x512 ATLAS Dataset Images	61
Table 9. Segmentation performance of images of different sizes in CU-HCC Dataset.....	63
Table 10. Segmentation Performance of Different Gaussian Sigma Values on 128x128 Size Images of CU-HCC Dataset.....	65
Table 11. Segmentation Performance of Different Gaussian Sigma Values on 256x256 Size Images of CU-HCC Dataset.....	67
Table 12. Segmentation Performance of Different Gaussian Sigma Values on 512x512 Size Images of CU-HCC Dataset.....	69
Table 13. Segmentation Performance of Different Histogram Equalization Methods on 128x128 Images of CU-HCC Dataset.....	71
Table 14. Segmentation Performance of Different Histogram Equalization Methods on 256x256 Images of CU-HCC Dataset.....	73
Table 15. Segmentation Performance of Different Histogram Equalization Methods on 512x512 Images of CU-HCC Dataset.....	74
Table 16. Averages of Performance Metrics by Size and Filter	88
Table 17. Repeated Measures ANOVA Results: F, p, and η^2 Values for Size, Filter, and Size×Filter Interaction.....	90

LIST OF ABBREVIATIONS

AI	: Artificial Intelligence
AUC–ROC	: Area Under the Receiver Operating Characteristic Curve
ASD	: Average Symmetric Surface Distance
CAD	: Computer-Aided Diagnosis
CCA	: Cholangiocarcinoma
CDF	: Cumulative Distribution Function
CE-MRI	: Contrast-Enhanced Magnetic Resonance Imaging
CHAOS	: Combined (CT-MR) Healthy Abdominal Organ Segmentation
CLAHE	: Contrast Limited Adaptive Histogram Equalization
CNN	: Convolutional Neural Network
CT	: Computed Tomography
CU-HCC Dataset	: Cukurova University HCC Dataset
DCNN	: Deep Convolutional Neural Network
DICE	: Dice Similarity Coefficient
DICOM	: Digital Imaging and Communications in Medicine
DL	: Deep Learning
DL HASTE-FS	: Deep Learning-Accelerated Half-Fourier Acquisition Single-shot Turbo Spin-Echo with Fat Suppression
DNN	: Deep Neural Network
DSC	: Dice Similarity Coefficient
FCN	: Fully Convolutional Network
FN	: False Negative
FP	: False Positive
GMM	: Gaussian Mixture Model
HCC	: Hepatocellular Carcinoma
HE	: Histogram Equalization
IOU	: Intersection over Union
JDRD	: Japanese Society of Radiological Technology Database
LI-RADS	: Liver Imaging Reporting and Data System
ML	: Machine Learning
MRI	: Magnetic Resonance Imaging

PA	: Pixel Accuracy
PDF	: Probability Density Function
PIXEL_ACU	: Pixel Accuracy
PLU	: Parametric Linear Units
REC	: Recall
RVD	: Relative Volume Difference
SEN	: Sensitivity
SPE	: Specificity
TP	: True Positive
TN	: True Negative
TUBITAK	: The Scientific and Technological Research Council of Turkey
TARE	: Transarterial Radioembolization
U-Net	: U-Net (Encoder–Decoder Architecture)
VOE	: Volumetric Overlap Error
3DIRCADb	: 3D Image Reconstruction for Comparison of Algorithm Database

1. INTRODUCTION

The liver is a complex organ composed of various cell types, including hepatocytes, bile duct epithelial cells, immune cells, and liver sinusoidal endothelial cells. These cells are specialized to perform distinct physiological functions. The smallest structural and functional unit of the liver is the lobule, where liver cells are organized around a central venous vessel. Hepatocytes are radially arranged around this central vein (Trefts E., 2017). The liver is the largest internal organ in the human body and is responsible for numerous vital functions. It is located in the upper right quadrant of the abdominal cavity, just beneath the diaphragm. On average liver weight is approximately 1400–1800 grams in males and 1200–1400 grams in females (Taşdoğan BE., 2021).



Figure 1. Anatomical location of the liver in the human body (Bhan I., 2020).

The liver plays a central role in the synthesis and metabolism of various substances. Its primary functions can be summarized as follows:

- Synthesis of proteins and lipids (including cholesterol), and lipid storage
- Storage and release of carbohydrates (particularly glucose)
- Bile production
- Urea synthesis

- Detoxification and elimination of various drugs and exogenous substances
- Metabolism of certain hormones (Taşdoğan BE., 2021).

Due to its critical physiological functions, the liver is regarded a vital organ. Consequently, any condition that impairs liver function such as tumors, infections, or other pathological processes can pose a direct threat to human life. Various types of liver tumors exist, depending on the cell type from which they originate. These include hepatic adenoma, hepatoblastoma, focal nodular hyperplasia, dysplastic nodules, cholangiocarcinoma, and HCC. In addition, other less common tumor types may also occur (Ergüney S., 2019).

HCC is the most common primary malignant tumor of the liver, accounting for approximately 90% of all primary liver cancers and representing the fourth leading cause of cancer-related deaths worldwide (Villanueva, 2019). This condition is responsible for an estimated 745,000 deaths annually on a global scale (Bray et al., 2018). Projections suggest that the global mortality burden due to HCC may reach one million deaths per year by 2030 (Anstee et al., 2019). Given its severity, HCC must be accurately and promptly assessed by clinicians to ensure timely initiation of the most appropriate treatment strategy. The diagnosis of HCC relies primarily on imaging modalities, including ultrasonography (USG), computed tomography (CT), and MRI (Alawyia & Constantinou, 2023).

Imaging plays a crucial role in the diagnosis of HCC. Unlike most solid tumors, HCC can often be diagnosed based solely on imaging techniques without the need for biopsy confirmation, allowing patients to be directed to appropriate treatment based on imaging results alone. Even when biopsy is required, imaging guidance is typically necessary for accurate sampling. Serum biomarkers such as alpha-fetoprotein generally play a limited role in the diagnosis of HCC. For high-risk patients with clinical suspicion of HCC, CT or MRI is recommended as the initial diagnostic modality. The American College of Radiology has issued guidelines on how CT and MRI should be performed and reported through the CT/MRI Liver Imaging Reporting and Data System (LI-RADS) (Tang et al., 2018).

LI-RADS categorizes lesions or pseudolesions identified on imaging that are greater than 10 mm in diameter based on the likelihood of representing benign entities, HCC, or other hepatic malignant neoplasms (e.g., cholangiocarcinoma [CCA] or combined HCC-CCA).

- **LI-RADS 1:** Definitely benign; this category typically includes cysts and classic hemangiomas.
- **LI-RADS 2:** Probably benign; includes atypical hemangiomas and focal parenchymal abnormalities that may be associated with underlying cirrhosis.
- **LI-RADS 3:** Indicates a low probability of HCC.
- **LI-RADS 4:** Indicates a high probability of HCC.
- **LI-RADS 5:** Represents definite HCC (Marrero et al., 2018).

With recent technological advancements, MRI now provides increasingly higher-resolution images, further enhancing its role in the diagnosis of HCC. MRI is a non-ionizing imaging modality that produces cross-sectional images using a strong magnetic field and radiofrequency waves. When a patient is placed within the static magnetic field, protons in the body align either parallel or antiparallel to the direction of the magnet and begin to exhibit spin motion. Radiofrequency pulses are then applied, causing a temporary displacement in the alignment of hydrogen atoms within tissues. When these pulses are turned off, the protons return to their original alignment, releasing energy in the process. This released energy is captured by a receiver and converted into a signal. Since different tissues exhibit varying degrees of displacement and relaxation times, these signal differences are translated into the contrast seen in MRI (Herek & Karabulut, 2010).

In MRI, two primary sequences, T1-weighted and T2-weighted are commonly employed. T1-weighted sequences provide high contrast between soft tissues and superior spatial resolution, enabling detailed assessment of anatomical structures. In contrast, T2-weighted sequences are particularly effective in detecting pathological changes by highlighting differences in signal intensity. In both sequences, the signal intensity characteristics of lesions vary depending on their internal composition, such as the presence of blood, fat, or fluid (Herek & Karabulut, 2010).

In T1-weighted imaging, short repetition times (< 700 ms) and short echo times ($TE < 30$ ms) are preferred. These parameters create contrast based on differences in longitudinal relaxation times, allowing for high-resolution visualization of anatomical details. Fat tissue typically

appears hyperintense (bright) on T1-weighted images. Following contrast agent administration, T1-weighted sequences are particularly valuable for assessing vascularity and enhancement patterns. Subacute hemorrhages and contrast-enhancing malignant lesions are also better visualized with this sequence (Low RN., 2007).

In T2-weighted imaging, both repetition time (> 2000 ms) and echo time ($> 70\text{--}80$ ms) are prolonged. These sequences generate contrast based on differences in transverse relaxation times. On T2-weighted images, fluid-containing structures, such as edema, typically appear hyperintense, whereas some solid tissues may exhibit reduced signal intensity. Due to these characteristics, T2-weighted sequences are clinically valuable for detecting inflammation, edema, cystic structures, and malignant lesions, offering high sensitivity for tissue characterization. In hypervascular tumors such as HCC, T2 hyperintensity is nonspecific but gains diagnostic value when interpreted in conjunction with other sequences. While T1-weighted imaging highlights anatomical detail, T2-weighted sequences are more sensitive to edema, necrosis, and fluid-containing components. The appropriate combination of these sequences provides high diagnostic accuracy in the detection, staging, and treatment response monitoring of malignant lesions such as HCC (Low RN., 2007).

Despite advancements in imaging technologies, the early detection of HCC remains a significant clinical challenge, as the disease is often asymptomatic in its initial stages (Perez & Grande, 2020). Consequently, research has increasingly focused on strategies for early diagnosis. In conventional MRI-based approaches, liver and tumor segmentation is typically performed manually by evaluating the three-dimensional image in sequential slices. In computer-aided diagnosis (CAD) systems, manual segmentation techniques are commonly employed. However, these methods rely heavily on visual interpretation and the radiologist's level of expertise, and are often time-consuming. Liver segmentation provides critical anatomical information that supports the planning of interventions such as tumor resection and minimally invasive procedures (Ratke et al., 2002; Vorontsov et al., 2019).

Beyond visual assessment of radiological images by the human eye, numerous studies have focused on the development of computer-based systems aimed at identifying the three-dimensional structures of tumors. Early versions of such systems relied on rule-based algorithms, which interpreted image-derived data according to predefined rules.

Subsequently, machine learning (ML) techniques were introduced, enabling significant advances in image recognition and classification tasks. By the late 20th century, especially with the rise of supervised learning methods, ML became widely applied across various domains. These algorithms learned from data by extracting predefined features. Building upon these developments, deep learning (DL) methods emerged, capable of modeling more complex structures and achieving improved classification accuracy (LeCun et al., 2015). In particular, DL architectures such as convolutional neural networks (CNNs) have enabled automated feature extraction from images, facilitating highly accurate tumor detection and segmentation (Gul et al., 2022).

CNNs utilize medical images as input during the learning process and perform learning through a series of hierarchical layers within the framework of DL. As a result, CNNs are widely applied for detection, classification, and segmentation tasks in radiological imaging (Vorontsov et al., 2019; Drozdal et al., 2018). Among CNN architectures, U-Net has been shown to yield effective results in medical image segmentation tasks (Isensee et al., 2018). Several studies have reported successful applications of DL and CNN models for liver tumor segmentation. Wang et al. (2019) demonstrated that CNNs could be effectively used for the automatic segmentation of the liver. In another study, Bousabarah et al. (2021) showed that both the liver and HCC could be automatically segmented using deep convolutional neural networks (DCNNs) (Bousabarah et al., 2021).

Following the study by Ronneberger et al. (2015) on the implementation of the U-Net architecture (encoder–decoder structure), the use of deep learning techniques for the segmentation of radiological images has gained significant popularity (Ronneberger et al., 2015). This advancement has contributed to the widespread adoption of AI-based CAD systems, further promoting the use of deep learning algorithms supported by high-performance computing and advanced hardware resources.

AI-based CAD systems utilize pretrained models, whereas manual CAD systems lack such modeling capabilities. During the training phase, AI models are trained on datasets containing liver and tumor regions, allowing them to learn relevant features such as shape, tissue structure, and intensity values of both liver and tumor areas. Once the training process is completed, AI-based CAD systems can provide rapid and efficient information regarding the

structural characteristics of tumors without the need for human intervention (Nishida et al., 2019). With the growing global population, the burden on healthcare systems is also increasing. In diseases such as HCC, where timely diagnosis and treatment are of critical importance, AI-based applications have the potential to reduce physicians' workload and facilitate faster diagnostic decision-making.

In the current literature, the majority of studies on automatic segmentation of HCC lesions are based on CT images. One of the primary reasons for this is the long-standing and widespread clinical use of CT, as well as the greater availability of multicenter CT datasets. Additionally, CT images are typically acquired using standardized protocols and exhibit less variability in imaging characteristics, making them more amenable to standardization for AI algorithms. As a result, CT has offered advantages in early-stage studies for training and validating segmentation models. However, MRI provides a far superior imaging modality for HCC diagnosis, particularly due to its enhanced soft tissue contrast and ability to convey functional information. However, due to the variability of MRI sequences, differences in acquisition parameters across institutions, and limited data availability, MRI has not yet been adopted as widely as CT in automatic segmentation studies. Despite MRI's potential to yield higher diagnostic accuracy, these challenges have contributed to the relatively limited volume of MRI-based datasets in the literature. Therefore, studies focusing on the automatic segmentation of HCC from MRI address a critical gap in the field and hold significant potential to contribute to future research and clinical advancements.

In this thesis study, MR images were used for the automatic segmentation of HCC. The CU-HCC Dataset was composed of MRI data (T2 sequence) obtained from the Department of Interventional Radiology at Çukurova University Faculty of Medicine Hospital and the publicly available ATLAS dataset (T1 sequence).

The primary aim of this thesis is to systematically evaluate the performance of deep learning-based automatic segmentation methods applied to MRI for the diagnosis of HCC, in relation to different datasets, resolution levels, and preprocessing techniques. Unlike similar studies in the literature, this thesis applies the developed deep learning-based segmentation approaches to both T2-weighted MR images (CU-HCC Dataset) and the T1-weighted ATLAS dataset. In doing so, the impact of dataset-specific structural differences such as tumor annotation styles,

imaging phases, and contrast parameters on segmentation accuracy is examined through a comparative analysis.

Within the scope of this thesis, MR image scans were also rescaled to different dimensions (128×128, 256×256, 512×512), and the segmentation performance was evaluated under conditions with and without the application of preprocessing techniques such as Gaussian filtering (CLAHE) and histogram equalization (HE). The performance metrics obtained in the study (DICE, IoU, Recall, Specificity, Sensitivity) were evaluated not only descriptively but also statistically across groups. This allowed for a scientific assessment of the effects of filter type, sigma value, resolution, and sequence type on segmentation performance.

This thesis aims not only to develop models with high algorithmic accuracy, but also to provide insight into the determining role of different data types, resolutions, and preprocessing steps on model performance. One of the key contributions of this study to the literature is that the evaluation of segmentation performance is not limited to descriptive statistics; rather, these metrics are subjected to comparative analysis using statistical testing methods across different groups. In many studies in the literature, performance metrics such as Dice, IoU, and Recall are typically reported using mean and standard deviation values, with limited emphasis on statistically assessing the significance of differences between groups. In contrast, this thesis employs statistical testing to evaluate performance outcomes across different image sizes, filtering techniques, and preprocessing steps, thereby providing a scientific basis for determining the significance of the observed differences. In this regard, our study addresses methodological gaps that are often overlooked but critically important in the medical image segmentation literature. It provides original and systematic contributions in areas such as data source comparison, parametric optimization, and statistical difference analysis. Supporting segmentation performance not only with high metric scores but also with statistically significant differences enhances the reliability and clinical applicability of the developed models. This approach represents a significant methodological distinction that sets our study apart from similar works. The study aim to enhance HCC segmentation performance through deep learning-based methods, optimize model accuracy in this domain, and provide original, multidimensional contributions to areas often underrepresented in the medical image segmentation literature such as data source comparison, parametric preprocessing strategies, and statistical difference analysis.



2. LITERATURE REVIEW

Advancements in medical imaging technologies have become increasingly important for the early diagnosis and monitoring of liver diseases. In this context, the accurate detection and segmentation of common and life-threatening liver tumors such as HCC play a critical role in clinical decision support systems. In the literature, three main approaches are commonly used for liver and tumor segmentation: manual, semi-automatic, and fully automatic methods. Manual segmentation relies heavily on human expertise and is a time-consuming process. As a result, with the integration of computer technologies into medical research, its practical use has become increasingly rare. Semi-automatic segmentation is performed by a human operator with the assistance of computer-aided algorithms. While this approach can save time, its effectiveness may vary depending on the computational power and performance of the system. Fully automatic segmentation, on the other hand, has the potential to significantly reduce the workload of medical personnel and is therefore regarded as the most ideal solution by researchers. In this context, fully automated segmentation systems offer the dual advantage of saving time and minimizing human error. As a result, the number of studies focusing on the automatic segmentation of HCC has been steadily increasing (Tran et al., 2021).

Although automatic segmentation yields promising results with current image processing techniques, it still presents several limitations, particularly in structurally complex organs like the liver and in heterogeneous structures such as tumors. Segmentation accuracy can be adversely affected by factors such as variability in tumor size, shape, and location. At this point, deep learning techniques hold significant potential for overcoming these challenges. In particular, deep learning-based algorithms have demonstrated greater success in detecting small and visually inconspicuous tumors. When provided with sufficient data and computational power, these methods enable not only the accurate segmentation of tumors that may be imperceptible to the human eye, but also of anatomically complex and irregularly shaped organs (Campadelli et al., 2009).

With the advantages offered by deep learning techniques in the field of medical imaging, numerous methods have been developed for the automatic segmentation of the liver and its lesions. Among these methods are fully convolutional networks (FCNs) (Pang et al., 2019; Zheng et al., 2019; Sun et al., 2017), fast fuzzy c-means and adaptive watershed algorithms

(Wu et al., 2017; Anter & Hassenian, 2019), encoder-decoder architectures (Jiang et al., 2019; Seo et al., 2020), and the SegNet architecture (Almotairi et al., 2020). Among encoder-decoder architectures, the U-Net model (Ronneberger et al., 2015) has been widely adopted in the field of medical image segmentation. In addition to liver and tumor segmentation, U-Net has demonstrated successful results in various anatomical regions, including the pancreas, colorectal tumors, prostate, and brain tumors (Tran et al., 2021).

In recent years, various deep learning-based studies have been conducted to automate liver segmentation. One notable example is the study by Araújo et al. (2021), which aimed to automate liver segmentation by combining CNNs with image processing techniques. The results demonstrated that deep learning approaches can significantly improve segmentation accuracy. The reported performance metrics in the study included an average sensitivity of 95.45%, specificity of 99.86%, Dice coefficient of 95.64%, volumetric overlap error of 8.28%, relative volume difference of -0.41%, and Hausdorff distance of 26.60 mm. These results indicate that liver segmentation can be performed with high accuracy and efficiency using deep learning techniques (Araújo et al., 2021).

A recent study conducted by Wang et al. (2023) proposed a deep learning-based method for the automatic segmentation of liver tissue and tumors using MRI. The study was carried out on 105 patients diagnosed with hepatocellular carcinoma. The results demonstrated high accuracy for liver segmentation, with Dice Similarity Coefficient (DSC) values ranging from 0.91 to 0.92, and a substantial reduction in segmentation time (manual: 182.9 seconds; automatic: 2.2 seconds). For liver tumors, the average DSC ranged from 0.612 to 0.687. These results highlight the capacity of deep learning to enable fast and accurate segmentation of both liver tissue and tumors on MRI, while also offering time efficiency for clinical applications and a robust foundation for advanced analyses (Wang et al., 2023).

Another important study was conducted by Hossain et al. (2023), in which T1-weighted MRI scans from 20 patients in the Combined (CT-MR) Healthy Abdominal Organ Segmentation (CHAOS) dataset were used to evaluate liver segmentation across 24 segmentation networks of varying depths. T1-weighted images are particularly useful for segmentation due to their ability to clearly depict fat content. The researchers proposed a novel cascaded network for segmenting axial liver slices, which outperformed existing segmentation approaches in the

literature. The proposed model achieved a Dice Similarity Coefficient of 95.15% and an IoU of 92.10%, demonstrating its superior effectiveness compared to previous methods (Hossain et al., 2023).

A study by Skwirczyński et al. (2023) explored a multi-class segmentation approach for differentiating liver lesions and normal tissue on multiparametric MR images. The segmentation was performed using the nnU-Net architecture, a deep learning model based on convolutional neural networks. Unlike traditional methods that rely on manual feature extraction, the model autonomously learned tissue and lesion characteristics directly from the imaging data. The results indicated strong performance, with the Area Under the Receiver Operating Characteristic Curve (AUC ROC) reaching 0.925 on the training set and 0.852 on the test set. The researchers emphasized that such machine learning-based approaches are effective in detecting hepatic lesions and accurately classifying tumors. Moreover, they highlighted the potential of these methods to contribute to the standardization of medical evaluation processes for liver lesions (Skwirczyński et al., 2023).

In a retrospective analysis, Vorontsov et al. (2019) employed a deep learning-based fully convolutional network (FCN) for the automatic detection and segmentation of liver lesions on contrast-enhanced CT images. Manual segmentation was used as the reference standard, and the accuracy of both automatic and user-corrected segmentations was compared to the manual annotations. The sensitivity of automatic lesion detection varied according to lesion size: 10% for lesions smaller than 10 mm, 71% for those between 10–20 mm, and 85% for lesions larger than 20 mm. The Dice Similarity Coefficient was reported as 0.14, 0.53, and 0.68 for the respective size groups. The researchers noted that segmentation efficiency could be improved through targeted corrections to the input images, and concluded that FCNs show promising potential for liver tumor segmentation (Vorontsov et al., 2019).

Li et al. (2013) conducted an analysis on 30 enhanced CT volumes to compare the performance of machine learning methods and CNN-based models for liver CT scan segmentation. The researchers proposed a fully automated segmentation approach that combined the fuzzy c-means clustering algorithm with the level set method. In the initial step, contrast enhancement was applied to the images to sharpen boundary delineation. Subsequently, spatial fuzzy c-means clustering, supported by anatomical information, was

employed for the automatic segmentation of the liver region. To further improve segmentation accuracy, a distance-based refinement strategy using the level set method was introduced. In the final stage, morphological operations were applied to complete the segmentation process. The results demonstrated high accuracy (99.86%) and specificity (99.89%). Moreover, this method was shown to address the over-segmentation problem more effectively than the standard level set approach. Performance analysis revealed a Dice Similarity Index of $80.06\% \pm 1.63$, sensitivity of $82.67\% \pm 1.43$, and recall of $84.34\% \pm 1.61$. The study concluded that CNN-based methods outperform traditional machine learning algorithms in liver segmentation tasks (Li et al., 2013).

Das et al. (2018) developed a novel method capable of automatically detecting liver cancer. This approach is based on a Watershed-Gaussian Deep Learning (WGDL) technique and was tested on a total of 225 CT images. In the initial phase, the liver region was isolated using watershed segmentation, followed by lesion segmentation through a Gaussian Mixture Model (GMM). After segmentation, the extracted texture features from the lesions were passed to a deep neural network (DNN) classifier, which successfully distinguished among three types of liver cancer: hemangioma, hepatocellular carcinoma, and metastatic carcinoma. The method demonstrated highly successful results, achieving 99.38% accuracy, a Jaccard Index of 98.18%, and a low validation loss of 0.062. The researchers suggested that this model, when tested on larger datasets, could serve as a supportive tool for radiologists in clinical applications (Das et al., 2018).

Sun et al. (2017) developed a segmentation method for liver tumors based on a fully convolutional network (FCN) approach. To process multiphase contrast-enhanced CT images, the researchers designed a Multi-Channel Fully Convolutional Network (MC-FCN) model. Since each phase of contrast-enhanced imaging provides different pathological characteristics, separate networks were trained for each phase, and high-level features from these networks were subsequently fused. The model was evaluated using the 3D Image Reconstruction for Comparison of Algorithm Database (3Dircadb) and the Japanese Society of Radiological Technology Database (JDRD). In the 3Dircadb dataset, segmentation performance was assessed using metrics such as volumetric overlap error (VOE), relative volume difference (RVD), and surface distance, yielding results within acceptable error ranges. Tests on the JDRD dataset further demonstrated that the MC-FCN model achieved lower error rates and

higher accuracy compared to previous methods. The study concluded that the MC-FCN approach offers superior accuracy and reliability and may serve as an effective solution for liver tumor segmentation (Sun et al., 2017).

Ben-Cohen et al. (2016) investigated the use of fully convolutional networks (FCNs) for liver segmentation and the detection of liver metastases, highlighting the strength of FCNs as a tool for semantic segmentation. The study evaluated the performance of FCNs on a small dataset and compared the results with those of patch-based CNNs and sparse classification approaches. Three-fold cross-validation tests demonstrated that FCNs outperformed the alternative methods. Notably, the fully automated FCN algorithm achieved a positive detection accuracy of 86% with an average of only 0.6 false positives per case. These results suggest that FCNs offer an effective solution for both liver segmentation and metastasis detection (Ben-Cohen et al., 2016).

Vivanti et al. (2017) developed a method for the automatic detection of newly developed tumors and tumor burden estimation in the long-term follow-up of liver tumors. The approach integrates the boundaries and appearance features of existing tumors from baseline and follow-up CT scans using a predefined model. This methodology enabled the automatic identification of new tumors as well as the calculation of total tumor burden. Tests conducted on 246 tumors showed that the method successfully detected new tumors with an accuracy of 86% and achieved a tumor burden estimation error of 16%. These results suggest that the proposed method may serve as an effective tool for monitoring liver tumors over time (Vivanti et al., 2017).

Yasaka et al. (2018) evaluated the use of deep learning methods and convolutional neural networks for the differential diagnosis of liver masses using dynamic contrast-enhanced CT. In the study, liver lesions were classified into five categories: Category A (HCC), Category B (malignant liver tumors other than typical and early-stage HCC), Category C (indeterminate masses or mass-like lesions), Category D (hemangiomas), and Category E (cysts). The CNN model consisted of six convolutional layers, three max-pooling layers, and three fully connected layers. The results showed a median diagnostic accuracy of 0.84 for liver mass classification and an ROC-AUC of 0.92 for differentiating Categories A–B from Categories C–E. These results demonstrate that deep learning and CNN-based methods can be effectively

applied to the differential diagnosis of liver masses in dynamic contrast-enhanced CT imaging (Yasaka et al., 2018).

Fan et al. (2020) developed a deep learning-based segmentation algorithm to improve the accuracy of needle placement procedures guided by CT in the treatment of liver tumors. The proposed segmentation method integrated multi-scale input and output features to optimize contextual information. The results demonstrated high accuracy, with a Dice Similarity Coefficient of 96.1%. The segmentation outputs facilitated the integration of interventional CT images with diagnostic scans, enabling clearer visualization of tumors and enhancing the overall efficiency of the procedural workflow (Fan et al., 2020).

Fallahpoor et al. (2024) addressed the segmentation of the liver and hepatic lesions using deep learning techniques, with applications in nuclear medicine, image-guided interventions, and large-scale image processing. In this study, T1-weighted (T1w) and T2-weighted (T2w) MR images were collected from 128 patients, and manual annotations of the liver and lesions were performed. The deep learning model was trained on data derived from both imaging modalities. The test results showed a Dice Similarity Coefficient of 88% for liver segmentation and 53% for tumor segmentation. The researchers concluded that while liver segmentation could be achieved with high accuracy, the performance in lesion segmentation remained suboptimal (Fallahpoor et al., 2024).

In a recent study, Dai et al. (2025) evaluated the performance of a deep learning-based software in the automatic detection and measurement of liver lesions using MRI data. During the segmentation phase, the AI model successfully delineated lesions with an average Dice Similarity Coefficient of 0.62. AI-assisted detections demonstrated higher accuracy compared to those performed solely by radiologists. When AI and radiologists worked in combination, the detection rate increased to 0.894 and sensitivity to 0.883, whereas radiologist-only detection yielded a rate of 0.825 and sensitivity of 0.806. Notably, for lesions smaller than 20 mm, AI outperformed radiologists in sensitivity (0.848 vs. 0.788). For lesions ≥ 20 mm, both methods showed comparable performance, with Dice scores of 0.867 for AI and 0.881 for radiologists. These results suggest that deep learning-based software can make a significant contribution to the automatic identification and measurement of liver lesions in clinical practice (Dai et al., 2025).

In line with recent advancements in imaging protocols, Shanbhogue et al. (2021) compared deep learning-accelerated T2-weighted fat-suppressed MR imaging (Deep Learning-Accelerated Half-Fourier Acquisition Single-shot Turbo Spin-Echo with Fat Suppression, DL HASTE-FS) with conventional T2-FS sequences. The DL HASTE-FS method reduced acquisition time by a factor of four, from 178.9 seconds to 51.23 seconds. Moreover, DL HASTE-FS achieved statistically significant superior scores across all evaluated parameters including fat suppression efficiency, image homogeneity, liver edge sharpness, intrahepatic vessel delineation, respiratory motion artifacts, overall artifact level, liver-lesion contrast, and overall image quality (all $p < 0.0001$). Lesion conspicuity and edge sharpness were also significantly better with DL HASTE-FS (all $p < 0.001$). However, no statistically significant difference was observed between the two techniques in terms of liver lesion contrast ($p > 0.05$). These results highlight the clinical utility and diagnostic advantage of accelerated deep learning-based MR imaging techniques for lesion detection (Shanbhogue et al., 2021).

Studies in the literature have demonstrated that deep learning-based automatic methods for liver and HCC segmentation offer significant advantages in terms of both speed and accuracy compared to traditional approaches. However, the majority of research has been conducted using CT data, and the number and scope of MRI-based automatic segmentation studies remain limited. Although existing methods have achieved high performance using various architectures (e.g., FCN, U-Net, DenseUNet, SegNet) and techniques (e.g., fuzzy c-means, watershed, encoder-decoder structures), there is a lack of systematic comparison regarding the impact of dataset sources, phase selection, labeling strategies, image resizing, and filtering operations on segmentation performance. Furthermore, the extent to which model generalizability can be maintained across different imaging modalities and phases has not yet been clearly established.

At this point, our study aims to comprehensively examine the effects of labeling strategies, phase selection, image resizing methods, and filtering techniques on automatic segmentation performance by simultaneously utilizing both an original MRI dataset obtained from Çukurova University Faculty of Medicine Hospital and the publicly available ATLAS dataset. By addressing existing data limitations in the literature, this study contributes novel theoretical and practical insights into MRI-based HCC segmentation.



3. MATERIAL AND METHODS

The methodological steps followed in this study are presented in Figure 2, and the process consists of the following main stages.

- Liver MRI scans, initially in DICOM format, were converted to PNG format. Corresponding segmentation masks were also converted in the same manner.
- Both the images and their masks were resized to predefined resolutions (128×128, 256×256, and 512×512 pixels) to ensure compatibility with the input requirements of the model.
- Several preprocessing techniques were applied to the images, including Gaussian filtering, HE, and CLAHE. These techniques enhance image contrast and contribute to more effective feature learning by the model.
- The preprocessed images and masks were then converted into numpy array format, making them suitable for use in the training phase of the model.
- The segmentation task was performed using a convolutional neural network based on the U-Net architecture. This model features a four-level encoder-decoder structure. The encoder progressively extracts low-dimensional features through max pooling, while the decoder reconstructs the original resolution using transposed convolutions. Skip connections ("crop and concatenate") between corresponding encoder and decoder layers enable the model to retain spatial information lost during downsampling.
- The dataset was divided into training and testing sets. The model was trained using the training images and evaluated on the test set. After training, the model was used to generate segmentation predictions for unseen input images.
- Optional postprocessing techniques, such as morphological operations or filtering, can be applied to the predicted masks to refine the segmentation results.
- As a result of the entire workflow, tumor segmentation masks were successfully generated for the liver MRI scans.

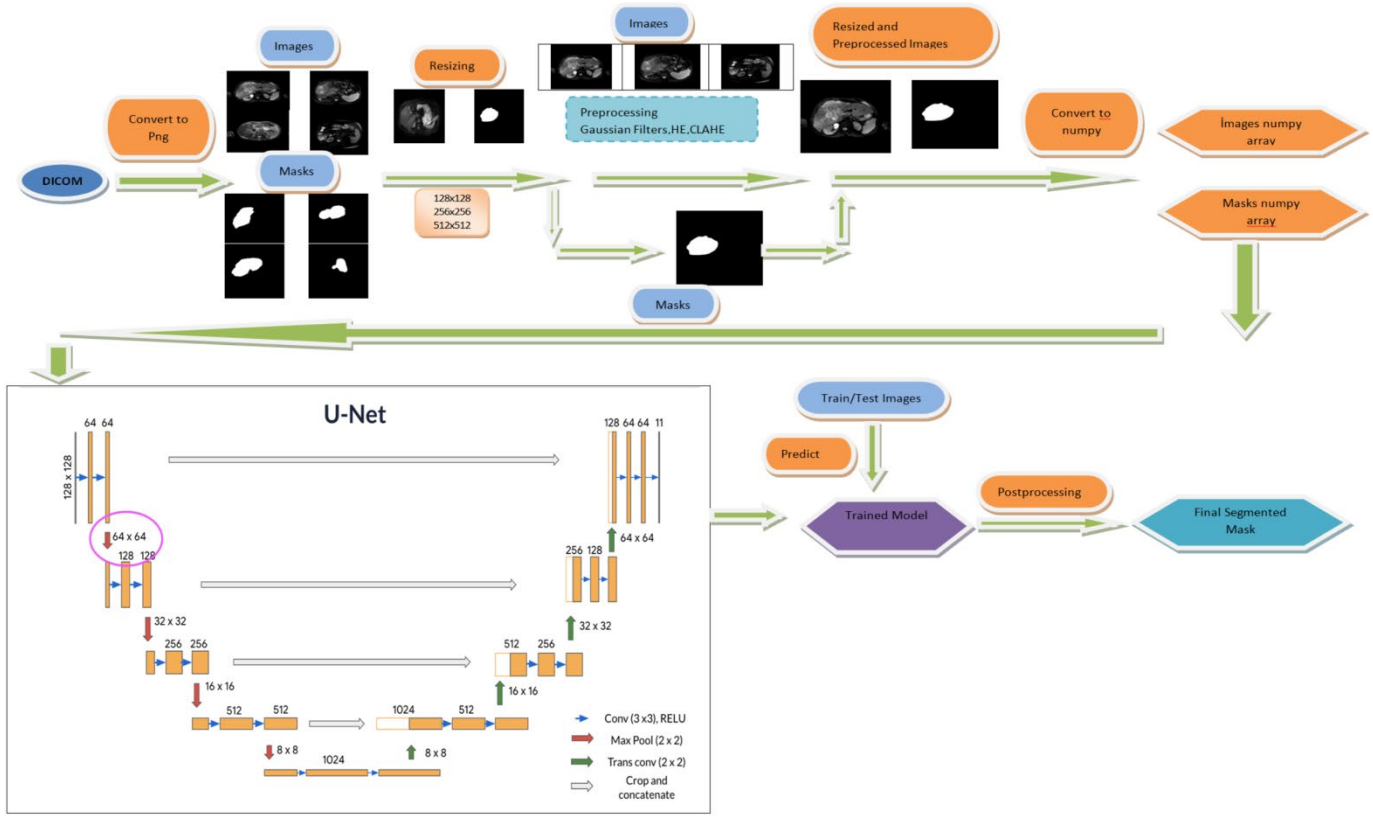


Figure 2. Workflow of the MRI-Based Liver Tumor Segmentation Using U-Net Architecture and Preprocessing Techniques

3.1. Datasets

The magnetic resonance (MR) images used in this study were obtained from two different sources: the publicly available ATLAS dataset (Quinton et al., 2023) and the Information Management System of Çukurova University Faculty of Medicine Hospital. The images from Çukurova University were collected as part of the TÜBİTAK-funded project titled “Prediction of Disease Progression Using Artificial Intelligence Algorithms Based on Radiological, Pathological, and Clinical Data from Patients with Hepatocellular Carcinoma” (Project No: 122S588).

In terms of imaging modalities, the ATLAS dataset includes T1-weighted contrast-enhanced MR images, while the Çukurova University dataset consists of T2-weighted contrast-enhanced MR images. The dataset from Çukurova University is referred to in this study as the

CU-HCC Dataset. All images were reviewed and interpreted by board-certified radiologists with at least ten years of professional experience.

The ATLAS dataset used in this study comprises CE-MRI scans and corresponding segmentation masks for the liver and liver tumors, collected from 90 patients diagnosed with unresectable HCC. The patient cohort included individuals who underwent transarterial radioembolization (TARE) between October 2012 and February 2023. Inclusion criteria for the dataset were as follows: a clinical decision for TARE made by a multidisciplinary tumor board, availability of CE-MRI scans within eight weeks prior to treatment, and the presence of clearly segmentable lesions on T1-weighted CE-MR images. Patients who underwent imaging using alternative modalities, such as T2-weighted MRI or CT, were excluded from the dataset (Quinton et al., 2023).

The acquired MR images were divided into three groups based on resolution: 128×128 , 256×256 , and 512×512 . Segmentation results for each group were recorded separately. For each resolution, segmentation was performed both with and without the application of image filtering techniques. The resulting segmentations were evaluated using performance metrics, including DICE, IoU, Recall, Specificity, and Sensitivity.

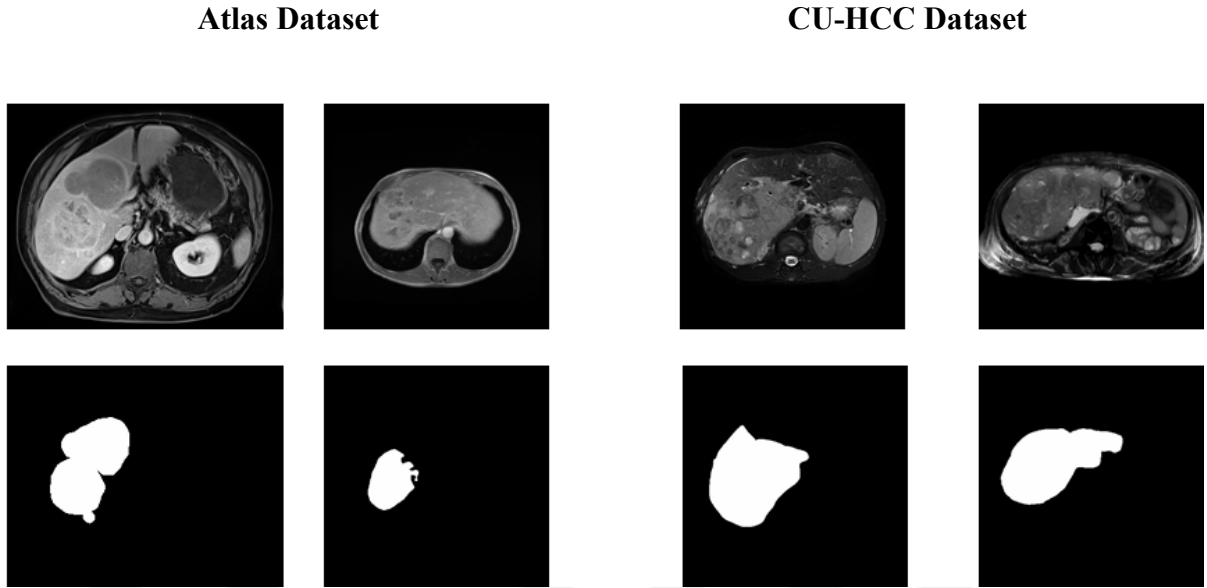


Figure 3. Sample images and masks in the Atlas Dataset and CU-HCC Dataset

3.2. Medical Image Segmentation

In recent years, advances in a wide range of medical imaging technologies have revolutionized the way we visualize functional and pathological processes within the body and identify the anatomical structures in which these processes occur. Imaging modalities such as X-ray, CT, MRI, USG, and nuclear medicine provide two-dimensional (2D) or tomographic three-dimensional (3D) images, offering in vivo insights into the internal structures and functions of the body. These technologies play a critical role in diagnosis, prognosis, treatment planning, and other clinical purposes. Along with the acquisition of such information, proper storage and management of the collected data are also essential.

To ensure compatibility between medical imaging systems and other information systems used in healthcare environments, and to enhance workflow efficiency, the Digital Imaging and Communications in Medicine (DICOM) standard was established. This internationally adopted collaborative standard provides a platform that supports the exchange of biomedical diagnostic and therapeutic information for the communication of digital images and associated data. The DICOM standard includes both a file format specification and a network communication protocol, and is utilized for processing, storing, printing, and transmitting data

in medical imaging. It enables the sharing of images and patient information between two units that support the DICOM format (Özcan et al., 2025).

Segmentation is the process of dividing an image into distinct regions or segments. The primary objective of segmentation is to partition the image into homogeneous and internally consistent regions that correspond to different objects within the scene. This process is performed solely based on the image's intrinsic features. One of the most fundamental features used for segmentation is the image's intensity values. In addition, other features such as edges and texture also play a significant role in the segmentation process. In medical imaging, these regions typically correspond to different tissue classes, organs, or anatomical structures. However, medical image segmentation can be adversely affected by factors such as noise, low contrast, and various imaging artifacts.

- In our study, the data in DICOM format were annotated using the labeling software available in the hospital system.
- The DICOM images were converted into PNG format using Python code, and both the images and their corresponding masks were stored under two separate folders named *image* and *mask*, respectively, with matching filenames.

These PNG-formatted images were then subjected to segmentation using machine learning techniques.

3.3. Segmentation Using Artificial Intelligence Algorithms

AI is a multidisciplinary field of science that aims to enable computer systems to mimic tasks that typically require human intelligence, such as learning, reasoning, problem-solving, pattern recognition, and decision-making (Hamet & Tremblay, 2017). AI systems operate on large-scale datasets and can extract meaningful patterns without human intervention, using these patterns to make predictions. Rather than merely executing predefined rule-based operations, AI also enables the development of dynamic systems that can adapt to new situations based on prior experiences. In this respect, AI has the potential to offer more efficient, faster, and more consistent solutions across a wide range of domains, from healthcare to autonomous vehicles, financial forecasting systems to personal assistants, by minimizing the need for human involvement.

Machine learning, a subfield of AI, enables systems to learn from data without being explicitly programmed. It involves algorithms that improve their performance through experience. Machine learning algorithms create decision models by learning various features for tasks such as classification, regression, and clustering.

Deep learning, a more advanced subfield of machine learning, enables high-level representation learning through multilayered artificial neural networks (Mathew et al., 2021). By hierarchically learning complex patterns in data, deep learning can automatically extract features without the need for manual definition. Owing to this capability, it has achieved remarkable success, particularly in fields such as image processing and natural language processing. Deep learning architectures are widely utilized in medical image analysis, especially for complex tasks such as tumor detection and segmentation.

3.3.1. Convolutional Neural Network

CNN is a type of deep learning architecture composed of multiple layers and is widely used in image processing tasks (Hamet & Tremblay, 2017). It assigns weights and biases to various elements in an image, enabling the differentiation between objects. Additionally, CNNs utilize filters to capture spatial and temporal features. They have shown high performance in a range of applications, including image classification, object detection, and medical image analysis (Alzubaidi et al., 2021).

In recent years, CNN has become a widely used method for achieving successful results in various computer vision applications, including image classification, object detection, facial and vehicle recognition, speech processing, and text analysis. The core principle of CNN is to extract local features from high-dimensional inputs and transfer them to lower layers for learning more complex representations. A typical CNN architecture consists of convolutional, pooling, and fully connected layers (Alzubaidi et al., 2021).

An example of a CNN structure with these layers is illustrated in Figure 4.

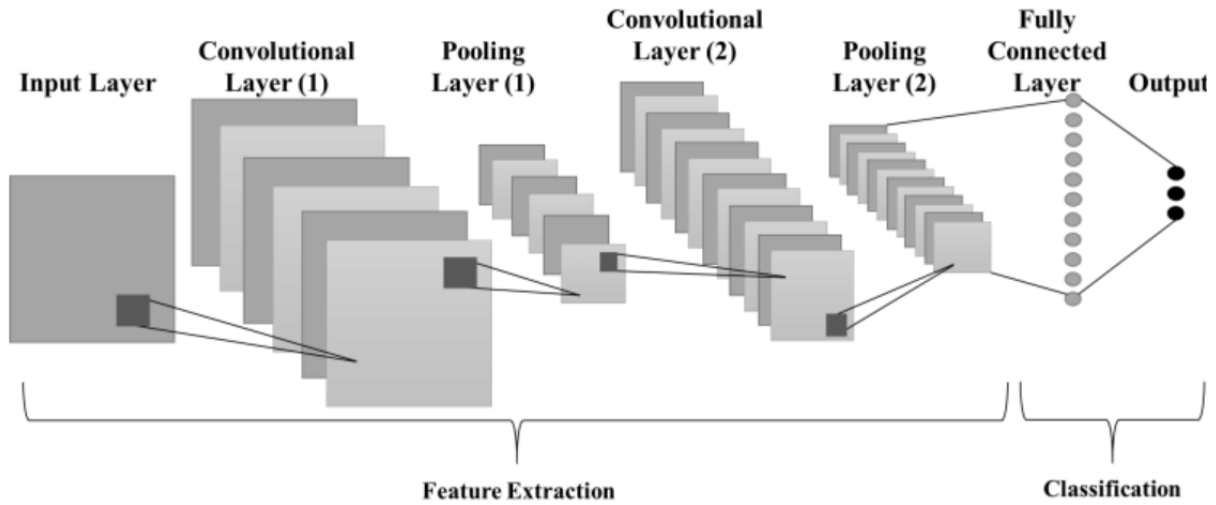


Figure 4. Schematic of Convolutional Neural Network (Samat et al., 2020).

Convolutional Layer

A machine typically interprets an image as a set of numbers organized in matrices, where each number represents the light intensity at a specific point, known as a pixel. The convolutional layer employs a kernel filter to compute the convolution of the input images and extract fundamental features. This kernel (or filter) has a fixed parameter size that is smaller than the input image dimensions.

The filter mask slides across the input image step by step and computes the dot product between the kernel weights and the corresponding pixel values in the input image. This operation produces a 2D activation map.

Through this process, the CNN learns the visual features of the image. The general formulation of the convolution operation in a convolutional layer can be expressed mathematically. Figures 5 and 6 illustrate this computational process and the generation of the activation map in a simplified manner.

$$\text{Activation map} = \text{Input} + \text{Filter}$$

$$= \sum_{y=0}^{\text{columns}} \left(\sum_{x=0}^{\text{rows}} \text{Input}(x-p, y-q) \text{Filters}(x, y) \right) \quad (1)$$

In this formula, $\text{Input}(x-p, y-q)$ refers to a localized region of the input image based on the filter size, where p and q represent the positional offsets. $\text{Filter}(x, y)$ denotes the convolution filter of a specific size. The $\sum$ symbol indicates the summation of element-wise multiplications between the input region and the filter values. The resulting total value at each position forms the activation map as the filter slides over the input image.

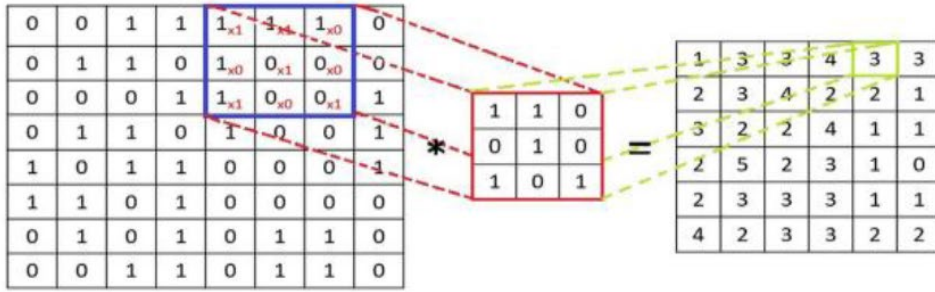


Figure 5. Convolutional Layer (Purwono et al., 2023).

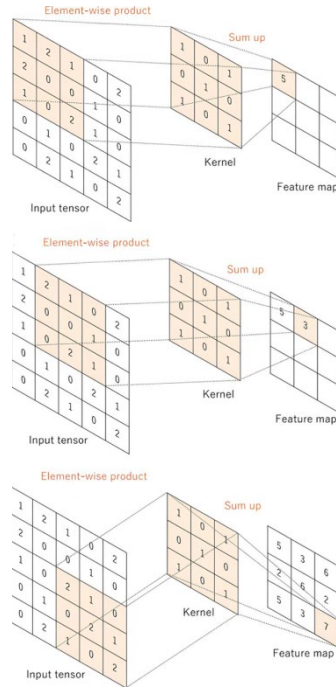


Figure 6. Convolution Operation with a Kernel Size (Yamashita et al., 2018)

A convolutional layer is defined by three key parameters: kernel size, stride length, and padding. The kernel size refers to the dimensions of the kernel filter or the size of the sliding window used during convolution. The stride length determines the distance the kernel moves across the input feature map before computing dot products and generating output pixels. Padding refers to the size of the frame created by adding zeros around the borders of the input feature map (Purwono et al., 2023).

Pooling Layer

The pooling layer connects two consecutive convolutional layers. This process enables the creation of downsampled representations by reducing computational cost and the number of parameters. During pooling, either the maximum or average value can be selected; however, the max pooling method is generally preferred for optimal performance. In addition, the pooling layer is effective in reducing computational load and preventing overfitting. Figure 7 illustrates a simplified example of how the size of an activation map is reduced using a max pooling function (Purwono et al., 2023).

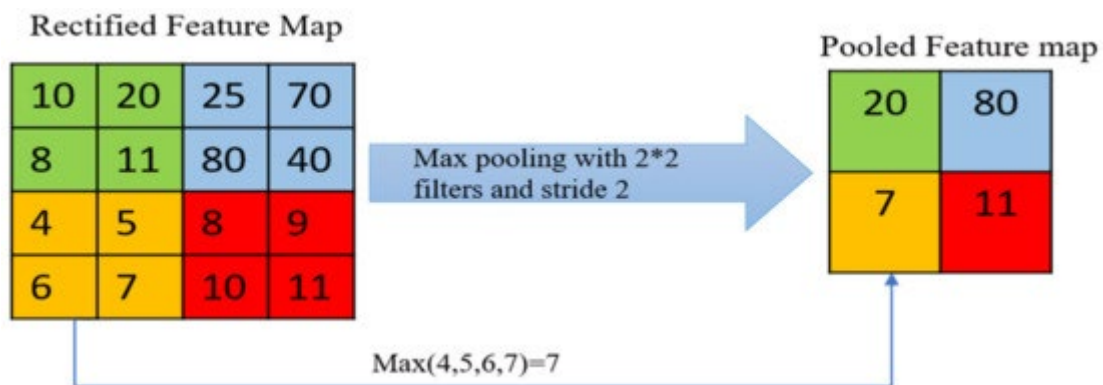


Figure 7. Pooling Layer (Zafar et al., 2022).

Fully Connected Layer

The third layer, commonly known as the fully connected layer or convolutional output layer, is illustrated in Figure 8. This layer resembles a feedforward neural network and is typically located at the bottom of the network. It is fed with input from the final pooling or convolutional output layer, which is flattened before being passed to the next layer. Balanced

distribution of the output refers to transforming all values obtained from the last pooling or convolutional layer into a vector (3D matrix). This method offers a straightforward approach for examining high-level nonlinear combinations of features represented by the convolutional output layer (Purwono et al., 2023).

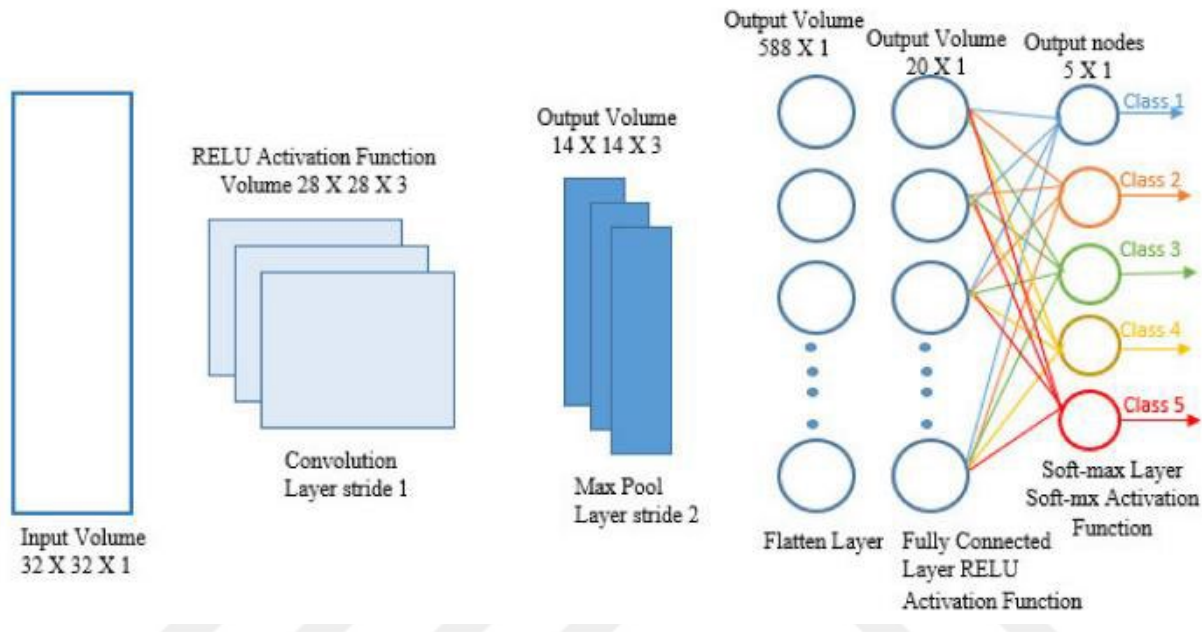


Figure 8. Fully Connected Layer (Purwono et al., 2023).

Nonlinearity Layer (Activation Function)

The activation function plays a crucial role within CNN layers. The filtered output undergoes another mathematical operation known as activation. ReLU, short for "Rectified Linear Unit," is the most commonly used activation function in feature extraction with CNNs. The primary purpose of the activation function is to determine the final output of a neural network for example, a binary decision such as 'yes' or 'no'. This function maps the output values to a specific range, such as between -1 and 1 or 0 and 1 (Bhatt et al., 2021).

Activation functions can be divided into two main categories:

Linear Activation Functions: The mathematical expression for linear activation functions can typically be simplified as $F(x) = CY$. In these functions, input values are multiplied by a constant parameter c , which is considered the weight of each neuron. The resulting output is proportional to the input. Although linear functions are capable of performing more than

simple binary operations (such as 'yes' or 'no'), they are limited in their ability to evaluate multiple outputs.

Non-linear Activation Functions: Non-linear activation functions are widely used in modern neural networks. These functions model the complex relationship between the input and output of the network, allowing for more sophisticated and deep learning processes. They are critical in the development of complex learning and modeling systems, as they enhance the network's ability to learn non-linear structures (Bhatt et al., 2021).

Some of the activation functions commonly used in CNNs and other neural networks are listed below (Alzubaidi et al., 2021):

Sigmoid: This activation function accepts real-valued inputs and maps the output to a range between 0 and 1. The curve of the sigmoid function has an S-shape and is mathematically expressed by the following formula:

$$f(x)_{\text{sigm}} = \frac{1}{1+e^{-x}} \quad (2)$$

In this function, x denotes the input value, typically the weighted sum of inputs in a neural network layer. The constant e represents Euler's number, the base of the natural logarithm. The output $f(x)$ is the transformed value, ranging between 0 and 1, which makes it suitable for representing probabilities in binary classification tasks.

Tanh: The tanh function has a structure similar to the sigmoid function, as both use real numbers as input. However, the tanh function constrains its output between -1 and 1. The tanh function can be mathematically expressed as follows:

$$f(x)_{\text{tanh}} = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (3)$$

x represents the input value. The constant e is Euler's number, the base of the natural logarithm. The output $f(x)$ maps the input to a value between -1 and 1, providing a zero-centered activation that can improve convergence during training in neural networks.

ReLU: ReLU is the most commonly used activation function in CNNs. It transforms all inputs into positive values. Compared to other activation functions, ReLU has a relatively lower computational cost. Mathematically, the ReLU function is represented as follows:

$$f(x)ReLU = \max(0, x) \quad (4)$$

x represents the input value, and $\max$ selects the greater of the two numbers. The function outputs zero for any negative input and returns the input itself if it is zero or positive.

Leaky ReLU: The ReLU function performs the operation of reducing negative inputs to zero, whereas the Leaky ReLU function ensures that inputs are never entirely ignored. This function is used to address the "dying neuron" problem observed in ReLU. The mathematical representation of Leaky ReLU is given as follows:

$$f(x)LeaklyReLU = \begin{cases} x, & \text{if } x > 0 \\ mx, & \text{if } x \leq 0 \end{cases} \quad (5)$$

x is the input value. When x is positive, the output equals x . When x is zero or negative, the output is a small fraction (m) times x , allowing a slight gradient for negative inputs.

Noisy ReLU: This function is used to introduce Gaussian noise. The mathematical expression of the Noisy ReLU function is as follows:

$$f(x)NoisyReLU = \max(0, x + Y), \text{ with } Y \sim N(0, \sigma(x)) \quad (6)$$

x is the input value, Y is a noise term sampled from a normal distribution with mean zero and variance dependent on x and $\max^+$ ensures the output is non-negative by taking the maximum between zero and the noisy input.

Parametric Linear Units (PLU): This function largely adopts the concept of Leaky ReLU. The key difference between the two lies in the leakage factor, which is updated during the training process. The mathematical representation of the PLU function is expressed as follows:

$$f(x)_{ParametricLinear} \begin{cases} x, & \text{if } x > 0 \\ ax, & \text{if } x \leq 0 \end{cases} \quad (7)$$

x is the input, and a is a learnable parameter. For positive x , the output is x , while for zero or negative x , the output is scaled by a , allowing the model to adjust the slope for negative inputs.

3.3.2. U-Net

Ronneberger and colleagues were the first to develop and apply the U-Net architecture for biomedical image segmentation. The U-Net structure is designed based on a convolutional neural network and features a symmetric architecture with two paths on either side. The contraction path (on the left) is known as the encoder, while the expansion path (on the right) is referred to as the decoder. The encoder is used to encode contextual features from the input data, whereas the decoder reconstructs the output using transpose convolutions and upsampling operations to better capture local features. These two paths incorporate convolutional layers, max-pooling operations, ReLU activation functions, concatenation, and upsampling components. The interaction between the encoder and decoder enables high accuracy and detailed results, particularly for segmentation tasks. The U-Net architecture is visually illustrated in Figure 9 (Ronneberger et al., 2015).

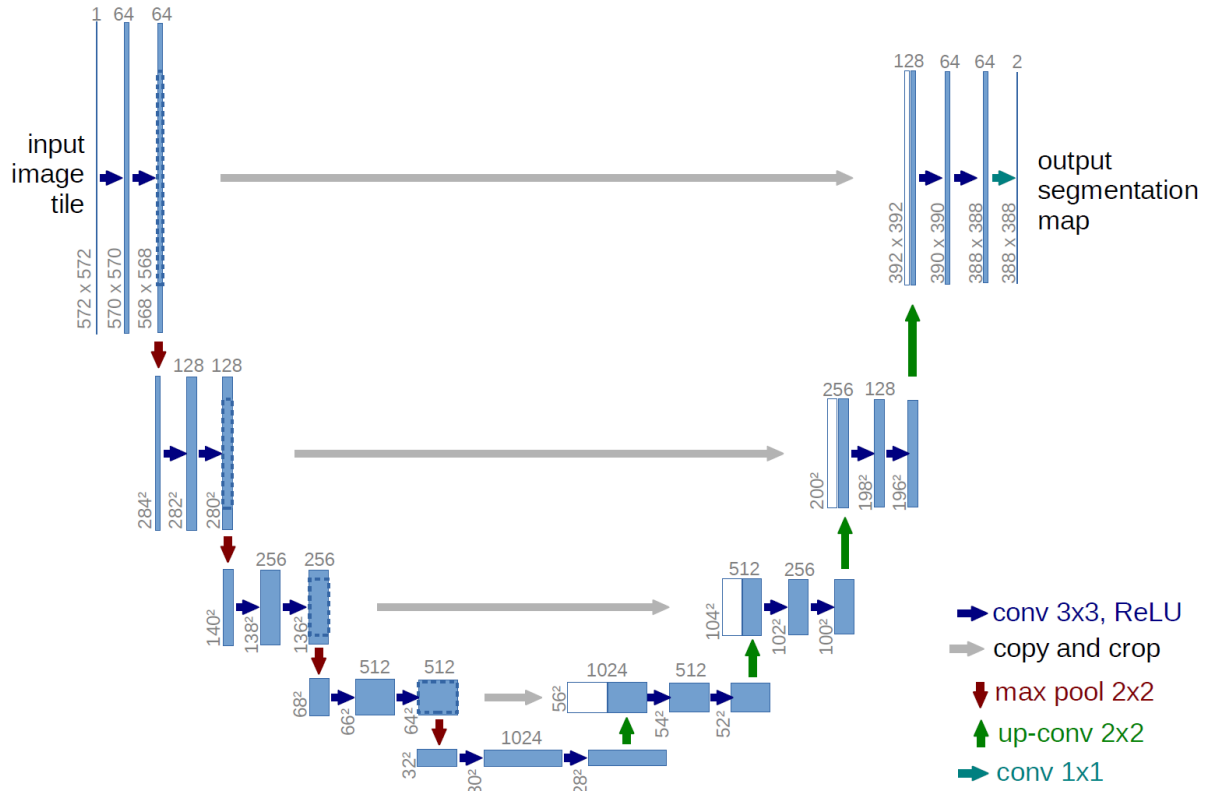


Figure 9. U-Net architecture for image segmentation. Adapted from U-Net (Ronneberger et al., 2015).

As illustrated in Figure 9, the image is input into the left path of the U-Net architecture. As previously described, this left path is used for compressing the image. The U-Net architecture consists of a total of nine layers. The input image is compressed through the first five layers, known as the left encoder path. The primary objective of the encoder path is to extract as many features as possible. As the image passes through the compression path, the number of feature maps doubles. This process facilitates deeper learning of important information in the image and enhances the representation of abstract features.

The compressed image is then passed to the bottleneck layer for upsampling and subsequently forwarded to the right decoder path. The decoder path reconstructs the image back to its original dimensions through several expansion blocks. This operation restores local features of the image, enabling accurate segmentation results (Ronneberger et al., 2015).

During the decompression process in the decoder path, the number of feature maps gradually reduces by half. In the final stage, the output from the last layer of the decoder path is flattened through a 1×1 convolutional layer and subsequently passed to an activation function for the final decision. The output obtained from the activation function is then resized to match the desired dimensions, similar to the input image. This process yields the ultimate predictions for the segmentation task (Ronneberger et al., 2015).

In studies on detection and segmentation of liver and liver tumors, the original U-Net architecture has been modified by integrating different skip connections and adding multiple paths to extract multiple features. These modifications allow the model to learn richer features and improve segmentation accuracy.

The deep learning model used in this study is primarily structured based on the U-Net architecture. The model accepts multi-channel MR images with dimensions of $512 \times 512 \times 3$ as input. The initial Lambda layer is employed for preprocessing or applying custom functions. Subsequent convolutional (Conv2D) layers extract features from the images, while dropout layers are incorporated to prevent overfitting.

The convolutional layers progressively increase the number of channels (16, 32, 64, 128, 256), enabling the learning of increasingly complex and abstract features. After each convolutional block, max pooling (MaxPooling2D) reduces the spatial dimensions of the feature maps by half, thereby decreasing the computational load.

In the decoder part of the model, transpose convolutional (Conv2DTranspose) layers and upsampling operations gradually restore the feature map sizes back to the original resolution. During this stage, concatenation operations merge low- and high-level features between the encoder and decoder layers, preserving detailed information and enhancing segmentation accuracy.

The final output layer reduces the number of channels to one using a 1×1 Conv2D layer, producing the segmentation mask. The model contains approximately 1.94 million trainable parameters and no non-trainable parameters.

This architecture allows the model to achieve high segmentation performance on high-resolution medical images while minimizing the risk of overfitting.

The detailed model architecture and inter-layer connections used in this study are summarized in Table 1.

Table 1. Architecture Summary of the U-Net-Based Deep Learning Model for Medical Image Segmentation

Layer(type)	Output Shape	Param #	Connected to
Input_layer(Input_Layer)	(None,512,512,3)	0	-
lambda(Lambda)	(None,512,512,3)	0	input_layer[0][0]
conv2d(Conv2D)	(None,512,512,16)	448	Lambda[0][0]
Dropout (Dropout)	(None,512,512,16)	0	conv2d[0][0]
conv2d_1(Conv2D)	(None,512,512,16)	2320	Dropout[0][0]
max_pooling2d(MaxPooling2D)	(None,256,256,16)	0	conv2d_1[0][0]
conv2d_2(Conv2D)	(None,256,256,32)	4640	max_pooling2d[0][0]
dropout_1(Dropout)	(None,256,256,32)	0	conv2d_2[0][0]
conv2d_3(Conv2D)	(None,256,256,32)	9248	dropout_1[0][0]
max_pooling2d_1(MaxPooling2D)	(None,128,128,3)	0	conv2d_3[0][0]
conv2d_4(Conv2D)	(None,128,128,64)	18496	Max_pooling2d_1[0][0]
dropout_2(Dropout)	(None,128,128,64)	0	conv2d_4[0][0]
conv2d_5(Conv2D)	(None,128,128,64)	36928	dropout_2[0][0]
max_pooling2d_2(MaxPooling2D)	(None,64,64,64)	0	conv2d_5[0][0]
conv2d_6(Conv2D)	(None,64,64,128)	73856	max_pooling2d_2[0][0]
dropout_3(Dropout)	(None,64,64,128)	0	conv2d_6[0][0]

Conv2d_7(Conv2D)	(None,64,64,128)	147584	dropout[0][0]
max_pooling2d_3(MaxPooling2D)	(None,32,32,128)	0	conv2d_7[0][0]
conv2d_8(Conv2D)	(None,32,32,256)	295168	max_pooling2d_3[0][0]
dropout_4(Dropout)	(None,32,32,256)	0	conv2d_8[0][0]
conv2d_9(Conv2D)	(None,32,32,256)	590080	dropout_4[0][0]
conv2d_transpose(Conv2DTranspose)	(None,64,64,128)	131200	conv2d_9[0][0]
concatenate(Concatenate)	(None,64,64,256)	0	conv2d_transpose conv2d_7[0][0]
conv2d_10(Conv2D)	(None,64,64,128)	295040	concatenate[0][0]
dropout_5 (Dropout)	(None,64,64,128)	0	conv2d_10[0][0]
conv2d_11 (Conv2D)	(None,64,64,128)	147584	dropout_5[0][0]
conv2d_transpose_1 (Conv2DTranspose)	(None,128,128,64)	32832	conv2d_11[0][0]
concatenate_1 (Concatenate)	(None,128,128,128)	0	conv2d_transpose... conv2d_5[0][0]
conv2d_12 (Conv2D)	(None,128,128,64)	73792	concatenate_1[0]...
dropout_6 (Dropout)	(None,128,128,64)	0	conv2d_12[0][0]
conv2d_13 (Conv2D)	(None,128,128,64)	36928	dropout_6[0][0]
conv2d_transpose_2 (Conv2DTranspose)	(None,256,256,32)	8224	conv2d_13[0][0]
concatenate_2 (Concatenate)	(None,256,256,64)	0	conv2d_transpose... conv2d_3[0][0]
conv2d_14 (Conv2D)	(None,256,256,64)	18464	concatenate_2[0]...

dropout_7 (Dropout)	(None,256,256,32)	0	conv2d_14[0][0]
conv2d_15 (Conv2D)	(None,256,256,32)	9248	dropout_7[0][0]
conv2d_transpose_3 (Conv2DTranspose)	(None,512,512,32)	2064	conv2d_15[0][0]
concatenate_3 (Concatenate)	(None,512,512,32)	0	conv2d_transpose... conv2d_1[0][0]
conv2d_16 (Conv2D)	(None,512,512,16)	4624	concatenate_3[0]...
dropout_8 (Dropout)	(None,512,512,16)	0	conv2d_16[0][0]
conv2d_17 (Conv2D)	(None,512,512,16)	2320	dropout_8[0][0]
conv2d_18 (Conv2D)	(None,512,512,1)	17	conv2d_17[0][0]
Total params: 1,941,105 (7.40 MB) Trainable params: 1,941,105 (7.40 MB) Non-trainable params: 0 (0.00 B)			

3.3.3. Digital Image Processing

Medical image segmentation faces significant limitations arising from various factors such as constraints in image quality, anatomical complexity, structural variability, scarcity of labeled data, and operational challenges during segmentation processes. Research in this field has focused on improving medical image segmentation through thresholding, region growing, edge detection, clustering, machine learning, and deep learning approaches.

The optimization of CNN performance critically depends on factors such as the availability of large-scale annotated datasets and consideration of variability in image quality. Particularly, to achieve effective segmentation in images with low contrast and irregular structures, preprocessing with image enhancement techniques is often necessary (Saifullah & Dreżewski, 2024).

The image processing techniques applied in this study are detailed below.

Histogram Equalization

Histograms are the basis of many spatial domain processing techniques. Histogram manipulation can be used effectively in image enhancement processes. The histogram of a digital image with grayscale in the range $[0, L-1]$ is defined as a discrete function as follows:

$$h(r_k) = n_k \quad (8)$$

Here, r_k represents the input or feature at index k , and h is a function mapping this input to an output n_k , which can represent a transformed value or class label associated with r_k . If we divide the histogram by the total number of pixels in the image, the normalized histogram is obtained. The normalized histogram is expressed as:

$$p(r_k) = \frac{n_k}{n} \text{ for } k = 0, 1, 2, 3, 4 \dots \dots (L-1) \quad (9)$$

$p(r_k)$ represents the probability of the event indexed by k , calculated as the ratio of the count n_k to the total number n . The index k ranges over discrete values $(0, 1, 2, \dots)$, and $L \approx 1$ indicates that the sum of all probabilities is approximately 1. Normalized histograms form the basis of techniques such as histogram equalization and play an important role in image enhancement processes.

In the Histogram Equalization method, the image is transformed so that the output has a uniform distribution that provides better contrast compared to the original image. The transformation $T(r)$ required to achieve this equalization can be expressed mathematically as:

$$s = T(r) = \int_0^w p_r(r) dw = CDF_i \quad (10)$$

$T(r)$ is a transformation function applied to the variable r , defined as the integral of the probability density function $Pr(r)$ from 0 up to w . This integral represents the cumulative distribution function (CDF) evaluated at w , denoted here as CDF_i . This transformation increases the image contrast by redistributing each grayscale level, and provides significant improvements, especially in low-contrast images. Histogram equalization is widely used in many applications, such as medical image processing and natural scene imaging.

$$S_k = T(r_k) = \sum_{j=0}^k p_r(r_j) \text{ for } k = 0, \frac{1}{L-1}, \frac{2}{L-1}, \dots, 1 \quad (11)$$

S_k is the transformed value of r_k through the function T , calculated as the cumulative sum of probabilities $p_r(r_j)$ from $j = 0$ up to k . The index k takes discrete steps between 0 and 1 in increments of $1/(L-1)$, representing a normalized scale. r and s represent the intensity values of the input and output images, respectively. The term $p_r(r)$ denotes the probability distribution function (PDF) of the input image, obtained by normalizing the histogram of the original image such that its total area equals one.

Additionally, CDF_i represents the cumulative distribution function (CDF) of the input image. The transformation required to perform histogram equalization is essentially the CDF of the original image. This transformation redistributes the intensity values in the output image to achieve a more uniform intensity distribution, thereby enhancing the image contrast (Srivastava & Rawat, 2013).

Contrast Limited Adaptive Histogram Equalization (CLAHE) Algorithm

The CLAHE method is a technique used to enhance image contrast. Unlike conventional histogram equalization, CLAHE provides an adaptive approach by computing separate

histograms for each region of the image. These histograms are used to redistribute brightness values within the regions, resulting in improved local contrast and enhanced edge definition across different parts of the image. This method effectively makes fine details more prominent. Images and other signals are often described using mathematical models. Signals can be represented as one-dimensional functions (time-dependent), two-dimensional functions (dependent on two planar coordinates), or three-dimensional functions. Typically, signals are modeled as functions dependent on multiple parameters. This approach can be applied to dimensional techniques such as tracking volumetric objects in space. Dimension-related equations are used for monitoring such objects (Venkatesh et al., 2022).

This method processes the image by dividing it into small tiles rather than using the entire image at once. This approach helps prevent excessive contrast amplification while also reducing noise. The operation is applied only to the luminance channel of a color image, and processing this channel alone yields better results compared to adjusting all BGR color channels. The CLAHE algorithm is used to enhance image contrast and can be applied to both grayscale and color images (Venkatesh et al., 2022).

The steps of the CLAHE algorithm are as follows:

- The process begins with an input image, which can be either grayscale or color.
- The image is divided into small, non-overlapping tiles or regions for local contrast enhancement.
- For each tile, a histogram representing the distribution of pixel intensities within that region is computed.
- A transformation function is applied to the histograms, and a clip limit is calculated. This clip limit defines the maximum allowable contrast enhancement to prevent excessive contrast amplification in the image.
- Each tile's histogram is modified according to the defined clip limit, ensuring that the contrast enhancement does not exceed the desired threshold.
- After applying the contrast adjustments, the image is evaluated to verify whether the desired contrast improvement has been achieved and to check for the presence of any unwanted artifacts.

- Finally, the modified tiles are recombined to produce the output image with enhanced local contrast (Venkatesh et al., 2022).

Gaussian Filtering

The Gaussian filter is a widely used technique in image processing, primarily applied to reduce noise in images. Since noise is associated with high-frequency components, the Gaussian filter suppresses these components to produce a more homogeneous and smoother image. Additionally, before edge detection and delineation processes, it is often necessary to smooth sharp transitions in the images. This requirement leads to the frequent use of Gaussian filtering as a preprocessing step in edge detection algorithms, such as the Sobel filter or Canny edge detection algorithm. Furthermore, the Gaussian filter is effective in more complex operations like image enhancement, feature extraction, and segmentation, particularly excelling in reducing image degradations such as blurring (Wuthrich et al., 2015).

One of the most important advantages of the Gaussian filter is its mathematically smooth distribution, which allows for simpler and faster analytical solutions. Additionally, since many phenomena in the natural world can be modeled by the Gaussian distribution, this filter effectively operates on real-world data. Another advantage is that the Gaussian function offers an efficient computational method for convolution operations. This enhances the speed of image processing tasks while optimizing memory and processor usage. Consequently, the Gaussian filter stands out as a crucial tool in image processing applications due to its effectiveness, efficiency, and mathematical accuracy.

The Gaussian filter is a fundamental linear filter widely used in image and signal processing. Its mathematical foundation is based on the Gaussian distribution, also known as the normal distribution function. This function represents the distribution of continuous random variables observed in many natural processes and is commonly associated with various physical measurements such as light, temperature, or sound.

The primary purpose of the Gaussian filter is to eliminate noise in images and smooth high-frequency components, thereby ensuring smoother transitions and preserving low-frequency information. Additionally, it is often employed as a preprocessing step for edge smoothing or more complex image analysis techniques.

In the context of image processing and statistical filtering, the approximate posterior distribution $q(x|y)$ represents the distribution of the variable x given the observed data y . For the parameters of $q(x|y)$ to be efficiently computed from joint expectations, it is important that the functional form of $q(x|y)$ allows for the derivation of these parameters through either analytical or approximate methods.

To overcome this computational complexity, existing joint expectation filters often approximate the true joint distribution $p(x,y)$ with a Gaussian distribution. This approximation is preferred due to the mathematical simplicity of the Gaussian distribution and its complete characterization by only a mean and variance. These properties make the Gaussian distribution particularly useful in image processing tasks such as noise reduction, smoothing, and denoising (Wuthrich et al., 2015).

$$q(x, y) = N \left(\begin{pmatrix} x \\ y \end{pmatrix} \middle| \begin{pmatrix} \mu_x \\ \mu_y \end{pmatrix}, \begin{pmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_{yy} \end{pmatrix} \right) \quad (12)$$

$q(x,y)$ denotes a joint normal (Gaussian) distribution of variables x and y . The notation $(x|y)$ means the distribution of x given y . The vector $(\mu_x|\mu_y)$ represents the means of x and y , and the matrix with elements $\Sigma_{xx}, \Sigma_{xy}, \Sigma_{yx}, \Sigma_{yy}$ represents the covariance between and within x and y .

Approximating the correct joint distribution $p(x,y)$ with a Gaussian distribution simplifies the parameter calculations of the joint expectation filters, thus reducing the complexity of the filtering process. In particular, the joint distribution $p(x,y)$ is approximated as a multivariate Gaussian distribution by assuming that x and y jointly follow a Gaussian distribution. This allows the joint expectations to be calculated efficiently using standard techniques from probability theory. As a result, approximating the correct joint distribution $p(x,y)$ with a Gaussian distribution enables the joint expectation filters to work more efficiently, because

the mathematical framework is simplified and the parameters required for the filtering process are easier to calculate (Wuthrich et al., 2015).

Sigma represents the standard deviation of the Gaussian distribution. Higher sigma values cause the distribution to broaden and have more smoothing effects. Higher sigma values cause the cleaning to be more aggressive, but this can also result in some important details being lost. In particular, it means that the edges of objects will become less distinct. Lower sigma values help preserve fine details and edges in the image, while providing less noise removal. It is generally preferred when it is desired to reduce noise in the dataset (Wuthrich et al 2015).

3.4. Performance Evaluation

The mean F-score (mFscore), also sometimes referred to as the average F1-score, is a widely used metric for evaluating the performance of semantic segmentation models. This metric provides a balanced measure between precision and recall by considering the model's performance across all classes. We utilize the mFscore as the primary metric to assess our models during both hyperparameter tuning and testing phases.

The calculation of the mFscore is typically based on the confusion matrix defined for binary classification problems. However, since our problem involves multi-class classification, we adapt the definitions of the confusion matrix accordingly to suit this context (Müller et al., 2022).

The confusion matrix in a multi-class problem summarizes the model's predictions and reflects the following classification outcomes:

True Positives (TP): Pixels correctly classified by the model. In the context of multi-class segmentation, a true positive refers to pixels that belong to a specific class and are accurately identified as such.

True Negatives (TN): Pixels that do not belong to the class that the model correctly classified. Multi-class segmentation features, true negative, do not belong to a certain class and are recorded pixels without correctly belonging to that class.

False Positives (FP): Pixels incorrectly classified as belonging to a class when they actually belong to another class. In multi-class problems, false positives occur when a pixel is assigned to the wrong class, potentially leading to errors in segmentation or classification tasks.

False Negatives (FN): Pixels that truly belong to a class but are incorrectly predicted as belonging to another class. False negatives indicate missed classifications by the model and are especially critical when evaluating model performance in applications where sensitivity is important.

The confusion matrix facilitates the evaluation of various metrics such as accuracy, precision, recall, and F1-score for each class in multi-class classification problems. By providing insights into the types of errors made by the model, it allows identification of areas requiring improvement.

The fundamental difference between FP and FN lies in their definitions relative to predicted versus actual classes. FP refers to pixels incorrectly classified as belonging to a certain class when their true class is different. In contrast, FN describes pixels that actually belong to a given class but are mistakenly classified as belonging to another class.

Using the confusion matrix, various evaluation metrics such as the F1-score, Dice coefficient, and Jaccard index can be calculated for each class. These metrics are particularly useful for assessing the performance of segmentation models in multi-class problems (Altaş İ., 2023).

Precision; is defined as the ratio between the observations correctly predicted as positive (True Positives) and the total predicted positives (True Positives + False Positives). That is, it shows how many of the predicted pixels as a class are actually correct.

$$Precision = \frac{TP}{TP+FP} \quad (13)$$

Precision is a metric that indicates how many of the model's positive predictions are correct. A high precision score indicates the model's ability to reduce false positives, which is

especially important in applications where false positives are costly. If low false positives are critical, precision values of 85%–95% or higher are typically targeted.

Recall (Sensitivity); is defined as the ratio between the observations correctly predicted as positive (True Positives) and the total true positives (True Positives + False Negatives). That is, it measures whether the pixels that actually belong to that class are correctly identified by the model.

$$Recall = \frac{TP}{TP+FN} \quad (14)$$

Especially in critical situations such as the detection of tumors or diseased tissues in the healthcare field, recall values between 85% and 95% or higher are targeted. This aims to ensure that all positive samples (e.g. diseased areas) are detected with the least possible error.

Accuracy Score, is a key metric used to measure a model's correct predictions. This metric takes the sum of both true positive (TP) and true negative (TN) predictions and divides it by the total number of predictions made.

$$Accuracy = \frac{TP+TN}{TP+TN+FN+FB} \quad (15)$$

In image segmentation, the accuracy metric is a commonly used metric as an indicator of good performance; however, its ideal value range can vary significantly depending on the application type, dataset, and problem difficulty. Accuracy is based on the sum of both true positive and true negative examples, which can be misleading, especially in imbalanced datasets or when the background is sparse. In medical imaging, accuracies above 85%–95% are generally considered satisfactory.

F1-score, It is the harmonic mean of Precision and Recall. It provides a balanced measure between Precision and Recall and is especially meaningful in cases where there is class imbalance (i.e., classes are unbalanced).

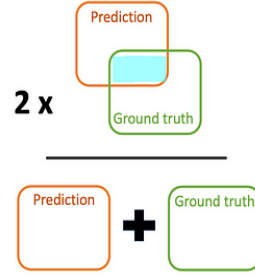
$$Fscore = \frac{2*(Precision*Recall)}{Precision+Recall} = \frac{2*TP}{2*TP+FP+FN} \quad (16)$$

The mean F-score (mFscore) is calculated by averaging the F1-scores of each class. This metric summarizes the overall performance of the model across all classes and gives equal weight to the performance in each class. mFscore is particularly useful when working with multi-class segmentation problems because it provides a balance of accuracy and recall for all classes.

Dice Coefficient; is a metric used to measure the similarity between two sets and has an important place in the field of image segmentation. Developed by American biologist Lee Raymond Dice, this coefficient is often used to evaluate accuracy in biomedical imaging and other segmentation problems. Dice Coefficient is a measure of the overlap between two binary sets. A higher Dice score indicates a more accurate segmentation; especially in medical image segmentation, a Dice score between 70% and 95% is generally acceptable and considered good performance. In more sensitive applications, such as organ demarcation or tumor segmentation, Dice scores above 90% are the target.

$$Precision = \frac{\text{intersecting area}}{\text{predicted area}} \quad (17)$$

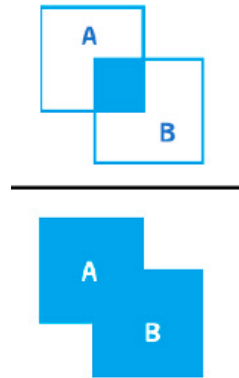
$$Recall = \frac{\text{intersecting area}}{\text{total area}} \quad (18)$$



$$Dice = \frac{2 * Area\ of\ overlap}{Total\ area} \quad (19)$$

F1-score and Dice Score are mathematically equivalent in binary or multi-class segmentation problems. This equivalence demonstrates their usefulness in evaluating the overlap between predicted and true domains.

Jaccard Index (IoU); Jaccard Index measures the similarity between two clusters. The similarity between the predicted positive pixel or region (A) in the output of a segmentation problem and the true positive pixel or region (B) is calculated as follows:



$$Intersection\ over\ Unit(IoU) = \frac{Area\ of\ overlap - |A \cap B|}{Area\ of\ union - |A \cup B|} \quad (20)$$

In image processing applications, when evaluating the performance of object detection models, IoU generally considers a value of 50% or above as “successful.” This threshold value can be adjusted depending on the model and application scenario. As a result, the Jaccard Index is an important tool for evaluating the performance of segmentation and object detection models and is considered a standard metric in model evaluations. Proper implementation of this metric is critical to understanding and improving the overall success of the model.

3.5. Model Parameters

The model training was conducted using the Keras API within the TensorFlow library. During the training process, the data directory and image size were printed to the terminal for tracking purposes. To save the best model, the ModelCheckpoint callback was employed, preserving the model with the lowest validation loss (val_loss). Additionally, EarlyStopping was used to halt training if the validation loss did not improve for 20 consecutive epochs, preventing overfitting. The TensorBoard callback was also activated to enable visualization throughout the training process. The model was trained with a batch size of 16 over 100 epochs, reserving 10% of the training data for validation. The training history was stored in the history variable for subsequent analysis.

Model training was conducted using the high-level Keras API within the TensorFlow library. To enhance traceability during the training process, the data directory and image size were outputted to the terminal. The ModelCheckpoint callback was employed to monitor model performance and save the model achieving the lowest validation loss (val_loss).

To prevent overfitting, the EarlyStopping callback was applied, terminating training early if no significant improvement in validation loss was observed for 20 consecutive epochs. Additionally, the learning rate was dynamically adjusted during training; the ReduceLROnPlateau callback reduced the learning rate when validation loss plateaued, aiming to improve the model’s generalization capability.

During training, the TensorBoard callback was enabled to visualize the model’s behavior and performance curves through TensorBoard. Additionally, the training history was saved in CSV format via the CSVLogger callback, allowing for subsequent analyses. The model was

trained for 100 epochs with a mini-batch size of 16, allocating 10% of the data for validation. Training outputs were stored in the history object, enabling continuous monitoring of the model's loss and metric changes throughout the training process.



4. RESULTS AND DISCUSSION

In this study, MR images subjected to segmentation using the ATLAS dataset and CU-HCC Datasets were evaluated separately at resolutions of 128×128 , 256×256 , and 512×512 . The results are presented sequentially, starting with the ATLAS dataset followed by CU-HCC Dataset. Deep learning techniques were implemented using the Keras and TensorFlow libraries. To assess the adequacy of model training, an early stopping strategy based on the "patience" criterion one of TensorFlow's training termination conditions was employed.

4.1. ATLAS Dataset Results

4.1.1. Segmentation Performance Evaluation According to Different Resolutions (Different Dimensions)

In the training conducted with the ATLAS dataset, segmentation was performed using the U-Net algorithm on data scaled to 128×128 resolution. Although the training process was initially planned for a total of 100 epochs. Throughout training, loss values consistently decreased while accuracy increased, indicating an effective learning process exhibited by the model.



Figure 10. Different size images, masks and result predictions of the same patient

The estimated regions on the original image are enclosed in the areas marked in red so that they can be easily examined by radiology specialists. This visual was prepared to better understand the results of the segmentation process (Figure 11).

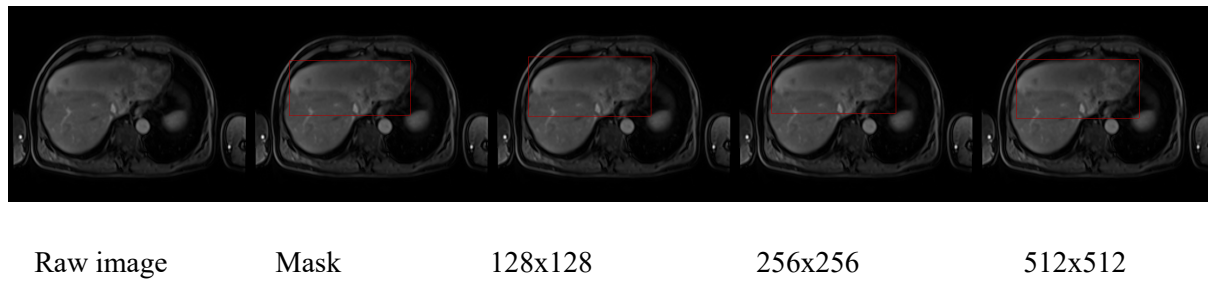


Figure 11. Framed form of mask and prediction dimensions in the input image of the same patient in different sizes

Figure 12 presents a visualization of the predicted region obtained as a result of the segmentation process. The areas represented by different colors in the image correspond to various classes and segments identified by the model.

This visualization is critically important for the analysis of segmentation outcomes and enables expert clinicians to evaluate the model's performance. In medical imaging applications, such detailed analyses significantly contribute to clinical decision-making processes and highlight the potential for improving segmentation model accuracy.

The predicted masks and ground truth masks were compared pixel-wise to generate a new colored image. Each color tone in Figure 12 corresponds to different values in the confusion matrix. This type of analysis provides a rich visualization for assessing segmentation model performance and allows for a more detailed examination of the model's accuracy and classification success.

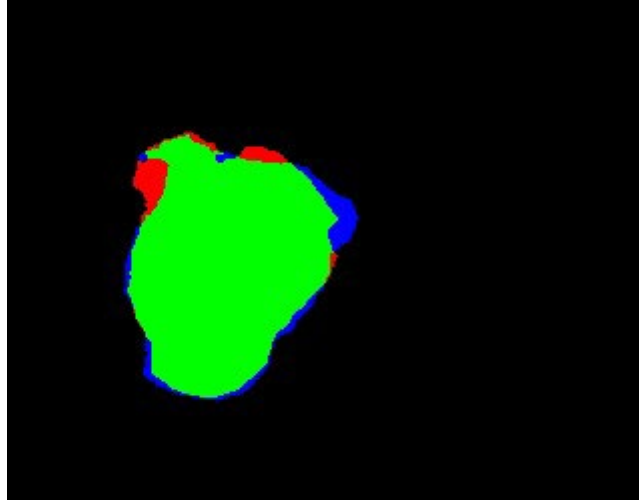


Figure 12. Segmentation masks prediction result

- **Green Areas (True Positives):** Represent positive segments accurately identified by the model. These regions correspond to areas effectively classified by the segmentation model.
- **Red Areas (False Negatives):** Indicate regions incorrectly predicted by the model, demonstrating that some positive examples were misclassified.
- **Blue Areas (False Positives):** Reflect regions classified as false negatives. These areas are actual positives that the model failed to detect.
- **Black Areas (True Negatives):** Denote regions marked as tumor-free in both the predicted and ground truth images.

In Figure 13, sample outputs of segmentation operations performed at different resolution levels (128×128 , 256×256 , 512×512) are visually compared. In addition to the original T2-weighted MR image, the visual includes the ground truth (reference) mask and the prediction masks produced by the model at different resolutions. In the segmentation outputs, green represents true positive regions, blue represents false negative regions and red represents false positive classifications. This visual comparison allows us to directly observe how segmentation performance varies depending on the resolution change.

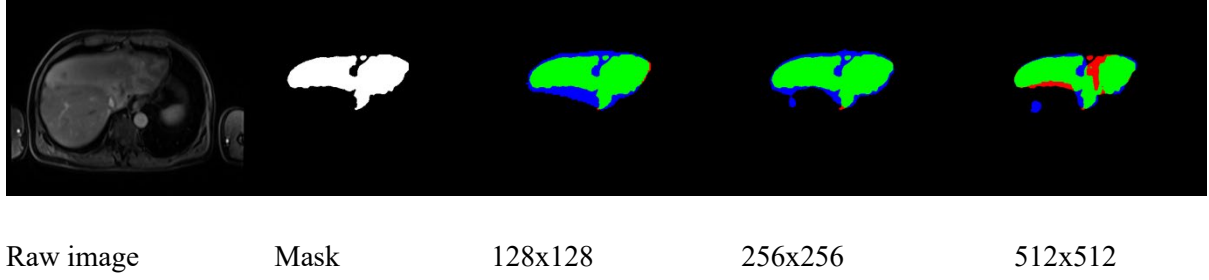


Figure 13. Mask and prediction results for the same patient's image given as input in different sizes

When the performance of the segmentation process performed according to MRI dimensions was evaluated (Table 2, Figure 14),

The Dice similarity coefficient was found to be 0.797 (%79.7) as a result of training with 128x128 data, 0.8419 (%84.19) as a result of training with 256x256 data, and 0.853 (%85.3) as a result of training with 512x512 data.

The IoU value was found to be 0.6625 (%66.25) as a result of training with 128x128 data, 0.727 (%72.7) as a result of training with 256x256 data, and 0.7437 (%74.37) as a result of training with 512x512 data.

Pixel Accuracy was recorded as 0.9881 (%98.81) as a result of training with 128x128 data, 0.9909 (%99.09) as a result of training with 256x256 data, and 0.9926 (%99.26) as a result of training with 512x512 data.

Precision value was obtained as 0.7195 (%71.95) as a result of training with 128x128 data, 0.7686 (%76.86) as a result of training with 256x256 data, and 0.8837 (%88.37) as a result of training with 512x512 data.

The recall value was determined as 0.8933 (89.33%) as a result of training with 128x128 data, 0.9307 (93.07%) as a result of training with 256x256 data and 0.8244 (82.44%) as a result of training with 512x512 data.

These results reveal the effect of datasets of different sizes on the performance of the segmentation model. In particular, it is noteworthy that segmentation success generally improves as the size increases. The Dice coefficients and IoU values obtained in the trainings at 256x256 and 512x512 sizes show that the model has undergone a positive learning process and provides more accurate segmentation.

When examined in terms of pixel accuracy, high values were obtained in all sizes, and these results show that the overall accuracy rate of the model is quite satisfactory. The similar increase in sensitivity and precision values reveals that the model has a strong capacity to detect positive classes. However, while the high sensitivity (88.37%) obtained at 512x512 size emphasizes the success of the model in detecting true positives, the low sensitivity (84.59%) indicates that some positive examples were missed.

Table 2. Segmentation performance of different sized images of ATLAS Dataset

SIZE	DICE	IoU	PIXEL_ACU	RECALL	PRECISION	SPECIFI
predicts_128	0,7967	0,6625	0,9881	0,8933	0,7195	0,9907
predicts_256	0,8419	0,727	0,9909	0,9307	0,7686	0,9925
predicts_512	0,853	0,7437	0,9926	0,8244	0,8837	0,9971

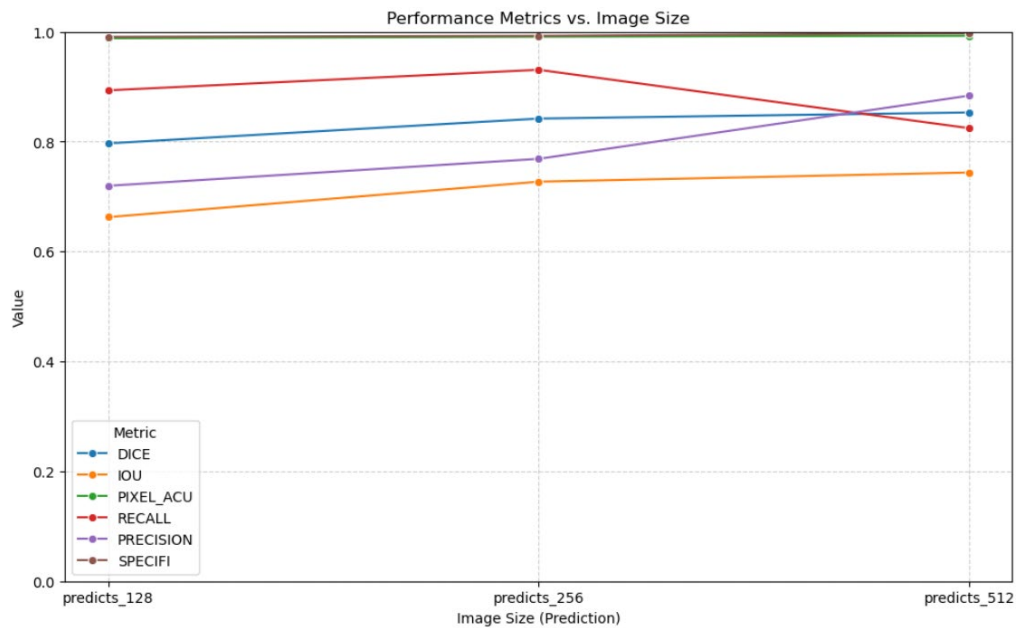


Figure 14. Segmentation performance of different sized images of ATLAS Dataset

4.1.2. Effect of Gaussian Filter Parameter (Sigma) on Segmentation Performance at Different Resolutions

The effect of Gaussian filter on deep learning algorithms was investigated. Gaussian filter was applied to different MR image sizes of 128x128, 256x256 and 512x512 and the effect of the filter parameter sigma value was investigated. In the application, MR images of different sizes were passed through the Gaussian filter and results were obtained for cases where the sigma value varied between 1 and 5.

In Table 3 and Figure 15, the segmentation performance of 128x128 images was evaluated according to different Gaussian sigma values. The obtained metrics show that the performance of the model varies depending on the sigma value.

Dice, initially the Dice coefficient of the estimation without using sigma was 0.7967, while this value increased to 0.7976 for sigma 1. This shows that low sigma values slightly improve the overall segmentation success of the model. However, with sigma 2, a significant regression occurred, with the Dice coefficient decreasing to 0.7275, followed by 0.7678 and 0.7655 values for sigma 3 and 5, respectively. It was determined as 0.7407 for sigma 4. This situation shows that moderate sigma values can negatively affect segmentation performance, while high or low values give better results.

IoU values were determined as 0.6633 for sigma 1 and 0.5717 for sigma 2. This shows that higher sigma values reduce the ability of the model to identify true positives. The highest IoU value was obtained with sigma 1.

The accuracy of the pixels was observed as 0.9875 at sigma 1 and 0.9813 at sigma 2. While sigma 1 increased the accuracy, higher sigma values caused a decrease in the overall accuracy.

Recall achieved the highest sensitivity of 94.24% for Sigma 1. A decrease in sensitivity was observed at higher sigma values, indicating that some positive segments were missed.

When the precision and Specificity values are examined, the upper values are 69.14%, 98.87% with Sigma 1, and 65.26%, 98.68% with Sigma 5. Again, it is seen that low sigma values increase the ability of the model to predict correctly positive and negative values.

Table 3. Segmentation Performance of Different Gaussian Sigma Values on 128x128 ATLAS Dataset Images

SIZE	DICE	IoU	PIXEL_AC U	RECAL L	PRECISIO N	SPECIF I
predicts_128	0,7967	0,6625	0,9881	0,8933	0,7195	0,9907
predicts_128_sigma_1	0,7976	0,6633	0,9875	0,9424	0,6914	0,9887
predicts_128_sigma_2	0,7275	0,5717	0,9813	0,9546	0,5877	0,9821
predicts_128_sigma_3	0,7678	0,6231	0,9855	0,9216	0,6579	0,9872
predicts_128_sigma_4	0,7407	0,5882	0,9828	0,9402	0,611	0,984
predicts_128_sigma_5	0,7655	0,6201	0,9852	0,9255	0,6526	0,9868

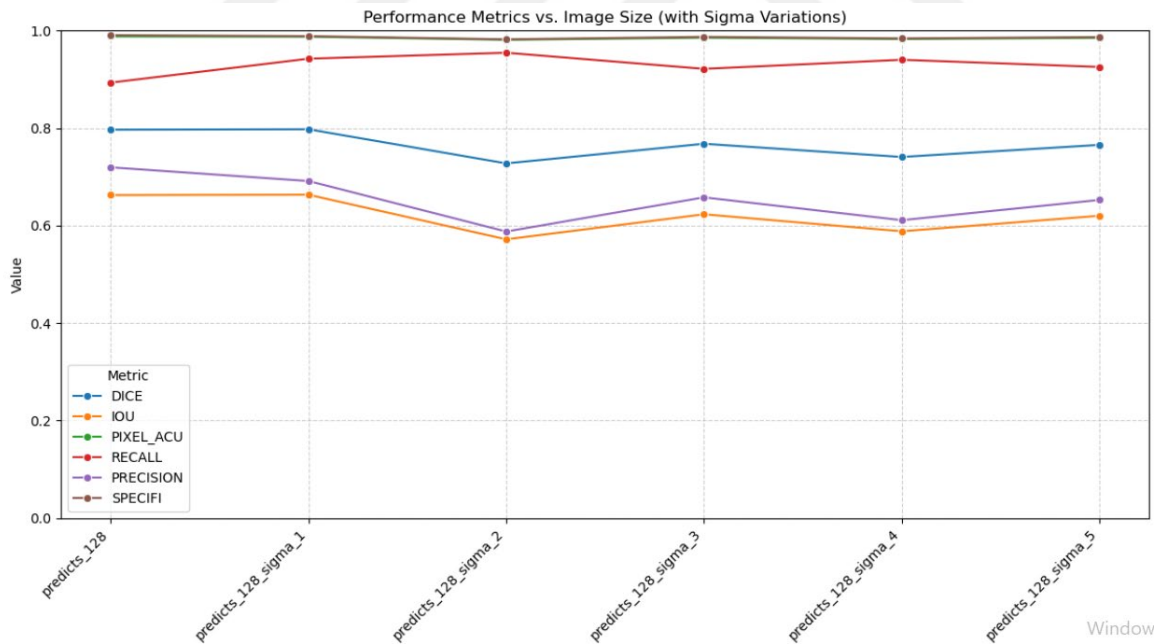


Figure 15. Segmentation Performance of Different Gaussian Sigma Values on 128x128 ATLAS Dataset Images

In Table 4 and Figure 16, the segmentation performance of 256x256 images is evaluated according to different Gaussian sigma values. The obtained metrics show that the performance of the model changes depending on the sigma value.

Initially, the Dice coefficient without applying sigma was 0.8419, which increased to 0.8640 with sigma set to 1. This indicates that low sigma values enhance the overall segmentation performance of the model. However, with sigma set to 2, the Dice coefficient decreased to 0.8301, reflecting a decline. For sigma values of 3 and 4, Dice coefficients of 0.8322 and 0.8463 were obtained, respectively, while sigma 5 led to a rise again to 0.8606. These results suggest that moderate sigma values may negatively impact segmentation performance, whereas both low and high sigma values tend to yield better outcomes.

While the IoU value obtained in Sigma 5 was 0.74, this value was determined as 0.7606 for sigma 1. It was observed that higher sigma values increased the ability of the model to determine true positives, and the highest IoU value was obtained with sigma 1.

Pixel Accuracy, 99.24% accuracy rates were observed with Sigma 1. This shows that low sigma values increase the overall accuracy of the model.

Recall, the highest sensitivity was obtained with 96.55% for Sigma 2. It was observed that the sensitivity decreased at higher sigma values, indicating that some positive segments were missed.

When the precision and specificity values are examined, 81.49% and 99.44% with Sigma 1 and 78.15% and 99.3% with sigma 5 are obtained, respectively. It is seen that low sigma values increase the ability of the model to make accurate positive and negative predictions, but high sigma values are similarly effective.

Table 4. Segmentation Performance of Different Gaussian Sigma Values on 256x256 ATLAS Dataset Images

SIZE	DICE	IoU	PIXEL_ACU	RECALL	PRECISION	SPECIFI
predicts 256	0,8419	0,727	0,9909	0,9307	0,7686	0,9925
predicts 256_sigma 1	0,864	0,7606	0,9924	0,9194	0,8149	0,9944
predicts 256_sigma 2	0,8301	0,7095	0,9897	0,9655	0,728	0,9903
predicts 256_sigma 3	0,8528	0,7434	0,9916	0,9381	0,7817	0,993
predicts 256_sigma 4	0,8486	0,737	0,9914	0,9279	0,7818	0,9931
predicts 256_sigma 5	0,8506	0,74	0,9914	0,9331	0,7815	0,993

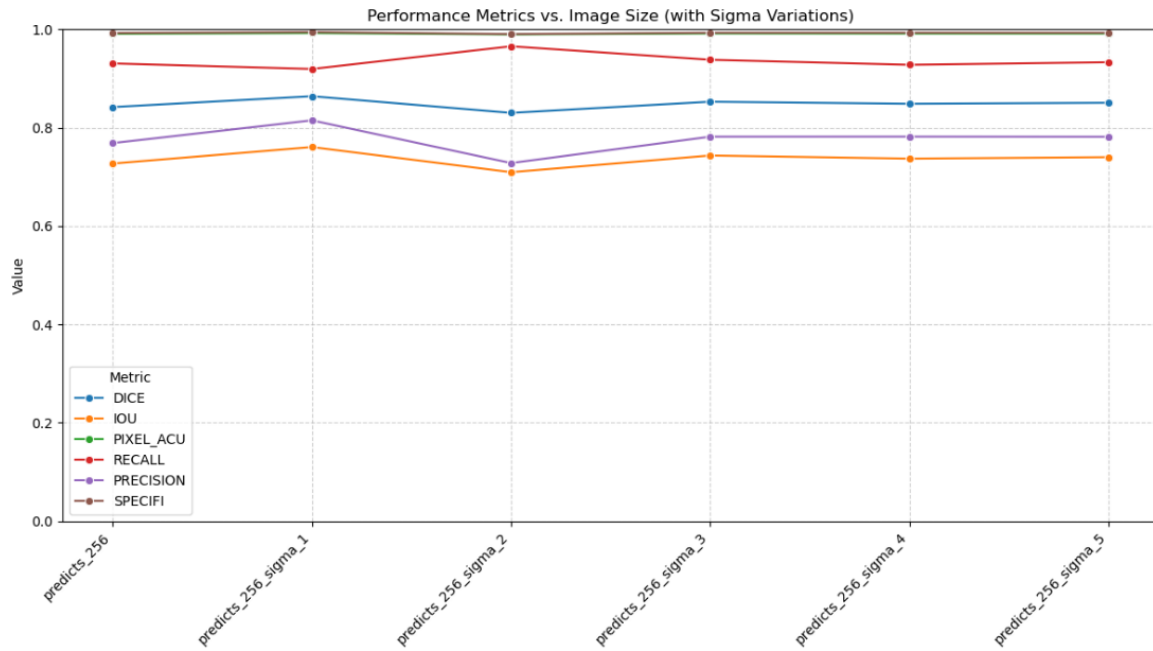


Figure 16. Segmentation Performance of Different Gaussian Sigma Values on 256x256 ATLAS Dataset Images

In Table 5 and Figure 17, the segmentation performance of 512x512 images is evaluated according to different Gaussian sigma values. The results show that the performance of the model varies depending on the sigma value.

The Dice coefficient for the prediction without sigma was determined as 0.8530, which increased to 0.8599 with sigma 1. At sigma 2, it slightly decreased to 0.8567, while sigma 3 recorded the highest value at 0.8885. For sigma 4 and 5, values of 0.7811 and 0.8321 were observed, respectively. These results suggest that low to moderate sigma values may enhance segmentation performance, whereas higher sigma values could adversely affect it.

The IoU value was measured as 0.7543 at sigma 1 and increased to 0.7994 at sigma 3, indicating that higher sigma values improve the model's ability to correctly identify true positives.

Pixel Accuracy was recorded as 99.25% with sigma 1 and 99.07% with sigma 5. This finding indicates that low sigma values enhance the overall accuracy of the model, with sigma 3 achieving an even higher accuracy of 99.38%.

Recall reached its highest value of 94.63% at sigma 3. A decrease in sensitivity was observed at higher sigma values, suggesting that some positive segments may be missed.

When examining Precision and Specificity, the highest values of 86.67% and 99.65% were observed at sigma 2, respectively, while these metrics decreased to 78.31% and 99.34% at sigma 5. This indicates that lower sigma values improve the model's ability to accurately predict both true positives and true negatives.

Table 5. Segmentation Performance of Different Gaussian Sigma Values on 512x512 ATLAS Dataset Images

SIZE	DICE	IoU	PIXEL_ACU	RECALL	PRECISION	SPECIFI
predicts_512	0,853	0,7437	0,9926	0,8244	0,8837	0,9971
predicts_512_sigma_1	0,8599	0,7543	0,9925	0,8867	0,8348	0,9953
predicts_512_sigma_2	0,8567	0,7493	0,9926	0,8469	0,8667	0,9965
predicts_512_sigma_3	0,8885	0,7994	0,9938	0,9463	0,8374	0,9951
predicts_512_sigma_4	0,7811	0,6409	0,9896	0,714	0,8622	0,9969
predicts_512_sigma_5	0,8321	0,7125	0,9907	0,8877	0,7831	0,9934

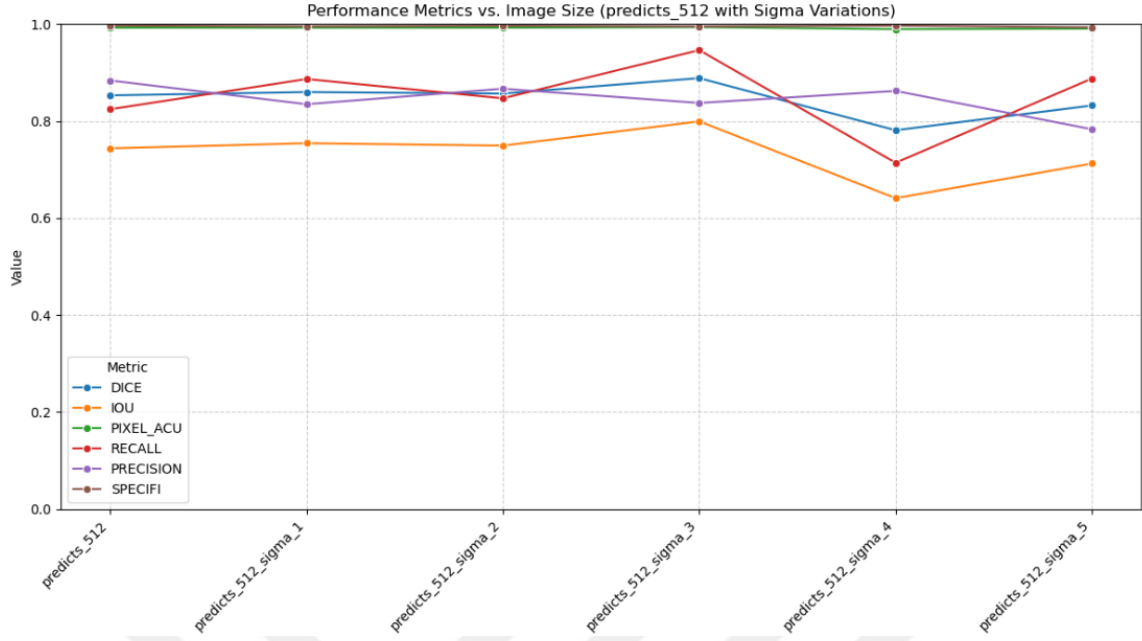


Figure 17. Segmentation Performance of Different Gaussian Sigma Values on 512x512 ATLAS Dataset Images

4.1.3. Effect of Histogram Equalization and CLAHE Methods on Segmentation Performance at Different Resolutions

Histogram equalization is an effective technique used to enhance image contrast and provide clearer visual information. This process rearranges the intensity distribution of pixels within the image, enabling a broader range of intensity values. In practice, histogram equalization helps mitigate issues caused by low illumination conditions and improves the performance of advanced image processing tasks such as object recognition, segmentation, and classification. The results highlight finer details within the image, offering analysts and researchers richer information. Both histogram equalization and CLAHE filters are applied in this process. The performance outcomes of the histogram equalization filter on MRI of different sizes 128×128 , 256×256 , and 512×512 are presented below.

Table 6 and Figure 18 present the segmentation performance of 128×128 resolution images evaluated using models without any filter, with HE, and with CLAHE. The Dice coefficient was 0.7970 for the model without filtering, which decreased to 0.7534 with HE application. When CLAHE was applied, the Dice coefficient improved to 0.7903.

Regarding IoU, the highest value of 0.6625 was achieved without filtering, decreasing to 0.6043 after HE, and partially recovering to 0.6534 with CLAHE.

Pixel accuracy was 98.81% without any filter, dropped to 98.39% with HE, and reached 98.74% when CLAHE was used. These results suggest that histogram equalization may adversely affect the overall accuracy of the model, whereas CLAHE provides a certain positive contribution.

These results indicate that histogram equalization increases the model's false positive rate, while CLAHE partially improves this metric; nevertheless, the model without filtering still offers higher precision.

In terms of recall, histogram equalization achieved 94.13%, outperforming the 89.33% recall of the unfiltered model. This suggests that HE enhances the model's ability to detect positive segmentations.

Precision and specificity values after histogram equalization were 62.80% and 98.51%, respectively, while CLAHE yielded higher values of 69.65% and 98.93%. These results imply that lower sigma values may improve the model's ability to accurately predict both true positives and true negatives.

Table 6. Segmentation Performance of Different Histogram Equalization Methods on 128x128 ATLAS Dataset Images

SIZE	DICE	IoU	PIXEL_ACU	RECALL	PRECISION	SPECIFI
predicts 128	0,7970	0,6625	0,9881	0,8933	0,7195	0,9907
predicts 128 HE	0,7534	0,6043	0,9839	0,9413	0,628	0,9851
predicts 128 CLAHE	0,7903	0,6534	0,9874	0,9134	0,6965	0,9893

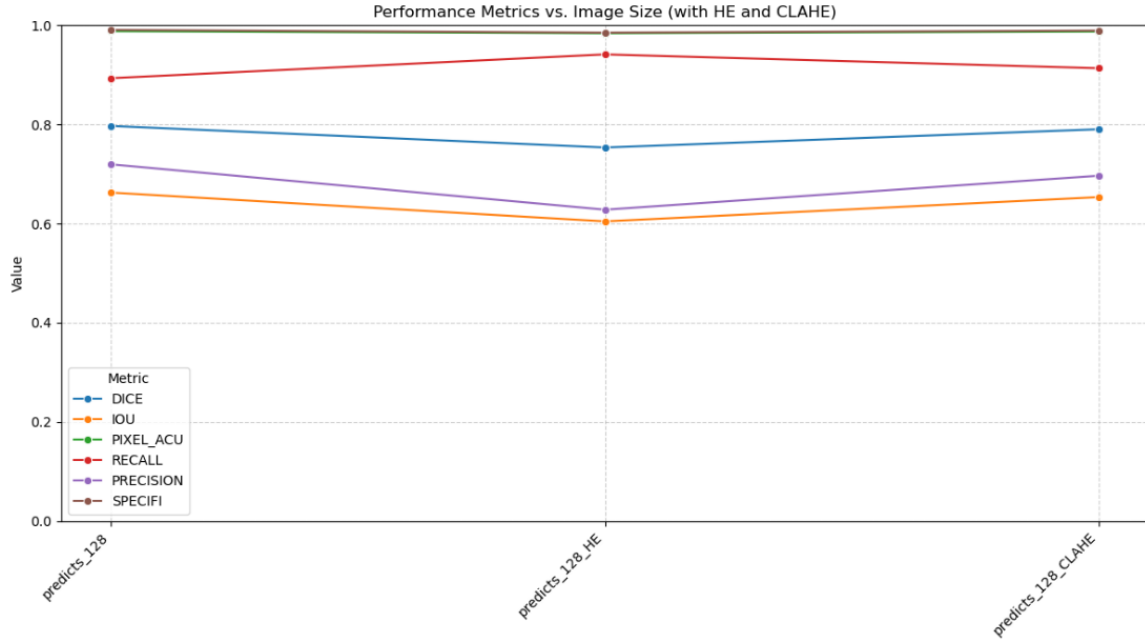


Figure 18. Segmentation Performance of Different Histogram Equalization Methods on 128x128 ATLAS Dataset Images

In Table 7 and Figure 19, the segmentation performance of 256x256 images was evaluated using the unfiltered model and HE and CLAHE methods. The results show that the Dice coefficient of the unfiltered model was 0.8419, while this value increased to 0.8628 with HE. When CLAHE was applied, the value was reached as 0.8651.

When the IoU values were examined, it was observed that this value increased to 0.7587 after HE, while it was 0.7270 without applying the filter. The highest value, 0.7623, was reached with CLAHE.

Pixel accuracy was 99.09% without filtering, but increased to 99.25% when HE was applied. The highest value of 99.28% was reached with CLAHE. These results show that histogram equalization and CLAHE made a certain positive contribution.

In terms of recall, 93.07% was obtained in the model without filtering, 89.89% in the model with HE and 88.72% for CLAHE. It was observed that HE increased the positive segmentation ability of the model.

The CLAHE method demonstrated a significant improvement in segmentation performance based on the obtained results. Considering all metrics, CLAHE achieved the highest Dice

score (86.51%) and IoU value (76.23%). Additionally, it attained the highest Pixel Accuracy at 99.28%. CLAHE also outperformed other methods in metrics such as precision (84.41%) and Specificity (99.56%). While the HE method showed improvements compared to the baseline model, it did not reach the level of success achieved by CLAHE. On the other hand, although the baseline model achieved the highest recall value (93.07%), it lagged behind CLAHE and HE in other metrics. These results indicate that CLAHE is an effective technique for contrast enhancement and improving segmentation accuracy.

Table 7. Segmentation Performance of Different Histogram Equalization Methods on 256x256 ATLAS Dataset Images

SIZE	DICE	IoU	PIXEL_ACU	RECALL	PRECISION	SPECIFI
predicts 256	0,8419	0,727	0,9909	0,9307	0,7686	0,9925
predicts 256 HE	0,8628	0,7587	0,9925	0,8989	0,8295	0,9951
predicts 256 CLAHE	0,8651	0,7623	0,9928	0,8872	0,8441	0,9956

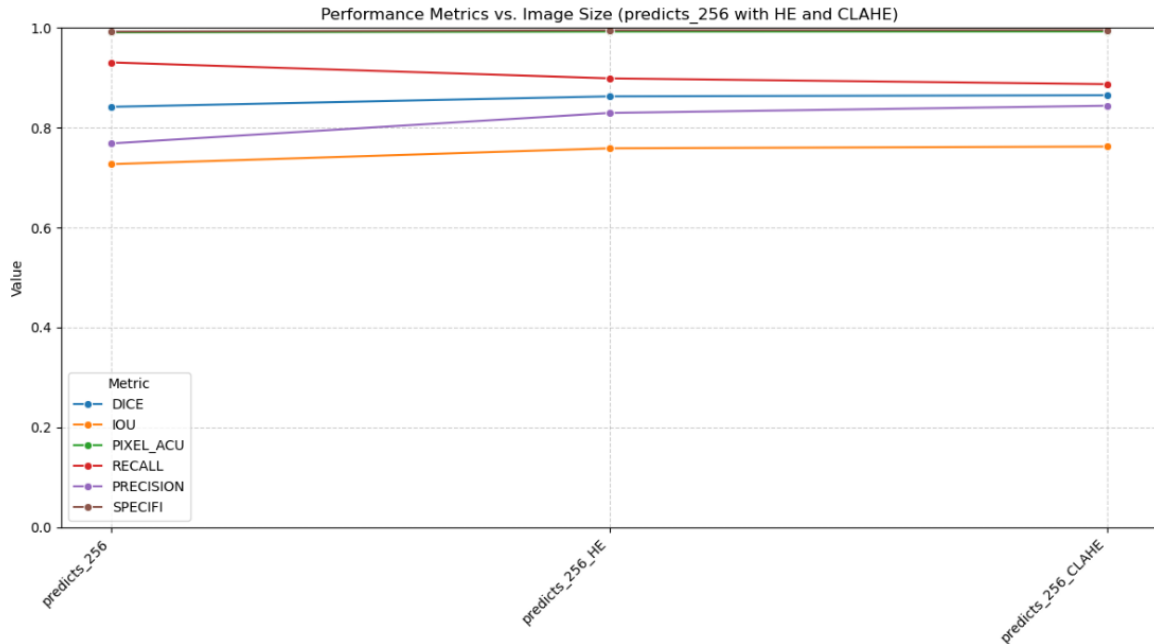


Figure 19. Segmentation Performance of Different Histogram Equalization Methods on 256x256 ATLAS Dataset Images

As shown in Table 8 and Figure 20, the segmentation performance of 512×512 resolution images was evaluated. The Dice coefficient was 0.8530 for the model without any filtering,

which decreased to 0.7914 after applying HE. However, the Dice coefficient increased to 0.8892 with the application of CLAHE.

Regarding IoU values, the unfiltered model achieved 0.7437, which dropped to 0.6549 following HE, and then improved to 0.8005 with CLAHE.

Pixel accuracy was 99.26% without filtering, decreased to 98.95% with HE, and increased again to 99.44% with CLAHE.

In terms of recall, histogram equalization resulted in 76.40%, the unfiltered model achieved 82.44%, and CLAHE reached 86.76%.

When histogram equalization was applied, precision was measured at 82.09% and specificity at 99.55%. In the unfiltered model, precision and specificity were observed as 88.37% and 99.71%, respectively. With CLAHE, these values increased to 91.2% and 99.78%. These results indicate that histogram equalization increases the model's false positive rate, whereas CLAHE provides improvement and offers higher sensitivity.

Table 8. Segmentation Performance of Different Histogram Equalization Methods on 512x512 ATLAS Dataset Images

SIZE	DICE	IoU	PIXEL_ACU	RECALL	PRECISION	SPECIFI
predicts_512	0,853	0,7437	0,9926	0,8244	0,8837	0,9971
predicts_512_HE	0,7914	0,6549	0,9895	0,7640	0,8209	0,9955
predicts_512_CLAHE	0,8892	0,8005	0,9944	0,8676	0,912	0,9978

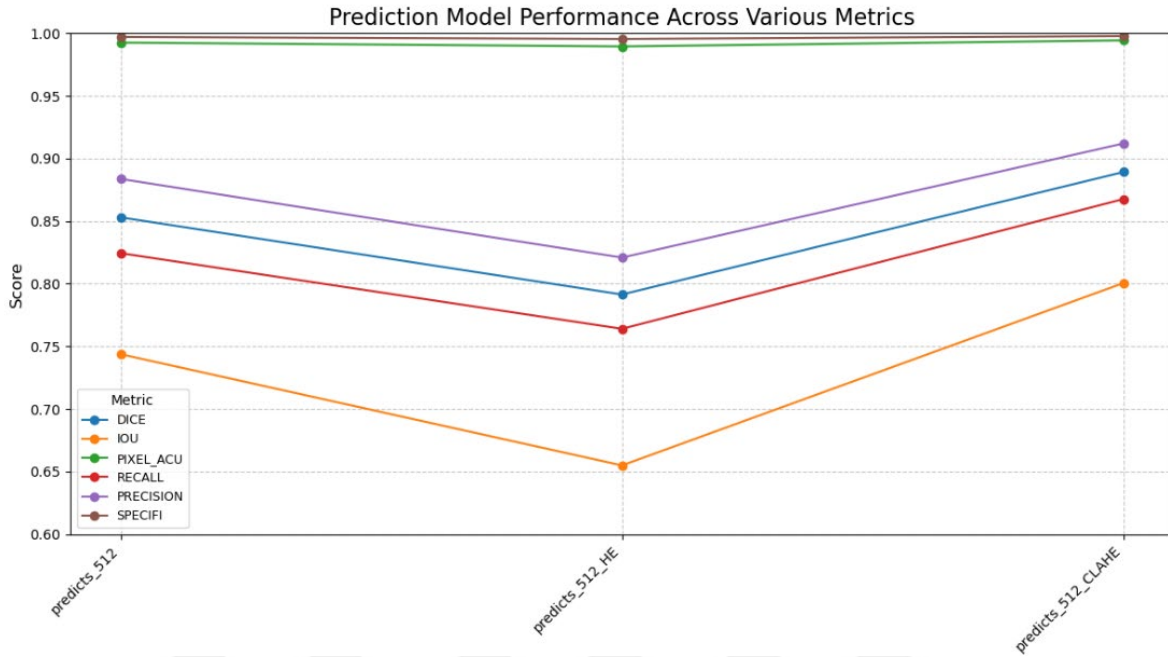


Figure 20. Segmentation Performance of Different Histogram Equalization Methods on 512x512 ATLAS Dataset Images

4.2. Results of CU-HCC Dataset

4.2.1. Segmentation Performance Evaluation According to Different Resolutions (Different Dimensions)

When evaluating the segmentation performance based on MR image sizes (Table 9 and Figure 21),

The Dice similarity coefficient was found to be 0.81 (81%) for the model trained with 128×128 resolution data, 0.7469 (74.69%) for the 256×256 resolution, and 0.8318 (83.18%) for the 512×512 resolution.

The IoU value was found to be 0.6807 (68.07%) as a result of training with 128×128 resolution data, 0.596 (59.6%) with 256×256 resolution data, and 0.712 (71.2%) with 512×512 resolution data.

Pixel Accuracy was recorded as 0.9857 (98.57%) following training with 128×128 resolution data, 0.9801 (98.01%) with 256×256, and 0.9892 (98.92%) with 512×512 resolution data.

The recall value was determined to be 0.9690 (96.9%) as a result of training with 128×128 resolution data, 0.9363 (93.63%) with 256×256, and 0.8529 (85.29%) with 512×512 resolution data.

These results reveal the impact of different input sizes on the performance of the segmentation model. In particular, the Dice coefficients, Pixel Accuracy, and IoU values obtained from the 512×512 training indicate that the model underwent a more effective learning process and achieved more accurate segmentation results.

When examining the recall results, the value for the 128×128 resolution data was quite high at 0.969. This indicates that smaller input sizes were more effective in detecting positive segments. At the 256×256 resolution, recall decreased to 0.9363, which may suggest a slight reduction in the model's ability to identify segments. The recall value at 512×512 was the lowest, at 0.8529. Although higher resolution provides more detail, it is thought that some positive instances might have been missed at this size.

When examining the specificity results, the value was 0.9863 for the 128×128 resolution, 0.9815 for the 256×256 resolution, and highest at 0.9936 for the 512×512 resolution. It was observed that all three image sizes demonstrated high performance in terms of specificity, with the highest performance achieved at the 512×512 resolution.

Table 9. Segmentation performance of images of different sizes in CU-HCC Dataset

SIZE	DICE	IoU	PIXEL_ACU	RECALL	PRECISION	SPECIFI
predicts 128	0,81	0,6807	0,9857	0,969	0,6959	0,9863
predicts 256	0,7469	0,596	0,9801	0,9363	0,6212	0,9815
predicts 512	0,8318	0,712	0,9892	0,8529	0,8116	0,9936

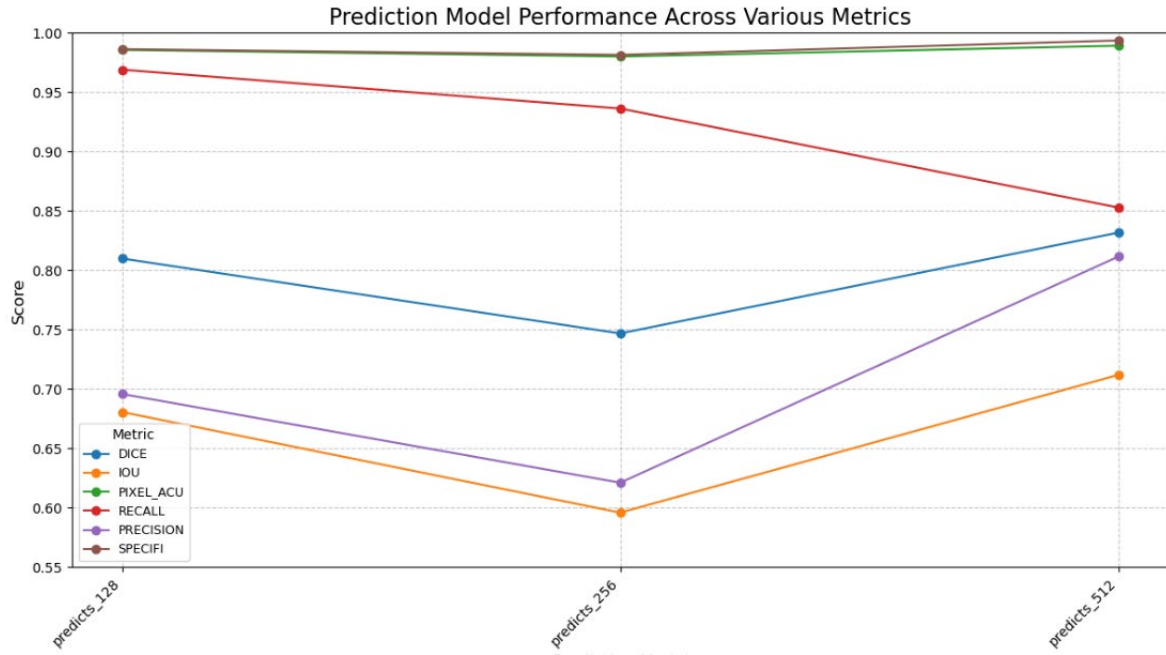


Figure 21. Segmentation performance of images of different sizes in CU-HCC Dataset

4.2.2. Effect of Gaussian Filter Parameter (Sigma) on Segmentation Performance at Different Resolutions

Table 10 and Figure 22 present the segmentation performance of 128×128 resolution images evaluated across different Gaussian sigma values. The obtained metrics demonstrate that the model's performance is influenced by the choice of sigma.

The Dice coefficient without any filtering was calculated as 0.8100. With a sigma value of 1, this score decreased to 0.7987. For sigma values of 2 and 3, the Dice coefficients further declined to 0.7868 and 0.7935, respectively. However, with sigma 4, the performance improved to 0.8024, and the highest Dice score of 0.8090 was achieved with sigma 5. These results suggest that lower sigma values may adversely affect segmentation performance, while moderate to higher sigma values have the potential to enhance it.

The IoU value, which was 0.6807 without filtering, decreased to 0.6649 with sigma 1. After reaching 0.6701 with sigma 4, a slight increase was observed, and a value of 0.6793 was obtained with sigma 5. These results indicate that HE negatively affected the model's performance.

According to pixel accuracy results, performance dropped from 98.57% without filtering to 98.47% with sigma 1. With sigma 4 and sigma 5, values slightly improved to 98.48% and 98.57%, respectively, suggesting that certain sigma values may positively influence the model's overall accuracy.

In terms of recall, the value decreased slightly from 96.90% without filtering to 96.86% with sigma 1, and to 96.67% with sigma 5. Although a gradual decline was observed with increasing sigma values, recall remained at a high level overall.

Precision, which was 69.59% without filtering, dropped to 67.96% with sigma 1. Similarly, specificity decreased from 98.63% to 98.52%. These results suggest that lower sigma values may improve the model's ability to make correct predictions.

Table 10. Segmentation Performance of Different Gaussian Sigma Values on 128x128 Size Images of CU-HCC Dataset

SIZE	DICE	IoU	PIXEL_ACU	RECALL	PRECISION	SPECIFI
predicts 128	0,81	0,6807	0,9857	0,969	0,6959	0,9863
predicts 128 sigma 1	0,7987	0,6649	0,9847	0,9686	0,6796	0,9852
predicts 128 sigma 2	0,7868	0,6486	0,9836	0,9667	0,6634	0,9841
predicts 128 sigma 3	0,7935	0,6577	0,984	0,9769	0,6681	0,9843
predicts 128 sigma 4	0,8024	0,6701	0,9848	0,9833	0,6778	0,9849
predicts 128 sigma 5	0,809	0,6793	0,9857	0,9667	0,6955	0,9863

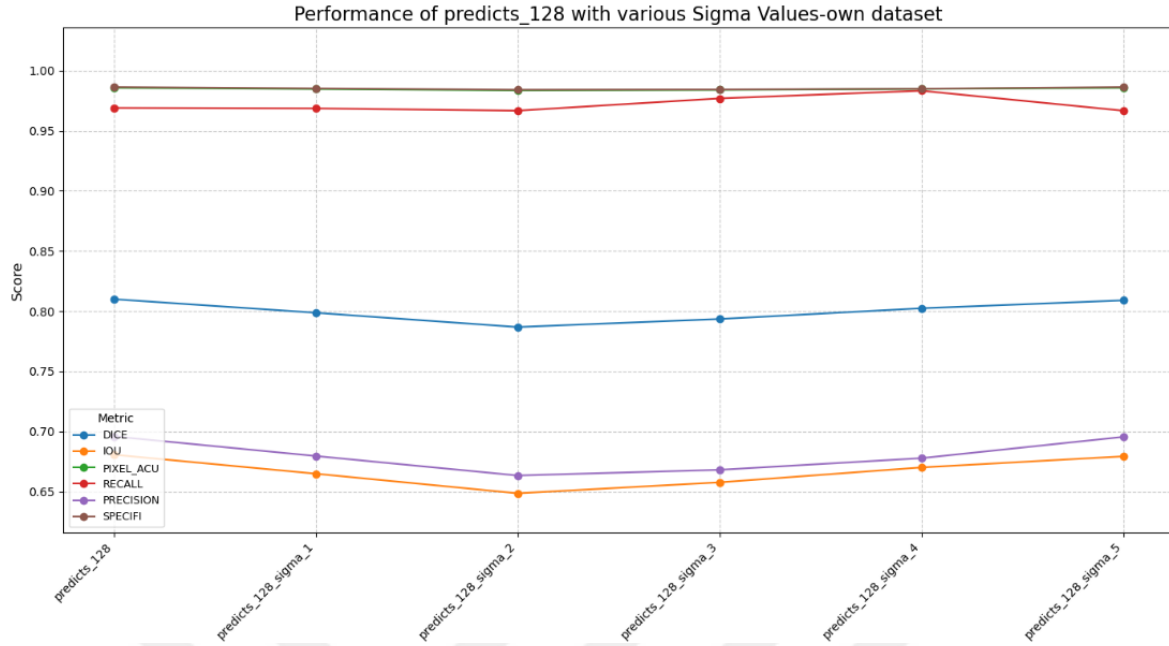


Figure 22. Segmentation Performance of Different Gaussian Sigma Values on 128x128 Size Images of CU-HCC Dataset

Table 11 and Figure 23 present the segmentation performance of 256×256 resolution images, evaluated across different Gaussian sigma values and compared with the unfiltered model. The results demonstrate the effect of the sigma parameter on model performance.

In the unfiltered model, the Dice coefficient was 0.7469, indicating relatively low initial performance. With sigma 1, this value increased significantly to 0.8632, suggesting that lower sigma values can substantially enhance segmentation accuracy. However, with sigma 5, the Dice score dropped to 0.6897, indicating a notable decline. These results suggest that while moderate sigma values may improve segmentation performance, higher sigma values may have a detrimental effect.

The IoU value increased from 0.596 without filtering to 0.7594 with sigma 1, but then declined to 0.5264 with sigma 5, indicating that higher sigma values may reduce the model's ability to correctly identify true positives.

According to the Pixel Accuracy results, the unfiltered model achieved 98.01%, which improved to 99.06% with sigma 1. However, this value decreased to 98.36% with sigma 5. These results suggest that lower sigma values can enhance the model's overall accuracy.

In terms of recall, the unfiltered model achieved 93.63%, which increased to 95.00% with sigma 1, but dropped significantly to 58.12% with sigma 5. This indicates that higher sigma values may cause the model to miss a substantial number of positive segments. When segmentation performance is evaluated in terms of precision, precision increased 62.12% in the unfiltered model to 84.82% with sigma 5.

Specificity, on the other hand, improved from 98.15% (unfiltered) to 99.19% with sigma 1. This suggests that lower sigma values reduce false negatives and enhance specificity. For sigma 5, specificity was recorded at 99.66%, indicating that higher sigma values may positively affect the model's overall specificity.

Table 11. Segmentation Performance of Different Gaussian Sigma Values on 256x256 Size Images of CU-HCC Dataset

SIZE	DICE	IoU	PIXEL_ACU	RECALL	PRECISION	SPECIFI
predicts 256	0,7469	0,596	0,9801	0,9363	0,6212	0,9815
predicts 256 sigma 1	0,8632	0,7594	0,9906	0,95	0,791	0,9919
predicts 256 sigma 2	0,805	0,6737	0,9868	0,8716	0,7479	0,9905
predicts 256 sigma 3	0,789	0,6515	0,9844	0,9289	0,6857	0,9862
predicts 256 sigma 4	0,779	0,638	0,985	0,8403	0,726	0,9897
predicts 256 sigma 5	0,6897	0,5264	0,9836	0,5812	0,8482	0,9966

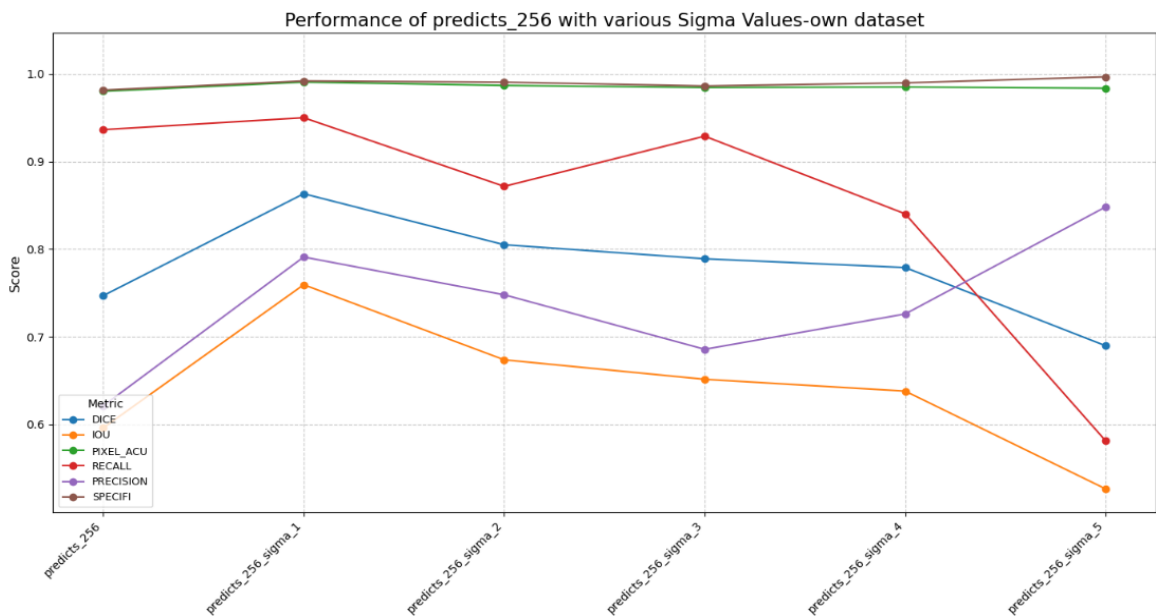


Figure 23. Segmentation Performance of Different Gaussian Sigma Values on 256x256 Size Images of CU-HCC Dataset

Table 12 and Figure 24 present the segmentation performance of 512×512 resolution images, evaluated using different Gaussian sigma values in comparison with the unfiltered model. The results demonstrate the influence of the sigma parameter on model performance.

In the unfiltered model, the Dice coefficient was calculated as 0.8318. With sigma 1, this value increased significantly to 0.9053, indicating a notable improvement in segmentation performance. For sigma 2 and sigma 3, the values were recorded as 0.9018 and 0.8844, respectively. However, with sigma 4, the Dice score dropped to 0.8281, and further declined to 0.6799 with sigma 5. These results suggest that low to moderate sigma values enhance segmentation performance, while higher sigma values may negatively impact it.

Regarding the IoU, the value without filtering was 0.712, which increased to 0.827 with sigma 1. For sigma 2 and sigma 3, the values were 0.8212 and 0.7928, respectively. However, the IoU declined to 0.7066 with sigma 4 and dropped significantly to 0.5151 with sigma 5. These results indicate that sigma 1 yielded the highest performance in this context.

According to the Pixel Accuracy results, the unfiltered model achieved 98.92%. This increased to 99.41% with sigma 1, while the values for sigma 2, 3, 4, and 5 were recorded as 99.35%, 99.27%, 98.88%, and 97.27%, respectively. These results suggest that low to moderate sigma values improved the overall accuracy of the model.

In terms of recall, the values were 90.28% for sigma 1, 94.75% for sigma 2, 88.60% for sigma 3, 85.86% for sigma 4, and 92.55% for sigma 5. It is suggested that histogram equalization enhanced the model's ability to detect positive segments.

The precision value for the unfiltered model was 81.16%, which increased to 90.78% with sigma 1, indicating that low sigma values improve the model's ability to identify positive segments. Precision was recorded as 86.04% for sigma 2 and 88.29% for sigma 3. However, with sigma 5, the value dropped to 53.73%, suggesting that higher sigma values may lead to missed detections of some positive segments.

Specificity in the unfiltered model was 99.36%, which increased to 99.70% with sigma 1. For sigma 2, 3, and 4, the values were 99.95%, 99.62%, and 99.30%, respectively. The lowest specificity was observed with sigma 5, at 97.42%. Although higher sigma values generally had a positive impact on specificity, sigma 5 appeared to cause a decline in the accuracy of negative predictions.

Overall, the results in the table clearly demonstrate that the choice of sigma value significantly affects the segmentation performance of the model. While low sigma values improve segmentation performance, high sigma values may lead to both missed positive segments and an increase in false positives. Therefore, it can be concluded that the sigma parameter must be selected carefully during noise reduction processes.

Table 12. Segmentation Performance of Different Gaussian Sigma Values on 512x512 Size Images of CU-HCC Dataset

SIZE	DICE	IoU	PIXEL_ACU	RECALL	PRECISION	SPECIFI
predicts_512	0,8318	0,712	0,9892	0,8529	0,8116	0,9936
predicts_512_sigma_1	0,9053	0,827	0,9941	0,9028	0,9078	0,997
predicts_512_sigma_2	0,9018	0,8212	0,9935	0,9475	0,8604	0,995
predicts_512_sigma_3	0,8844	0,7928	0,9927	0,886	0,8829	0,9962
predicts_512_sigma_4	0,8281	0,7066	0,9888	0,8586	0,7997	0,993
predicts_512_sigma_5	0,6799	0,5151	0,9727	0,9255	0,5373	0,9742

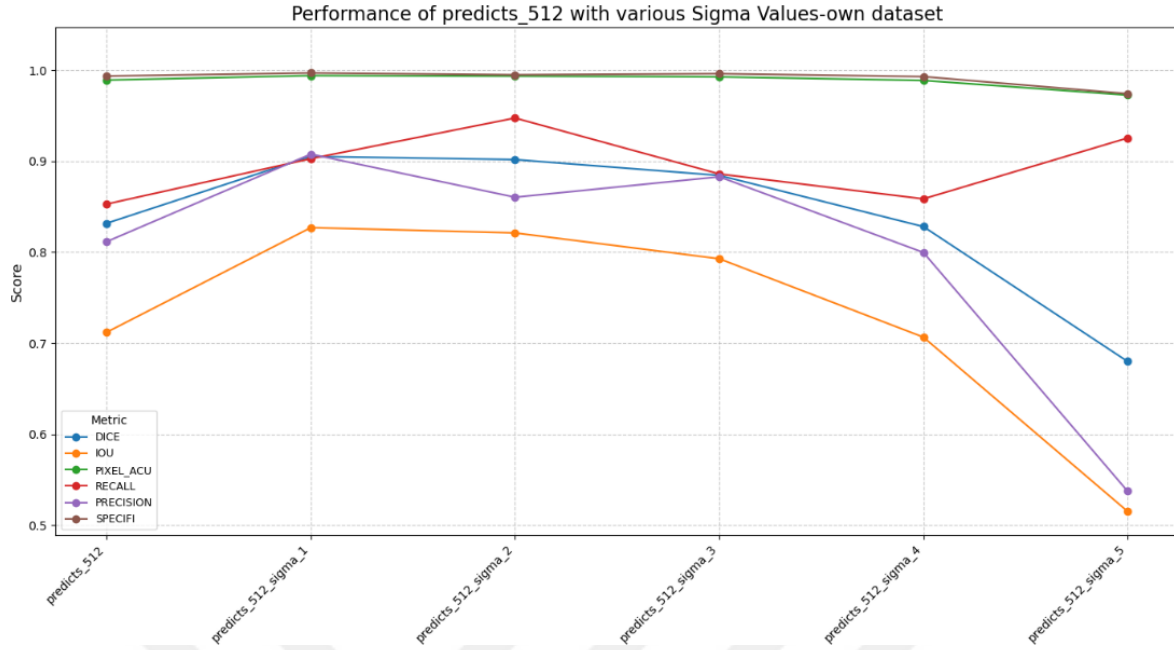


Figure 24. Segmentation Performance of Different Gaussian Sigma Values on 512x512 Size Images of CU-HCC Dataset

4.2.3. Effect of Histogram Equalization and CLAHE Methods on Segmentation Performance at Different Resolutions

Table 13 and Figure 25 present the segmentation performance of 128×128 resolution images evaluated using the unfiltered model, HE, and CLAHE preprocessing methods. The results demonstrate the impact of different preprocessing techniques on the model's segmentation performance.

For the unfiltered model, the Dice coefficient was recorded as 0.786. After applying HE, this value decreased to 0.6274, indicating that histogram equalization had a negative effect on segmentation performance. The Dice coefficient obtained using CLAHE was 0.6998, showing a partial improvement compared to HE.

The IoU value was 0.6474 in the unfiltered model, but dropped to 0.4571 following HE. With CLAHE, the IoU reached 0.5383. These results suggest that HE significantly reduced the model's ability to accurately identify true positives.

According to the Pixel Accuracy results, the accuracy dropped to 96.93% after HE, while it was 98.61% in the unfiltered model. With CLAHE, Pixel Accuracy improved to 97.81%. This indicates that HE adversely affected overall accuracy, whereas CLAHE provided a partial enhancement.

Recall was 95.51% in the unfiltered model and increased to 96.85% after HE. With CLAHE, recall reached 95.53%. These results indicate that HE may enhance the model's ability to detect positive segments.

In the unfiltered model, precision was 66.78% and specificity was 98.70%. After applying HE, precision dropped sharply to 46.40% and specificity to 96.93%. With CLAHE, precision increased to 55.22% and specificity to 97.88%.

Table 13. Segmentation Performance of Different Histogram Equalization Methods on 128x128 Images of CU-HCC Dataset

SIZE	DICE	IoU	PIXEL_ACU	RECALL	PRECISION	SPECIFI
predicts_128	0,786	0,6474	0,9861	0,9551	0,6678	0,987
predicts_128_HE	0,6274	0,4571	0,9693	0,9685	0,464	0,9693
predicts_128_CLAHE	0,6998	0,5383	0,9781	0,9553	0,5522	0,9788

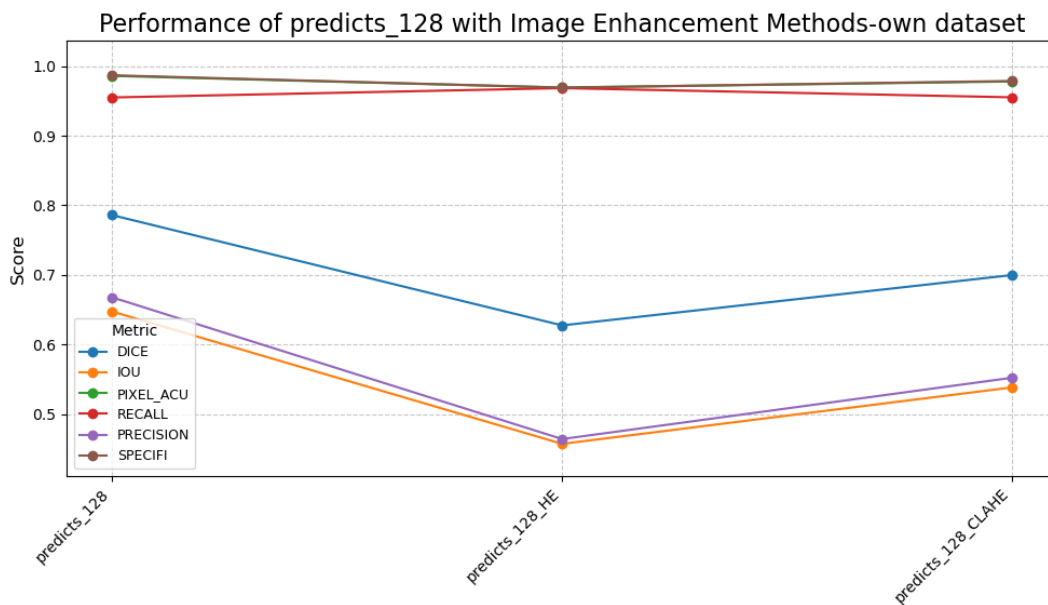


Figure 25. Segmentation Performance of Different Histogram Equalization Methods on 128x128 Images of CU-HCC Dataset

Table 14 and Figure 26 present the segmentation performance of 256×256 resolution images, evaluated using the unfiltered model along with HE and CLAHE preprocessing methods.

For the unfiltered model, the Dice coefficient was recorded as 0.7182. After applying HE, this value increased slightly to 0.7244, indicating a minor improvement in segmentation performance. However, with CLAHE, the Dice coefficient dropped to 0.6908, suggesting that HE may have the potential to improve segmentation, whereas CLAHE appeared to have a negative impact.

The IoU value was measured at 0.5603 in the unfiltered model and increased to 0.5679 with HE. In contrast, CLAHE resulted in a lower IoU value of 0.5277. This indicates that HE may partially enhance the model's ability to correctly identify positive segments, while CLAHE may have detrimental effects.

In terms of Pixel Accuracy, the unfiltered model achieved 98.05%, which rose to 98.44% with HE. With CLAHE, the accuracy decreased slightly to 97.86%. These results suggest that HE improved the overall accuracy of the model, whereas CLAHE caused a modest decline.

Recall was relatively high in the unfiltered model at 92.95%, but decreased significantly to 76.67% after HE. With CLAHE, recall reached 89.61%. These results suggest that histogram equalization may reduce recall, while CLAHE provides a less pronounced but still notable recovery.

Specificity in the unfiltered model was recorded as 98.19%. This increased to 99.04% following HE and was measured at 98.09% with CLAHE. While lower sigma values improved the model's ability to correctly predict positives and reduced false negatives, HE also had a positive effect on specificity, whereas CLAHE did not demonstrate similar effectiveness.

Table 14. Segmentation Performance of Different Histogram Equalization Methods on 256x256 Images of CU-HCC Dataset

SIZE	DICE	IoU	PIXEL_ACU	RECALL	PRECISION	SPECIFI
predicts_256	0,7182	0,5603	0,9805	0,9295	0,5852	0,9819
predicts_256_HE	0,7244	0,5679	0,9844	0,7667	0,6865	0,9904
predicts_256_CLAHE	0,6908	0,5277	0,9786	0,8961	0,5621	0,9809

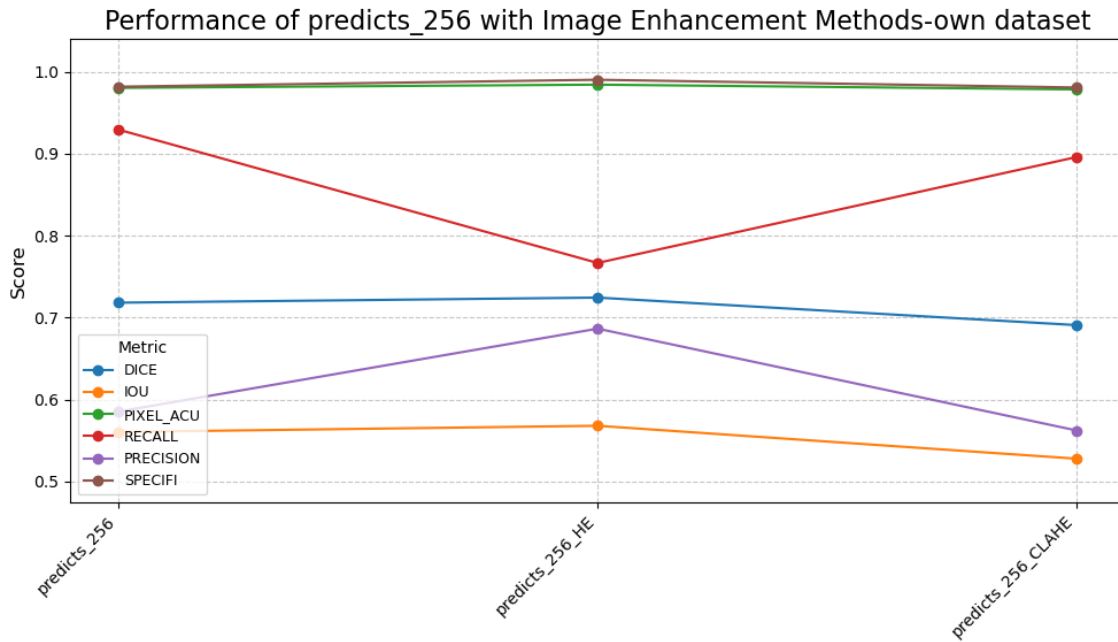


Figure 26. Segmentation Performance of Different Histogram Equalization Methods on 256x256 Images of CU-HCC Dataset

Table 15 and Figure 27 present the segmentation performance of 512×512 resolution images, evaluated using the unfiltered model as well as the HE and CLAHE preprocessing methods.

In the unfiltered model, the Dice coefficient was recorded as 0.7956. This value increased to 0.8157 following the application of HE, indicating that histogram equalization improved segmentation performance. With CLAHE, the Dice coefficient reached its highest value of 0.8927, revealing a significant enhancement in performance.

The IoU value in the unfiltered model was 0.6605, which increased to 0.6888 after HE. With CLAHE, the IoU rose to 0.8063, representing the highest value achieved. These results

suggest that while HE enhances the model's ability to identify true positives, CLAHE provides a more substantial improvement.

Pixel Accuracy was 98.86% in the unfiltered model and slightly increased to 98.90% with HE. With CLAHE, Pixel Accuracy reached a high value of 99.44%. These results indicate that both HE and CLAHE improved the overall accuracy of the model, with CLAHE producing the most significant gains.

When examining recall values, the unfiltered model achieved 83.08%. After applying HE, this value rose to 91.17%, while CLAHE resulted in 87.05%. HE enhanced the model's ability to detect positive segments, while CLAHE also proved effective in certain cases.

Precision in the unfiltered model was recorded at 76.32%, which decreased to 73.80% with HE. However, CLAHE significantly increased precision to 91.62%. These results indicate that CLAHE improves the model's ability to detect positive segments, while HE may lead to the omission of some positive instances. The variation in precision highlights the importance of selecting an appropriate preprocessing technique to improve the model's perception of the positive class.

Specificity in the unfiltered model was 99.29%, which slightly decreased to 99.11% after HE. However, with CLAHE, specificity increased to 99.78%, marking the highest value observed. This result indicates that CLAHE enhances the model's ability to reduce false negatives. Conversely, histogram equalization with low sigma values appeared to have a slightly negative effect on specificity. Overall, the consistently high specificity values demonstrate the model's effectiveness in correctly classifying negative predictions.

Table 15. Segmentation Performance of Different Histogram Equalization Methods on 512x512 Images of CU-HCC Dataset

SIZE	DICE	IoU	PIXEL_ACU	RECALL	PRECISION	SPECIFI
predicts_512	0,7956	0,6605	0,9886	0,8308	0,7632	0,9929
predicts_512 HE	0,8157	0,6888	0,989	0,9117	0,738	0,9911
predicts_512 CLAHE	0,8927	0,8063	0,9944	0,8705	0,9162	0,9978

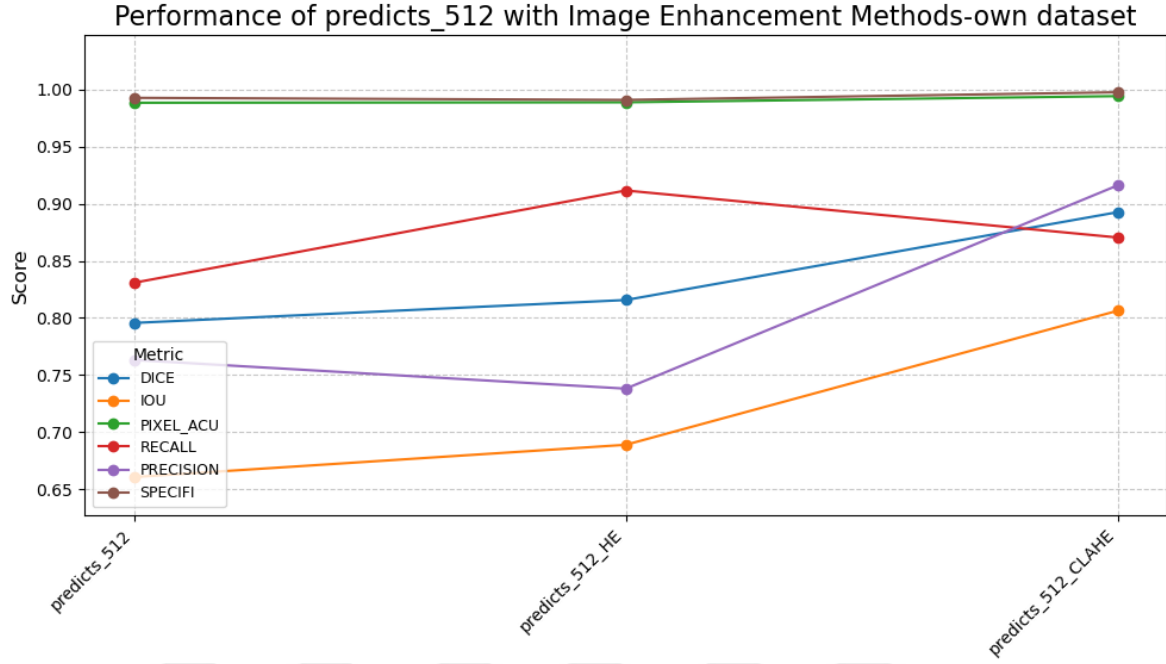


Figure 27. Segmentation Performance of Different Histogram Equalization Methods on 512x512 Images of CU-HCC Dataset

4.3. Statistical Comparison of the Obtained Results

Segmentation performance metrics obtained at three different resolutions (128×128 , 256×256 , and 512×512) were compared between the ATLAS dataset, which contains T1-weighted MRI, and CU-HCC Dataset composed of T2-weighted MRI (Figure 28). Overall, the ATLAS dataset produced more consistent and higher results in metrics such as Dice, IoU, Pixel Accuracy, and Specificity, likely due to its larger size and more balanced sampling. Notably, at the highest resolution (512×512), the Dice (0.853) and IoU (0.7437) values achieved using the ATLAS dataset were particularly striking.

In contrast, despite the limited number of T2-weighted images in CU-HCC Dataset, competitive results were achieved in certain metrics. For example, at 128×128 resolution, CU-HCC Dataset outperformed the ATLAS dataset in recall (0.969) and precision (0.6959), which reflect the model's ability to correctly detect true positives. This suggests that T2-weighted images may offer advantages in lesion contrast and can enhance the model's sensitivity even with a relatively small number of samples.

Moreover, an overall improvement in segmentation performance was observed in both datasets as the resolution increased to 512×512 . This indicates that higher resolutions enhance segmentation accuracy in both T1- and T2-weighted MRI. In particular, U-HCC Dataset demonstrated an improvement in reducing false positives, as reflected by a Specificity value of 0.9936 at the 512×512 resolution.

In conclusion, while the ATLAS dataset benefiting from a larger number of images yielded higher results in some metrics, T2-weighted dataset achieved comparable or even superior performance in others. These results highlight the significant impact of image type and contrast characteristics on model performance. It also suggests that models trained with T2-weighted MRI can produce clinically meaningful results even with a limited dataset.

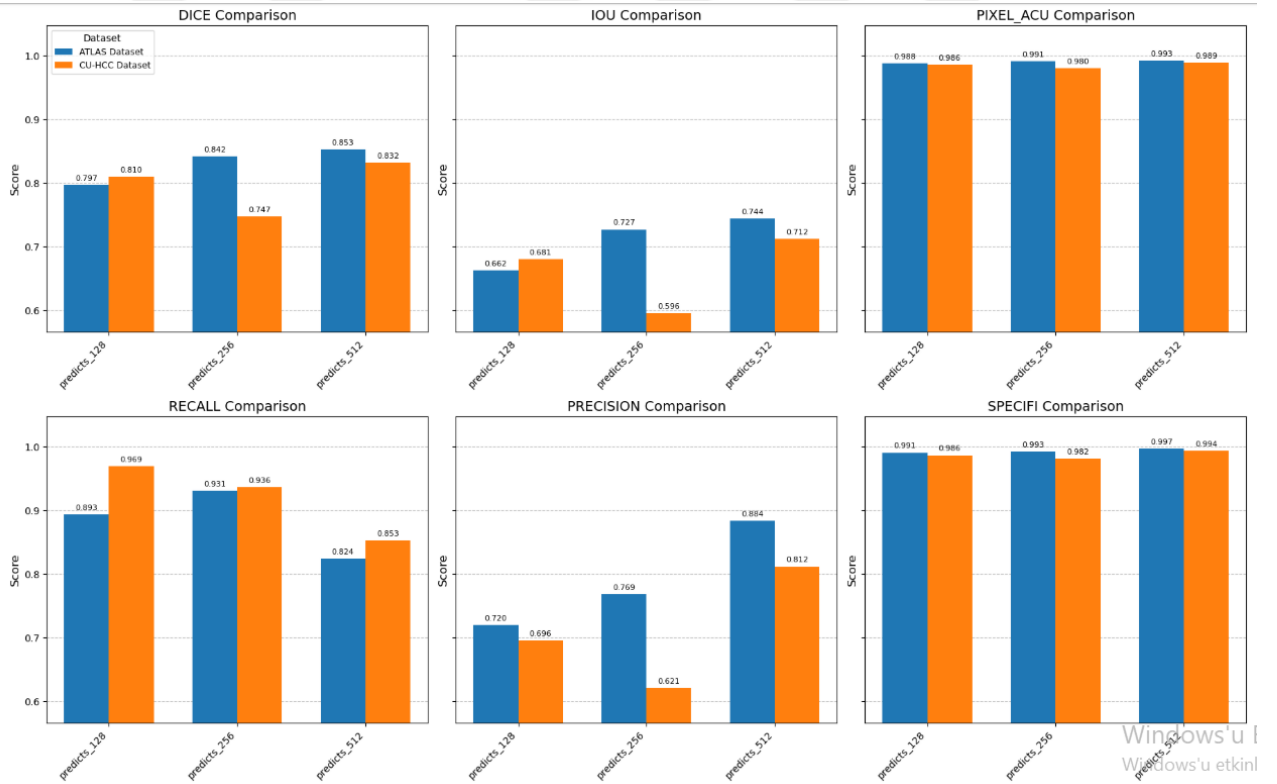


Figure 28. Comparison of segmentation performance metrics obtained on the ATLAS Dataset and CU-HCC Dataset at different image sizes

The impact of different Gaussian sigma values on segmentation performance for 128×128 resolution images was evaluated by comparing T1-weighted MRI from the ATLAS dataset with the T2-weighted images used in CU-HCC dataset (Figure 29).

In terms of Dice and IoU metrics, the Dice scores for T2-weighted images were observed to be not only higher but also more stable across different sigma values (ranging from 0.7868 to 0.8100). In contrast, T1-weighted images from the ATLAS dataset showed wider fluctuations in performance for instance, the Dice score dropped to as low as 0.7275 with sigma 2. This suggests that T2-weighted sequences offer a more stable structure for segmentation.

Regarding Pixel Accuracy, both datasets exhibited generally high accuracy levels, but the values were more consistent in T2-weighted images. Similarly, in terms of sensitivity, T2-weighted sequences demonstrated a more balanced distribution compared to T1-weighted ones. While sharp declines were observed in certain sigma values for T1-weighted data, no abrupt changes were seen in the sensitivity values for T2-weighted data.

Recall values remained high in both datasets but showed a more consistently elevated trend in T2-weighted images. In terms of specificity, both datasets demonstrated robustness against Gaussian filtering, maintaining high levels across all sigma values.

Overall, T2-weighted MRI provided more stable segmentation results, and Gaussian filtering led to more predictable and consistent performance variations within this data type. In contrast, T1-weighted images showed broader metric fluctuations, highlighting that the type of MRI sequence is a significant factor in segmentation sensitivity.

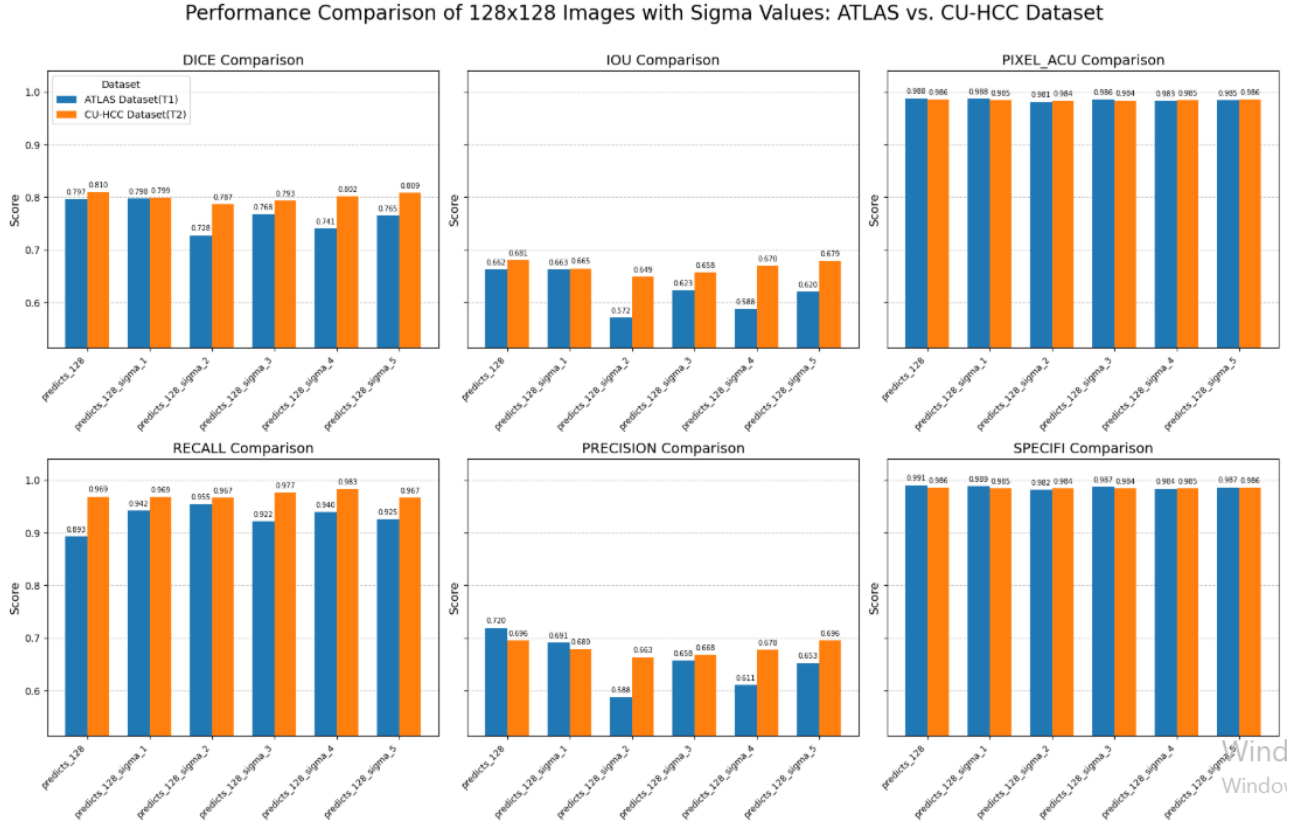


Figure 29. Effect of different Gaussian sigma values on segmentation performance on 128×128 images, comparison of the ATLAS Dataset and the CU-HCC Dataset

The effect of different Gaussian sigma values on segmentation performance at 256×256 resolution was compared using T1-weighted images from the ATLAS dataset and the T2-weighted CU-HCC dataset (Figure 30).

In terms of Dice and IoU metrics, the highest performance in the T1-weighted ATLAS dataset was achieved at sigma 1 (Dice: 0.8640). As the sigma value increased, segmentation performance remained relatively stable, and high values were maintained. In contrast, for the T2-weighted images, Dice scores peaked at sigma 1 but then showed a steady decline, reaching the lowest value at sigma 5 (Dice: 0.6897). This indicates that T2-weighted images at 256 resolution exhibit a more fragile and fluctuating response to filtering.

The Pixel Accuracy metric remained at high levels in both datasets. However, while T1-weighted data maintained stable and consistently high accuracy, T2-weighted data showed a

noticeable decline and fluctuation after sigma 1. This suggests that Gaussian filtering may adversely affect overall classification accuracy, particularly in T2-weighted images.

In terms of precision, the T1-weighted dataset produced more stable and consistently high values, with performance not deteriorating as sigma increased. In contrast, T2-weighted images showed marked fluctuations in sensitivity, peaking at sigma 5 (0.8482). This suggests that while the model captured more positive instances in T2 data, it also likely produced more false positives.

Although both datasets initially exhibited high recall values, the T2-weighted data showed a significant drop at sigma 5 (0.5812), indicating a reduced ability of the model to correctly identify true positives. Regarding specificity, both datasets demonstrated high and stable trends; however, more pronounced sigma-related effects were observed in T2-weighted images.

In conclusion, at 256×256 resolution, the T1-weighted ATLAS dataset demonstrated greater robustness and stability against Gaussian filtering, while T2-weighted CU-HCC dataset showed degradation in segmentation quality, particularly at higher sigma values. This analysis highlights that both image resolution and sequence type significantly influence the impact of Gaussian filtering on segmentation performance.

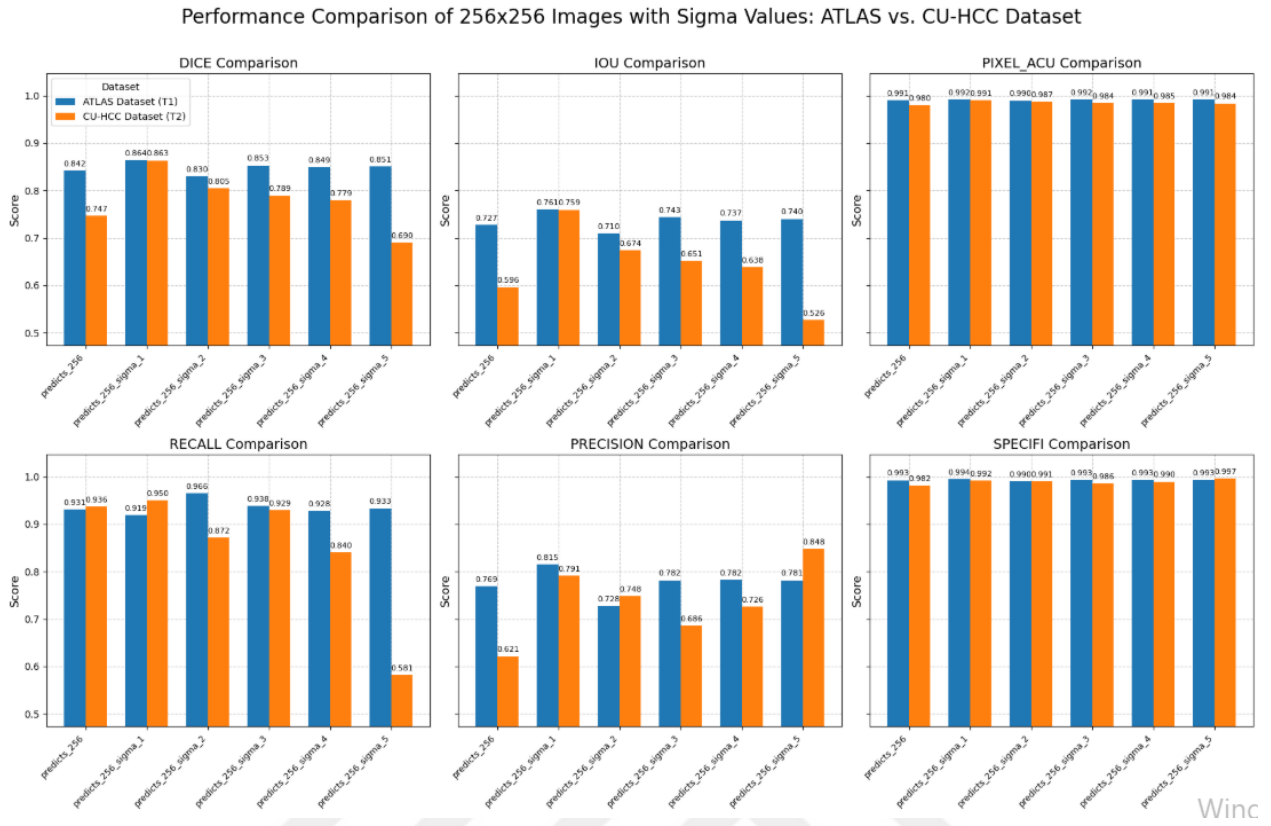


Figure 30. Effect of different Gaussian sigma values on segmentation performance on 256x256 images, comparison of the ATLAS Dataset and the CU-HCC Dataset

The effect of different Gaussian sigma values on segmentation performance at 512×512 resolution was evaluated by comparing T1-weighted images from the ATLAS dataset with the T2-weighted images used in CU-HCC dataset (Figure 31). In terms of Dice and IoU, the highest performance in the T1-weighted ATLAS data was achieved at sigma 3 (Dice: 0.8885, IoU: 0.7994). However, performance declined markedly at higher sigma values (e.g., Dice dropped to 0.7811 at sigma 4). In contrast, the T2-weighted images reached peak Dice scores around sigma 1 and sigma 2 (Dice: 0.9053 and 0.9018, respectively), followed by a downward trend. This suggests that T2-weighted images respond with a more pronounced performance improvement to filtering, although this effect is limited to a narrow sigma range.

Pixel Accuracy remained high across both datasets. In T2-weighted data, the highest value (0.9941) was achieved at sigma 1, and in general, filtering contributed positively to overall accuracy. In the T1-weighted dataset, values remained largely stable, with only a slight drop observed at sigma 4 (0.9896).

Regarding precision and sensitivity, T2-weighted images produced high scores at sigma 1 and 2 (e.g., sensitivity: 0.9078), indicating that Gaussian filtering at moderate sigma values helped reduce false positives and improved detection accuracy. In the T1-weighted dataset, these metrics fluctuated more noticeably, particularly declining at higher sigma levels. Both datasets achieved high and balanced recall values. However, T2-weighted data maintained this metric more consistently, while the T1-weighted dataset showed greater variability as sigma increased.

Specificity remained high in both datasets, though it was more stable in the T1-weighted ATLAS data. In the T2-weighted images, lower specificity was observed at small sigma values but gradually recovered toward sigma 5.

In conclusion, Gaussian filtering generally had a positive impact on segmentation performance in T2-weighted images, with the most pronounced effects observed at sigma values between 1 and 2. In contrast, the T1-weighted ATLAS data showed a more fluctuating response, with performance degradation at higher sigma levels. This comparison reinforces the notion that the MRI sequence type directly influences how segmentation models respond to filtering.

Performance Comparison of 512x512 Images with Sigma Values: ATLAS vs. CU-HCC Dataset

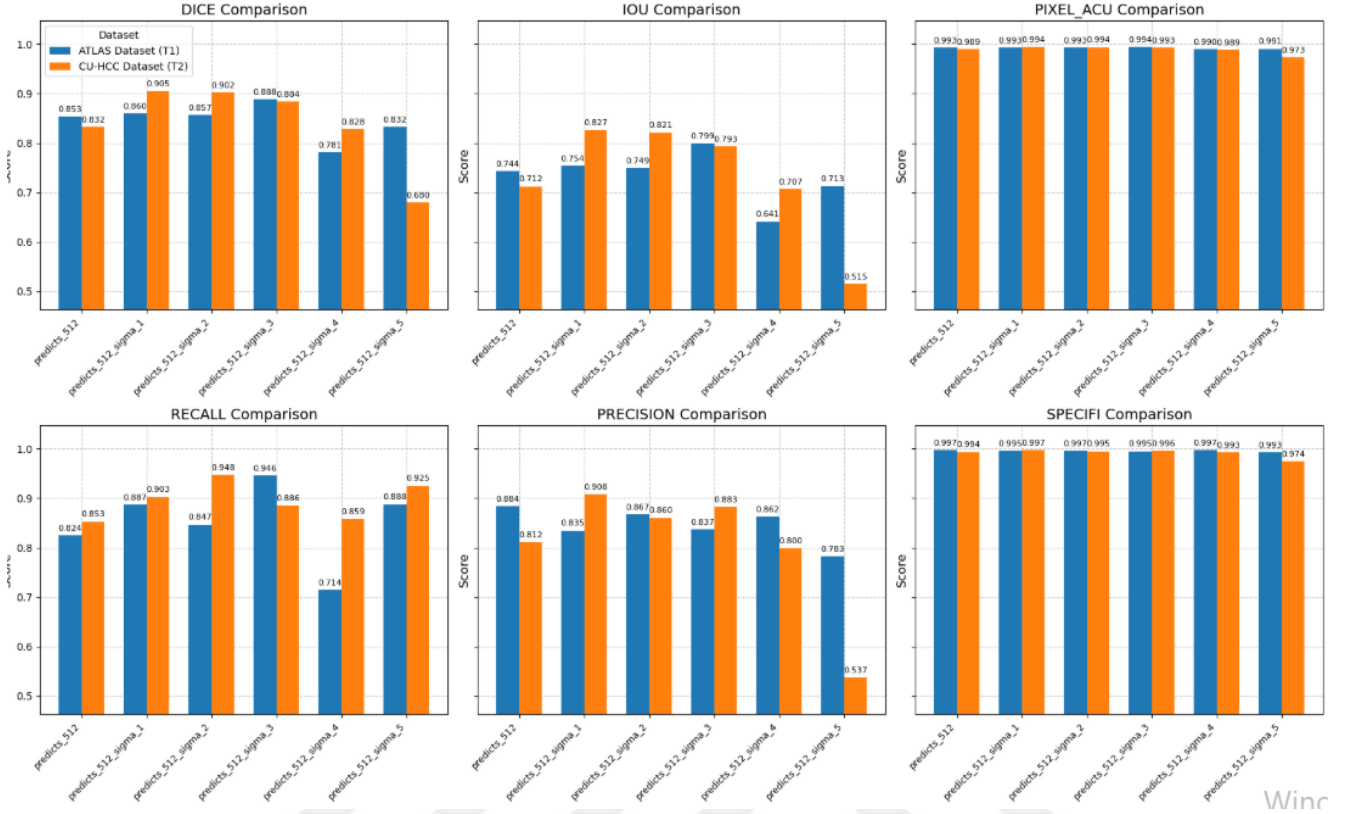


Figure 31. Effect of different Gaussian sigma values on segmentation performance on 512x512 images, comparison of the ATLAS Dataset and the CU-HCC Dataset

The effect of different histogram equalization methods on 128×128 resolution images was evaluated using T1-weighted images from the ATLAS dataset and T2-weighted images from CU-HCC dataset (Figure 32).

In terms of Dice and IoU metrics, the highest performance in both datasets was achieved using raw images without any histogram equalization (T1 Dice: 0.7970; T2 Dice: 0.7860). When histogram equalization was applied, the HE method notably reduced segmentation performance in both datasets. While the CLAHE method resulted in less performance degradation compared to HE in T1 data, it still yielded lower Dice scores than the unprocessed data in the T2 dataset, despite a slight improvement over HE. These results suggest that histogram equalization may reduce the model's learning effectiveness in low-resolution images.

Pixel Accuracy showed a declining trend in both datasets following histogram equalization, particularly with the HE method, indicating a negative impact on overall classification accuracy.

In terms of sensitivity, different patterns were observed: HE caused a substantial decrease in T1 data, whereas CLAHE produced better sensitivity results in the T2 dataset.

Recall values increased in both datasets after histogram equalization. This indicates that the models became more permissive in identifying positive classes, though potentially at the expense of accuracy and precision.

Specificity showed a slight decrease in both datasets following histogram equalization, suggesting a modest loss in the ability to correctly identify negative classes.

In conclusion, histogram equalization generally had a negative impact on segmentation performance, particularly in low-resolution (128×128) images. In T1-weighted ATLAS data, HE led to the most pronounced performance decline, while in T2-weighted images, CLAHE produced relatively more balanced results. These results indicate that histogram equalization does not universally benefit all image types or resolutions and should be applied cautiously based on the characteristics of the dataset.

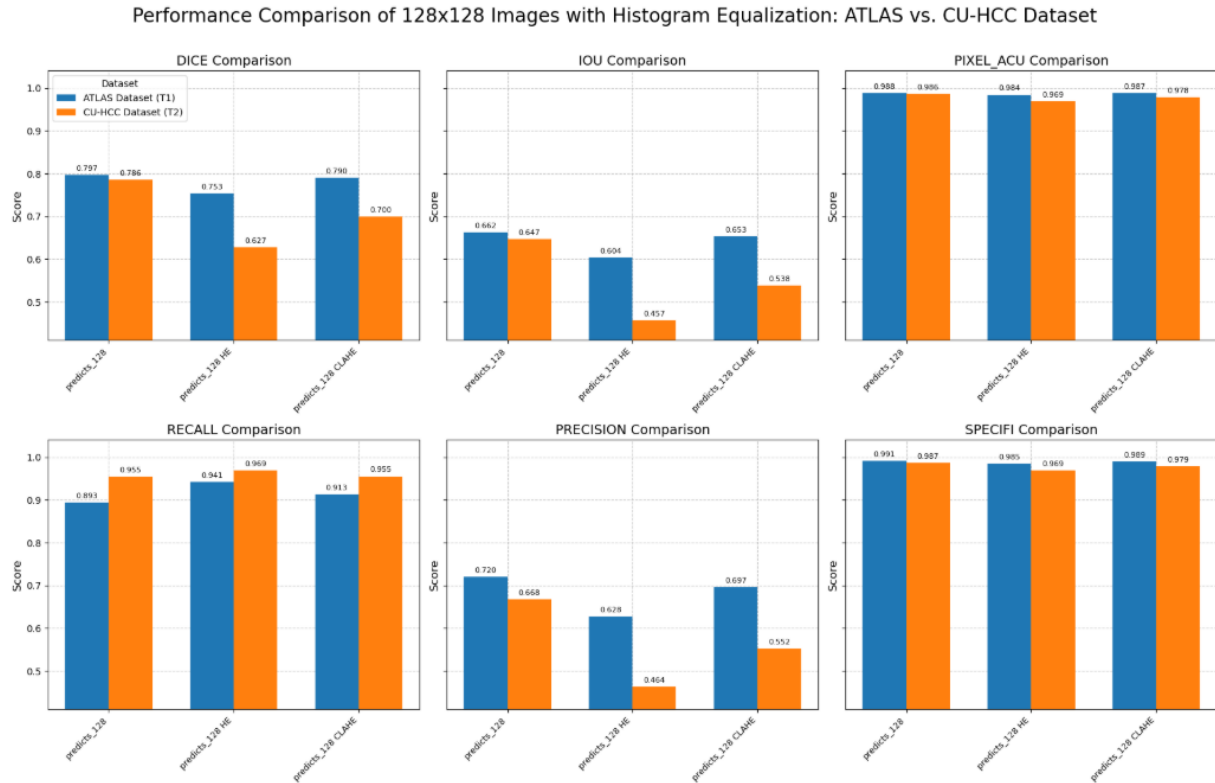


Figure 32. Effect of different histogram equalization methods on segmentation performance on 128x128 images, comparison of the ATLAS Dataset with the images used in CU-HCC Dataset

The impact of different histogram equalization methods on segmentation performance was evaluated using two datasets with 256×256 resolution (Figure 33): the ATLAS dataset (T1-weighted images) and the CU-HCC dataset (T2-weighted images). Performance was assessed using several metrics, including Dice, IoU, Pixel Accuracy, Recall, and Specificity.

In terms of Dice and IoU scores, both HE and CLAHE improved performance in the ATLAS dataset, with the highest results obtained using CLAHE (Dice: 0.8651; IoU: 0.7623). In contrast, the CU-HCC dataset achieved its highest Dice and IoU values with the HE method (Dice: 0.7244; IoU: 0.5679), while CLAHE led to a decline in performance, falling below even the raw image results. This discrepancy suggests that histogram equalization may enhance segmentation accuracy in T1-weighted images but does not necessarily generalize to T2-weighted images.

Pixel Accuracy improved in both datasets after histogram equalization. The highest accuracy in the ATLAS dataset was achieved with CLAHE (0.9928), while in the CU-HCC dataset, HE yielded the best result (0.9844). This indicates that histogram equalization, particularly CLAHE, may enhance pixel-level classification accuracy in T1-weighted images.

Recall values remained high across all configurations, with raw images yielding the best results in both datasets (ATLAS: 0.9307; CU-HCC: 0.9295). Histogram equalization generally reduced recall, especially in the CU-HCC dataset where HE caused a substantial drop (0.7667), suggesting reduced sensitivity in detecting positive cases.

Regarding specificity, slight increases were observed in both datasets following histogram equalization. The highest specificity was achieved using CLAHE in the ATLAS dataset (0.9956) and HE in the CU-HCC dataset (0.9904), indicating modest improvements in correctly identifying negative classes.

In conclusion, histogram equalization techniques particularly CLAHE improved segmentation performance across nearly all metrics in the ATLAS dataset. This highlights the effectiveness of such methods for T1-weighted images. On the other hand, results in the T2-weighted CU-HCC dataset were more inconsistent; in this case, HE outperformed CLAHE. These results emphasize the importance of carefully considering data characteristics such as image type, contrast level, and resolution when selecting preprocessing techniques for medical image segmentation.

Performance Comparison of 256x256 Images with Histogram Equalization: ATLAS vs. CU-HCC Dataset

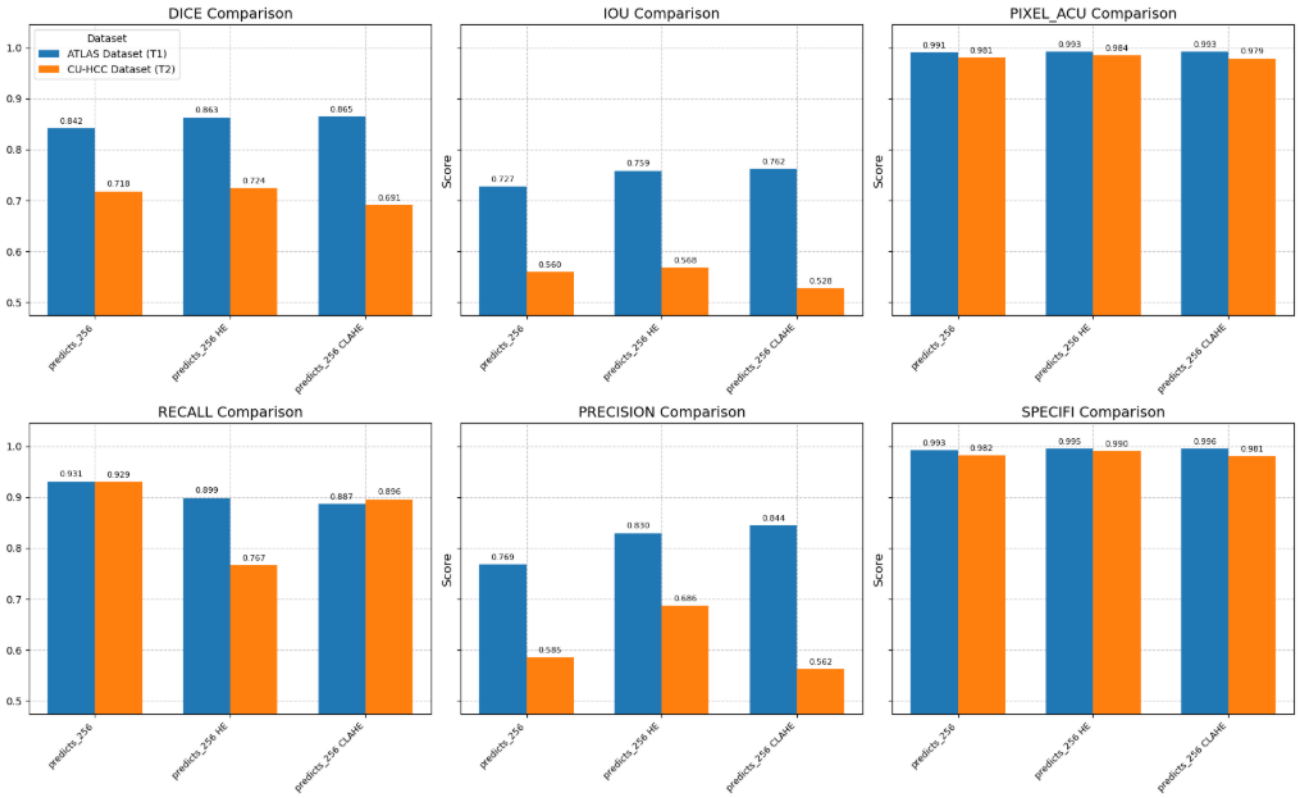


Figure 33. Effect of different histogram equalization methods on segmentation performance on 256x256 images, comparison of ATLAS Dataset with the images used in CU-HCC Dataset

The impact of different histogram equalization methods on segmentation performance was evaluated using the 512×512 resolution ATLAS dataset (T1-weighted) and the CU-HCC dataset (T2-weighted) (Figure 34). The evaluation was conducted using metrics such as Dice, IoU, Pixel Accuracy, Recall, and Specificity.

In terms of Dice and IoU scores, CLAHE provided a clear improvement in both datasets. In the ATLAS dataset, the highest values were achieved using CLAHE (Dice: 0.8892; IoU: 0.8005), and similarly, in the CU-HCC dataset, CLAHE yielded the best performance (Dice: 0.8927; IoU: 0.8063). The fact that CLAHE outperformed both raw and HE-processed images across both datasets suggests that it offers strong generalizability at high resolutions.

Pixel Accuracy followed a similar trend, with CLAHE achieving the highest accuracy in both datasets (0.9944). This indicates that CLAHE effectively enhances pixel-level segmentation accuracy regardless of image type.

Recall results exhibited more variability. While the ATLAS dataset achieved its highest recall with raw images (0.8244), CLAHE still demonstrated strong performance (0.8676). In the CU-HCC dataset, the highest recall was achieved with HE (0.9117), suggesting that HE tends to detect more positive instances though possibly at the expense of increased false positives.

Specificity remained high under all conditions. CLAHE achieved the highest specificity in both datasets (ATLAS: 0.9978; CU-HCC: 0.9978), highlighting its effectiveness in reducing false positives. These results indicate that CLAHE not only performs well in identifying positive segments but also provides strong discrimination for negative classes.

In conclusion, the 512×512 resolution evaluation showed that CLAHE clearly enhanced segmentation performance across nearly all metrics in both datasets. Although HE improved recall in the T2-weighted CU-HCC dataset, it fell short of CLAHE in terms of precision and overall balance. These results underscore the importance of considering data characteristics such as image type, contrast structure, and resolution when selecting preprocessing techniques. CLAHE emerges as a highly effective method for improving segmentation performance in both T1- and T2-weighted medical images.



Figure 34. Effect of different histogram equalization methods on segmentation performance on 512x512 images, comparison of ATLAS Dataset with the images used in CU-HCC Dataset

Table 16 presents the means and standard deviations of segmentation performance metrics across different image sizes (128, 256, 512) and Gaussian filter levels (sigma 1–5). Overall, a notable improvement in key metrics such as Dice and IoU was observed as image resolution increased particularly at 512 pixels. In contrast, increasing the Gaussian filter level led to a decline in performance across all resolutions. The highest specificity values were achieved with the 128_1 combination, while the best overall segmentation performance was recorded with the 512_1 and 512_2 combinations. These results suggest that high resolution and low filter levels are critical for improving segmentation accuracy.

Table 16. Averages of Performance Metrics by Size and Filter

Size_Filter (sigma)	DICE (Mean ± SD)	IoU (Mean ± SD)	PIXEL_ACCU (Mean ± SD)	RECALL (Mean ± SD)	F-SCORE (Mean ± SD)	PRECISIO (Mean ± SD)	SPECIFICITY (Mean ± SD)
128_1	0.64 ± 0.18	0.49 ± 0.18	0.9881 ± 0.0074	0.72 ± 0.42	0.64 ± 0.18	0.51 ± 0.17	0.989 ± 0.008
128_2	0.59 ± 0.22	0.45 ± 0.21	0.9871 ± 0.0079	0.79 ± 0.35	0.59 ± 0.22	0.47 ± 0.20	0.940 ± 0.213
128_3	0.64 ± 0.18	0.49 ± 0.18	0.9876 ± 0.0079	0.74 ± 0.42	0.63 ± 0.18	0.50 ± 0.18	0.988 ± 0.008
128_4	0.66 ± 0.15	0.51 ± 0.16	0.9886 ± 0.0075	0.73 ± 0.39	0.66 ± 0.15	0.53 ± 0.15	0.989 ± 0.008
128_5	0.57 ± 0.28	0.44 ± 0.25	0.9882 ± 0.0073	0.79 ± 0.35	0.57 ± 0.28	0.48 ± 0.25	0.870 ± 0.324
256_1	0.69 ± 0.23	0.56 ± 0.24	0.9923 ± 0.0065	0.81 ± 0.28	0.69 ± 0.23	0.64 ± 0.21	0.970 ± 0.153
256_2	0.50 ± 0.32	0.39 ± 0.29	0.9888 ± 0.0081	0.58 ± 0.38	0.50 ± 0.32	0.49 ± 0.31	0.850 ± 0.351
256_3	0.49 ± 0.31	0.38 ± 0.28	0.9841 ± 0.0115	0.66 ± 0.36	0.49 ± 0.31	0.41 ± 0.30	0.867 ± 0.323
256_4	0.47 ± 0.32	0.36 ± 0.27	0.9867 ± 0.0084	0.57 ± 0.36	0.47 ± 0.32	0.45 ± 0.32	0.777 ± 0.411
256_5	0.37 ± 0.31	0.27 ± 0.25	0.9883 ± 0.0112	0.31 ± 0.29	0.37 ± 0.31	0.55 ± 0.39	0.712 ± 0.456
512_1	0.73 ± 0.28	0.63 ± 0.28	0.9953 ± 0.0048	0.71 ± 0.30	0.73 ± 0.28	0.80 ± 0.24	0.950 ± 0.215
512_2	0.77 ± 0.23	0.66 ± 0.24	0.9948 ± 0.0049	0.84 ± 0.26	0.77 ± 0.23	0.72 ± 0.21	0.972 ± 0.154
512_3	0.67 ± 0.32	0.57 ± 0.30	0.9938 ± 0.0054	0.70 ± 0.34	0.67 ± 0.32	0.68 ± 0.31	0.902 ± 0.296
512_4	0.56 ± 0.30	0.44 ± 0.27	0.9901 ± 0.0072	0.62 ± 0.34	0.56 ± 0.30	0.56 ± 0.31	0.875 ± 0.325
512_5	0.40 ± 0.28	0.29 ± 0.24	0.9700 ± 0.0152	0.77 ± 0.31	0.40 ± 0.28	0.30 ± 0.25	0.901 ± 0.253

As shown in Table 17, both image size and filter level had statistically significant effects on key segmentation metrics such as Dice, IoU, Recall, and F-Score ($p < 0.001$). The largest effect sizes were observed for the filter factor in the IoU and Pixel Accuracy metrics ($\eta^2 = 0.683$ and $\eta^2 = 0.655$, respectively). The interaction between size and filter was also found to be significant for most metrics, demonstrating medium-to-high effect sizes. For the Specificity metric, both size and filter factors had significant effects, though their effect sizes were relatively lower compared to other metrics ($\eta^2 = 0.185$ and $\eta^2 = 0.166$). These results indicate that both resolution and filter level play an important combined role in determining segmentation performance.

In the case of the Pixel Accuracy metric, the effect of the size factor was not statistically significant ($p > 0.05$). This suggests that variations in image size had a limited influence on overall pixel-level accuracy. One possible explanation is that segmentation images predominantly consist of background pixels, and thus increasing image size may not lead to meaningful changes in overall accuracy. On the other hand, the filter factor and the size $\times$ filter interaction showed statistically significant differences ($p < 0.001$), indicating that model performance varied more substantially based on the type of filtering applied.

Table 17. Repeated Measures ANOVA Results: F, p, and η^2 Values for Size, Filter, and Size×Filter Interaction

Metric	Size F(df)	Size p	Size η^2	Filter F(df)	Filter p	Filter η^2	Interaction F(df)	Interaction p	Interaction η^2
DICE	25.053 (2,82)	< 0.001	0.379	66.718 (4,164)	< 0.001	0.619	17.696 (8,328)	< 0.001	0.301
IoU	31.625 (2,82)	< 0.001	0.435	88.435 (4,164)	< 0.001	0.683	25.936 (8,328)	< 0.001	0.387
PIXEL_ACC U	0.847 (2,82)	0.432	0.020	77.714 (4,164)	< 0.001	0.655	54.855 (8,328)	< 0.001	0.572
RECALL	15.417 (2,82)	< 0.001	0.273	11.827 (4,164)	< 0.001	0.224	13.768 (8,328)	< 0.001	0.251
F-SCORE	25.053 (2,82)	< 0.001	0.379	66.718 (4,164)	< 0.001	0.619	17.696 (8,328)	< 0.001	0.301
Sensitivity	17.152 (2,82)	< 0.001	0.295	32.812 (4,164)	< 0.001	0.445	31.310 (8,328)	< 0.001	0.433
Specificity	9.33 (1.52,62.4)	0.001	0.185	8.19 (2.21,90.8)	< 0.001	0.166	3.41 (3.86,158.3)	0.011	0.077
Repeated Measures ANOVA. F values are shown with degrees of freedom in parentheses. η^2 value indicates effect size. $p < 0.05$ was considered significant.									

Dice Score Results

Effect of Image Size:

Image size had a statistically significant and moderately strong effect on the Dice score ($F(2, 82) = 25.053$, $p < 0.001$, $\eta^2 = 0.379$). Dice scores increased markedly with image resolution—from 128×128 to 256×256 and further to 512×512 . The highest Dice scores were observed at 512×512 resolution, indicating that higher resolution significantly improves segmentation accuracy. The general trend shows a consistent increase in Dice performance as image size increases ($128 \rightarrow 256 \rightarrow 512$), with 512 yielding the most pronounced enhancement.

Effect of Gaussian Filtering:

Gaussian filtering had a statistically significant and very strong effect on the Dice score ($F(4, 164) = 66.718$, $p < 0.001$, $\eta^2 = 0.619$). Significant differences were observed between filter

levels: the lowest sigma value (sigma 1) resulted in the highest Dice scores, while the highest sigma value (sigma 5) led to the lowest performance. A clear decreasing trend in Dice scores was observed as the sigma level increased. These results highlight the importance of using lower sigma values to maximize segmentation accuracy.

Interaction Between Size and Filter:

The interaction effect of image size and filter level on the Dice score was also statistically significant and demonstrated a moderately strong influence ($F(8, 328) = 17.696$, $p < 0.001$, $\eta^2 = 0.301$). In other words, the effect of image size on Dice scores varied depending on the applied filter level. For example, the highest Dice scores were achieved at 512×512 resolution with a sigma value of 2, while at 256×256 and 128×128, sigma 5 resulted in a marked drop in performance. This finding emphasizes that optimizing both image size and filter level is critical for achieving high segmentation accuracy.

The highest Dice scores were achieved with large image size (512) combined with low sigma values, particularly sigma 1 and 2. In contrast, the lowest scores were observed with small or medium image sizes and a high sigma value sigma 5). These results highlight that the combination of high resolution and appropriate sigma level is critically important for optimal segmentation performance (Figure 35).

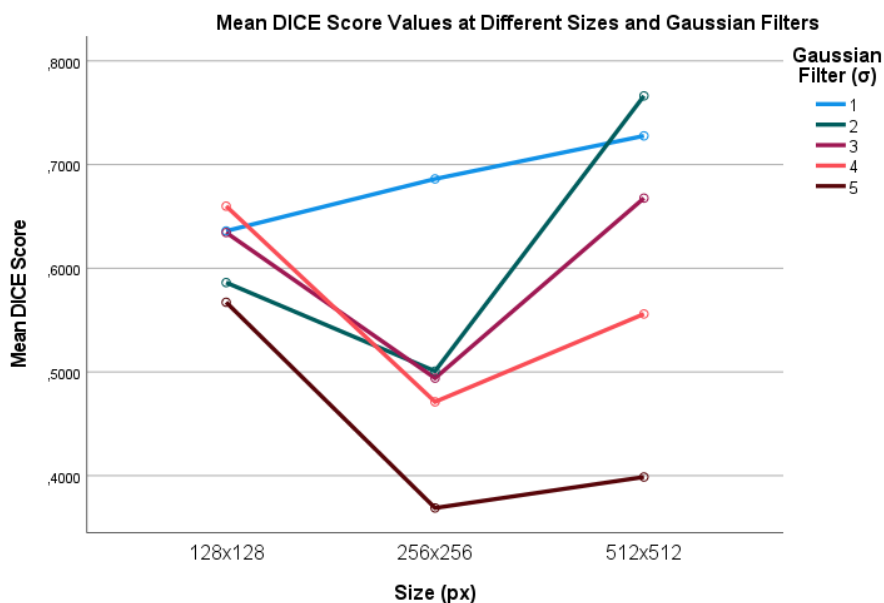


Figure 35. Graph of DICE Score Average Values in Different Dimensions and Filters

The effects of different image sizes (128×128, 256×256, 512×512) and Gaussian filter levels (sigma 1–5) on Dice scores were evaluated. The analysis was conducted using repeated measures ANOVA (GLM), examining both main effects and the interaction between image size and filter level. Descriptive statistics indicated that, for 128×128 resolution, Dice score means ranged between 0.567 and 0.660. For 256×256 resolution, the means ranged from 0.369 to 0.686. At 512×512 resolution, the highest Dice score was recorded as 0.766. Overall, there was a downward trend in Dice scores as the sigma level increased.

Upon examining the multivariate test results, image size was found to have a significant effect on Dice scores (Wilks' Lambda= 0.414, $F(2, 40) = 28.321$, $p < 0.001$, $\eta^2 = 0.586$). Similarly, Gaussian filter levels also significantly influenced Dice scores (Wilks' Lambda= 0.232, $F(4, 38) = 31.513$, $p < 0.001$, $\eta^2 = 0.768$). In addition, the interaction between image size and Gaussian filter level was statistically significant (Wilks' Lambda= 0.235, $F(8, 34) = 13.844$, $p < 0.001$, $\eta^2 = 0.765$). These results indicate that Dice scores are influenced not only by image size or filter level individually, but also by the combination of both factors.

The results of Mauchly's test of sphericity indicated that the assumption was met for the image size factor ($W = 0.943$, $p = 0.309$), but violated for the filter factor ($W = 0.221$, $p < 0.001$) and the size × filter interaction ($W = 0.037$, $p < 0.001$). Therefore, Greenhouse–Geisser corrections were applied where necessary.

According to the within-subjects effects table, the main effect of image size was statistically significant ($F(2, 82) = 25.053$, $p < 0.001$, $\eta^2 = 0.379$). A similarly significant main effect was observed for the Gaussian filter factor ($F(4, 164) = 66.718$, $p < 0.001$, $\eta^2 = 0.619$). The interaction between image size and Gaussian filter level was also found to be strong and statistically significant ($F(8, 328) = 17.696$, $p < 0.001$, $\eta^2 = 0.301$). These results indicate that combinations of filter levels and image sizes produce distinct differences in Dice scores.

Post-hoc comparisons with Bonferroni correction revealed a statistically significant difference in Dice scores between the 128×128 and 256×256 image sizes ($p < 0.001$), while no significant difference was found between 128×128 and 512×512 ($p = 1.000$). Regarding filter levels, significant differences were identified between all pairwise comparisons. In particular, Dice scores significantly decreased as the filter level increased for example, the difference between sigma 1 and sigma 5 was highly significant ($p < 0.001$).

IOU Score Results

Effect of Image Size

Image size had a statistically significant and strong effect on the IoU score ($F(2, 82) = 31.625$, $p < 0.001$, $\eta^2 = 0.435$). As image resolution increased from 128×128 to 256×256 and then to 512×512 , a notable rise in IoU scores was observed. In particular, the 512×512 resolution consistently yielded the highest average IoU values. This finding demonstrates that higher resolution plays a key role in enhancing segmentation accuracy. The progression from 128 to 256 to 512 showed clear improvements in IoU performance, reinforcing the positive impact of larger image sizes on segmentation outcomes.

Effect of Filtering

One-way ANOVA results revealed that Gaussian filtering had a statistically significant and very strong effect on IoU scores ($F(4, 164) = 88.435$, $p < 0.001$, $\eta^2 = 0.683$). Clear differences were observed among the filters: the lowest sigma level (filter 1) yielded the highest IoU scores, while sigma 5 resulted in a significant performance decline. Overall, a consistent downward trend in IoU scores was observed as the filter level increased from 1 to 5. These results indicate that lower sigma values are more effective in improving segmentation accuracy.

Interaction Between Size and Filter

When evaluating the interaction effect between image size and filter level (sigma) on IoU scores, the analysis yielded statistically significant results with moderate-to-strong effect size ($F(8, 328) = 25.936$, $p < 0.001$, $\eta^2 = 0.387$). This indicates that the effect of image size on IoU scores varies depending on the sigma value used. In other words, the relationship between resolution and segmentation accuracy is influenced by the level of filtering applied. For example, the combination of 512×512 resolution and sigma 2 resulted in the highest IoU scores, while the lowest performance was observed at sigma 5 for both 256×256 and 512×512 resolutions. These results demonstrate that the filter parameter plays a significant role in segmentation performance, particularly when considered in conjunction with image resolution.

The highest IoU scores were achieved with large-sized images (512×512) combined with low sigma values, particularly sigma 1 and 2. In contrast, the lowest performance was observed

with medium-sized images (256×256) in combination with a high sigma value (sigma 5) (Figure 36). These results highlight the critical importance of selecting appropriate filter parameters specifically lower sigma values along side high-resolution images to maximize segmentation performance.

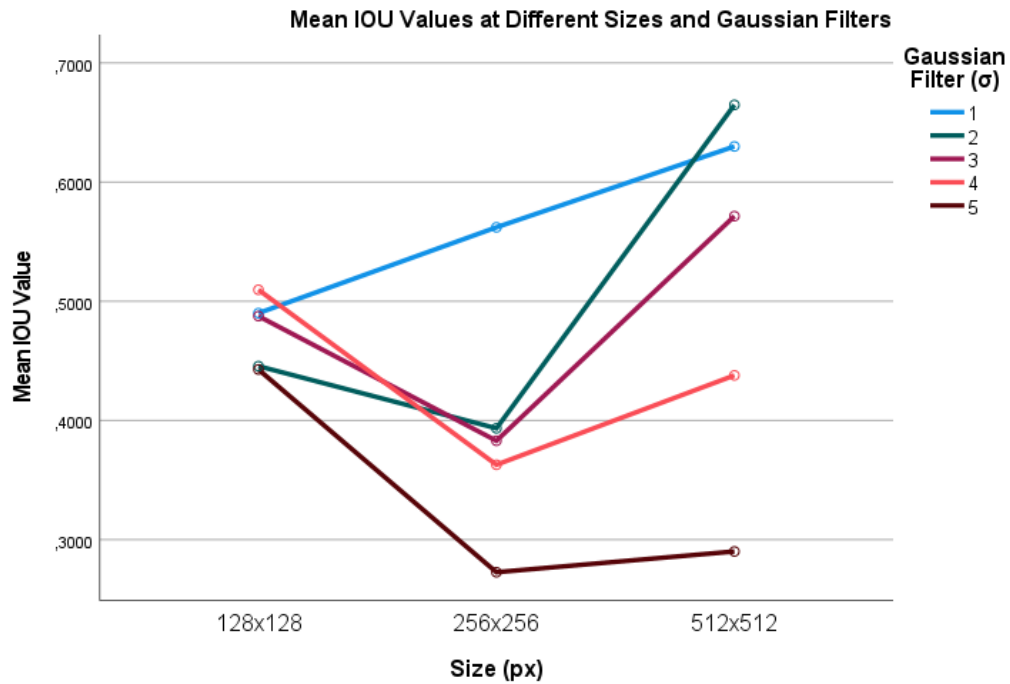


Figure 36. Graph of IoU Average Values in Different Dimensions and Filters

The effects of different image sizes (128×128, 256×256, 512×512 pixels) and Gaussian filter levels (sigma 1–5) on IoU scores were examined. A repeated measures analysis of variance (GLM) was conducted to assess the individual effects of image size and Gaussian filter level, as well as their interaction. Descriptive statistics showed that, for the 128×128 resolution, IoU means ranged from 0.443 to 0.510, while for 256×256 resolution, the means ranged from 0.273 to 0.562. At 512×512 resolution, the highest IoU score was recorded as 0.665, with particularly high scores observed at lower Gaussian filter levels.

Multivariate tests indicated that image size had a significant effect on IoU scores (Wilks' Lambda= 0.331, $F(2, 40)= 40.339$, $p < 0.001$, $\eta^2= 0.669$). A very strong main effect was also observed for the Gaussian filter factor (Wilks' Lambda= 0.166, $F(4, 38)= 47.606$, $p < 0.001$,

$\eta^2 = 0.834$). In addition, the interaction between image size and Gaussian filter level was found to be statistically significant (Wilks' Lambda = 0.177, $F(8, 34) = 19.821$, $p < 0.001$, $\eta^2 = 0.823$). These results indicate that IoU scores are influenced not only by image size or filter level independently, but also by the interaction between these two factors.

According to Mauchly's test of sphericity, the assumption of sphericity was met for the image size factor ($W = 0.939$, $p = 0.283$), but violated for the Gaussian filter factor ($W = 0.290$, $p < 0.001$) and the size $\times$ filter interaction ($W = 0.035$, $p < 0.001$). Therefore, Greenhouse–Geisser corrections were applied in the analyses.

The within-subjects effects analysis showed that the main effect of image size was statistically significant ($F(2, 82) = 31.625$, $p < 0.001$, $\eta^2 = 0.435$). A significant main effect was also observed for the Gaussian filter levels ($F(4, 164) = 88.435$, $p < 0.001$, $\eta^2 = 0.683$). Additionally, the interaction effect between image size and Gaussian filter level was found to be significant ($F(8, 328) = 25.936$, $p < 0.001$, $\eta^2 = 0.387$). These results indicate that both image size and Gaussian filter level are important factors in segmentation performance and should be considered jointly in performance optimization.

Post-hoc analyses with Bonferroni correction revealed a statistically significant difference in IoU scores between the 128×128 and 256×256 image sizes ($p < 0.001$), while the difference between 128×128 and 512×512 was marginal ($p = 0.051$). A significant difference was also observed between the 256×256 and 512×512 image sizes ($p < 0.001$). Regarding the Gaussian filter, the most pronounced difference was found between sigma 1 and sigma 5, and statistically significant differences were detected between all pairwise comparisons of filter levels ($p < 0.001$).

Pixel Accuracy Score Results

Effect of Image Size

One-way ANOVA results indicated that image size had no statistically significant effect on Pixel Accuracy scores ($F(2, 82) = 0.847$, $p = 0.432$, $\eta^2 = 0.020$). No substantial differences in Pixel Accuracy were observed among the 128, 256, and 512 pixel resolutions, with similar

accuracy rates recorded across all three sizes. These results suggest that Pixel Accuracy is largely unaffected by input image resolution when evaluating segmentation performance.

Effect of Filtering

One-way ANOVA results demonstrated that Gaussian filtering had a highly significant and strong effect on Pixel Accuracy scores ($F(4, 164) = 77.714$, $p < 0.001$, $\eta^2 = 0.655$). Notably, the highest Pixel Accuracy was achieved with the lowest sigma value (filter 1), while a clear downward trend in accuracy was observed as the sigma value increased. These results indicate that lower sigma values should be preferred to maximize segmentation accuracy.

Interaction Between Size and Filter

When the effects of image size and sigma value combinations on Pixel Accuracy scores were examined, the one-way ANOVA results revealed a statistically significant and strong interaction effect ($F(8, 328) = 54.855$, $p < 0.001$, $\eta^2 = 0.572$). This indicates that the impact of the sigma value on Pixel Accuracy varies depending on the image resolution. For instance, the highest Pixel Accuracy scores were achieved at lower sigma values (particularly sigma 1) with 512×512 resolution images, whereas performance significantly declined at sigma 5 within the same resolution. These results emphasize the importance of combining high-resolution images with low sigma values to optimize segmentation accuracy.

In terms of Pixel Accuracy scores, the highest performance was achieved with high-resolution (512×512) images combined with low sigma values, particularly sigma 1 and 2. In contrast, the lowest scores were also observed at the 512×512 resolution, but when a high sigma value (sigma 5) was applied (Figure 37). These results highlight the critical role of selecting an appropriate combination of high image resolution and low sigma values to enhance segmentation accuracy in terms of Pixel Accuracy.

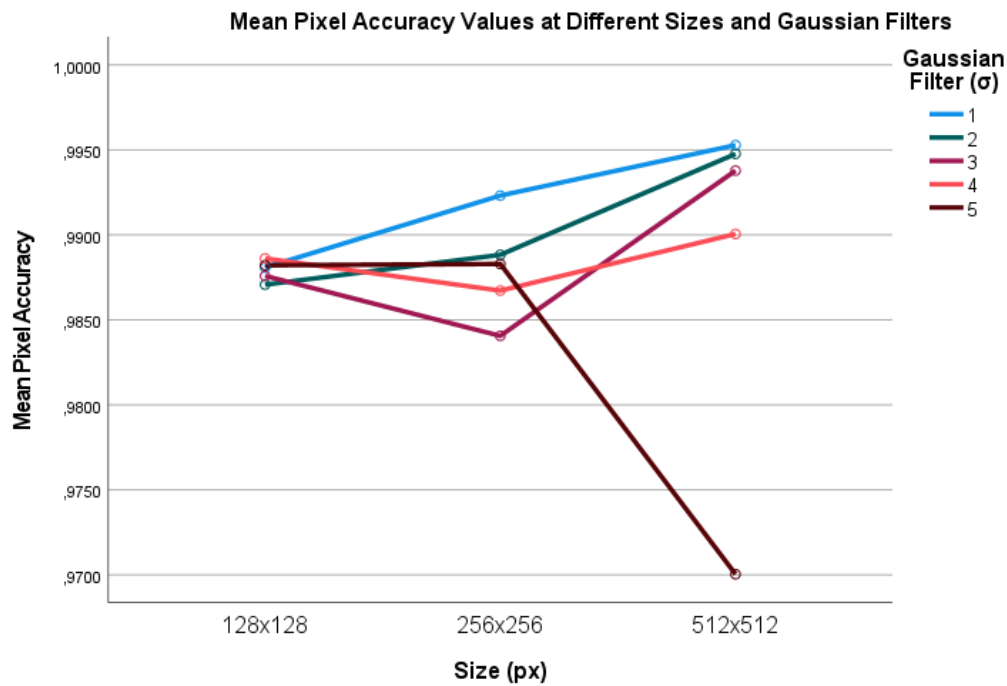


Figure 37. Pixel Accuracy Average Values Graph for Different Sizes and Filters

The effects of image dimensions (128×128, 256×256, and 512×512 pixels) and Gaussian filter levels (sigma 1–5) on pixel-wise accuracy scores were analyzed. Descriptive statistics based on repeated measures ANOVA revealed that PIXEL_ACU means ranged between 0.987 and 0.989 for the 128×128 size, and between 0.984 and 0.992 for the 256×256 size. In the 512×512 resolution, the highest mean score reached 0.995, with higher scores generally associated with lower filter levels.

Multivariate test results indicated that the main effect of image size was not statistically significant (Wilks' Lambda= 0.961, $F(2,40)= 0.817$, $p= 0.449$, $\eta^2= 0.039$). In contrast, the Gaussian filter levels demonstrated a significant main effect (Wilks' Lambda= 0.234, $F(4,38)= 31.151$, $p< 0.001$, $\eta^2= 0.766$). Additionally, the interaction between image size and Gaussian filter level was also found to be highly significant (Wilks' Lambda= 0.152, $F(8,34)= 23.694$, $p< 0.001$, $\eta^2= 0.848$), suggesting that segmentation accuracy is strongly influenced by the combined effects of these two factors rather than image size alone.

Mauchly's test of sphericity revealed a violation of the sphericity assumption for the image size factor ($W= 0.729$, $p= 0.002$), as well as for the Gaussian filter ($W= 0.193$, $p< 0.001$) and

their interaction ($W = 0.000$, $p < 0.001$). Accordingly, Greenhouse-Geisser corrections were applied in the analysis.

Within-subjects effects analysis confirmed that the main effect of image size was not statistically significant ($F(2,82) = 0.847$, $p = 0.432$, $\eta^2 = 0.020$). However, Gaussian filtering demonstrated a highly significant main effect ($F(4,164) = 77.714$, $p < 0.001$, $\eta^2 = 0.655$), and the interaction between size and filter level was also statistically significant and substantial ($F(8,328) = 54.855$, $p < 0.001$, $\eta^2 = 0.572$). These results indicate that the influence of Gaussian filtering on segmentation accuracy is considerably stronger than that of image size alone.

Post-hoc comparisons showed that the difference between sigma 1 and sigma 5 filter levels was statistically significant, with the highest Pixel Accuracy achieved at lower sigma values (e.g., sigma 1 vs. sigma 5: $p < 0.001$). In contrast, pairwise comparisons between different image sizes revealed no statistically significant differences (all $p > 0.05$).

Recall Score Results

Effect of Size

The effect of image size on Recall score was evaluated using one-way ANOVA, yielding a statistically significant result ($F(2,82) = 15.417$, $p < 0.001$, $\eta^2 = 0.273$). These results indicate that image size has a statistically significant and moderate impact on Recall performance. Notably, a substantial decrease was observed at the 256×256 resolution, whereas higher Recall scores were obtained at 128×128 and 512×512 resolutions. The mean Recall score was calculated as 0.753 for the 128×128 size and 0.728 for the 512×512 size, whereas it dropped to 0.587 at the 256×256 resolution. This outcome underscores the importance of selecting an appropriate image size to enhance segmentation sensitivity.

Effect of Filtering

The effect of sigma values on the Recall score was evaluated using one-way ANOVA, which revealed a statistically significant and moderate effect ($F(4,164) = 11.827$, $p < 0.001$, $\eta^2 = 0.224$). These results indicate that sigma values significantly influence Recall performance. Higher Recall scores were observed at sigma levels 1 and 2, while a noticeable decline was recorded at sigma levels 4 and 5. Specifically, sigma 1 (0.750) and sigma 2 (0.737) yielded

the highest performance, whereas sigma 5 resulted in the lowest Recall score (0.621). These results highlight the importance of selecting an appropriate sigma value to improve segmentation sensitivity.

Interaction Between Size and Filter

The interaction effect of image size and sigma value on Recall scores was evaluated using one-way ANOVA, yielding statistically significant and moderately strong results ($F(8,328)=13.768$, $p<0.001$, $\eta^2=0.251$). These results indicate a meaningful and robust interaction between image resolution and filter strength. For instance, the combination of 256×256 resolution and sigma 5 produced the lowest Recall score (0.312), whereas the 512×512 resolution paired with sigma 2 yielded the highest score (0.837). These results underscore the critical importance of selecting an optimal combination of image resolution and sigma value to enhance segmentation sensitivity.

In terms of recall metric, the highest accuracy was achieved with the combination of 512×512 resolution and sigma 2. In contrast, a significant performance drop was observed particularly with the combination of 256×256 resolution and sigma 5. These results indicate that optimizing both resolution and filter parameters is critically important to maximize sensitivity in the segmentation process (Figure 38).

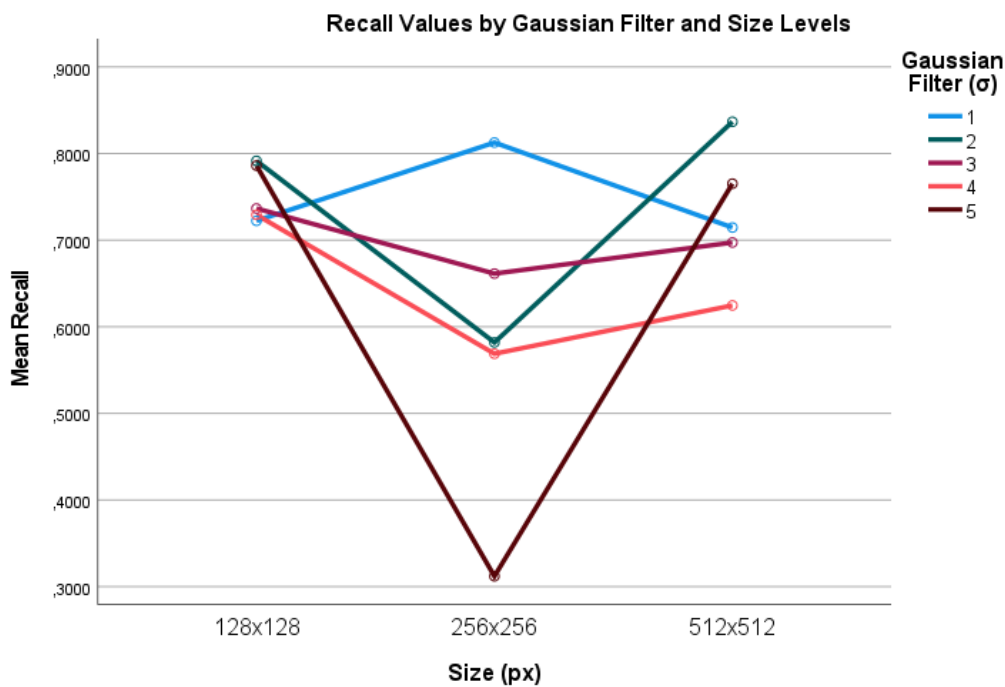


Figure 38. Graph of Recall Values According to Gaussian Filter and Image Size Levels

In this study, the effects of different image sizes (128×128, 256×256, and 512×512 pixels) and Gaussian filter levels (sigma 1–5) on recall scores were analyzed. Descriptive statistics revealed that the recall means ranged from 0.722 to 0.792 for 128×128 images, and from 0.312 to 0.813 for 256×256 images. For the 512×512 size, the highest average recall was observed at 0.837, particularly at lower filter levels.

According to the multivariate test results, the main effect of image size on recall was found to be statistically significant (Wilks' Lambda= 0.389, $F(2,40)= 31.472$, $p < 0.001$, $\eta^2= 0.611$). Similarly, Gaussian filter levels also showed a significant main effect (Wilks' Lambda= 0.481, $F(4,38)= 10.241$, $p < 0.001$, $\eta^2= 0.519$). The interaction effect between image size and Gaussian filter was also highly significant (Wilks' Lambda= 0.193, $F(8,34)= 17.799$, $p < 0.001$, $\eta^2= 0.807$). These results indicate that both image size and Gaussian filtering had a notable influence on recall scores, and their combination yielded even more pronounced effects.

Mauchly's sphericity test results showed that the assumption of sphericity was violated for the size factor ($W= 0.648$, $p= 0.000$), as well as for the Gaussian filter ($W= 0.566$, $p= 0.008$) and the size*filter interaction ($W= 0.067$, $p < 0.001$). Therefore, Greenhouse-Geisser corrections were applied in subsequent analyses.

In the within-subjects effects analysis, the main effect of size was significant ($F(2,82)= 15.417$, $p < 0.001$, $\eta^2= 0.273$). Gaussian filtering also showed a significant main effect ($F(4,164)= 11.827$, $p < 0.001$, $\eta^2= 0.224$). The interaction between size and Gaussian filtering was again statistically significant and moderately strong ($F(8,328)= 13.768$, $p < 0.001$, $\eta^2= 0.251$). These results confirm that both size and filtering levels individually and interactively affected recall performance.

Post-hoc analyses revealed that differences between Gaussian filter levels were statistically significant, particularly between sigma 1 and sigma 5, with higher recall values observed at lower sigma levels (e.g., 1–5 comparison: $p < 0.001$). Pairwise comparisons among image sizes showed that the 256×256 resolution had significantly lower recall scores compared to both 128×128 and 512×512 sizes (e.g., 2–1 and 2–3 comparisons: $p < 0.001$).

F-Score Results

Effect of Size

When evaluating the effect of image size on the F-Score, one-way ANOVA results indicated a statistically significant and strong influence ($F(2,82)= 25.053$, $p< 0.001$, $\eta^2= 0.379$). These results demonstrate that image size has a notable impact on F-Score performance. A marked performance decline was observed at the 256×256 resolution, whereas higher performance was achieved at 512×512 . The mean F-Score for the 512×512 size was 0.623 and for 128×128 it was 0.617, both relatively high. However, the average score for 256×256 (0.504) was significantly lower. These results underscore the importance of selecting larger image sizes to enhance segmentation performance.

Effect of Filtering

The evaluation of sigma value combinations revealed a significant and strong impact on F-Score performance ($F(4,164)= 66.718$, $p< 0.001$, $\eta^2= 0.619$). Among all sigma levels, sigma 1 achieved the highest F-Score (mean= 0.683), whereas increasing sigma values resulted in a notable decline. Sigma 5, in particular, showed the lowest performance with a mean score of 0.445. These results emphasize the importance of selecting lower sigma values to optimize segmentation accuracy.

Interaction Between Size and Filter

The analysis of size and sigma value combinations indicated a statistically significant and strong effect on F-Score performance ($F(8,328)= 17.696$, $p< 0.001$, $\eta^2= 0.301$). The combination of 512 image size and sigma 2 yielded the highest F-Score (0.766), whereas the pairing of 256 image size and sigma 5 resulted in the lowest performance. These results highlight the importance of carefully selecting the appropriate combination of image resolution and filter parameters to enhance segmentation outcomes.

In terms of F-Score, the highest accuracy was achieved with the combination of 512 image size and sigma 2. In contrast, the 256 image size combined with sigma 5 exhibited notably poor performance (Figure 39). These results indicate that selecting an appropriate combination of image size and sigma value plays a critical role in enhancing the model's F-Score performance.

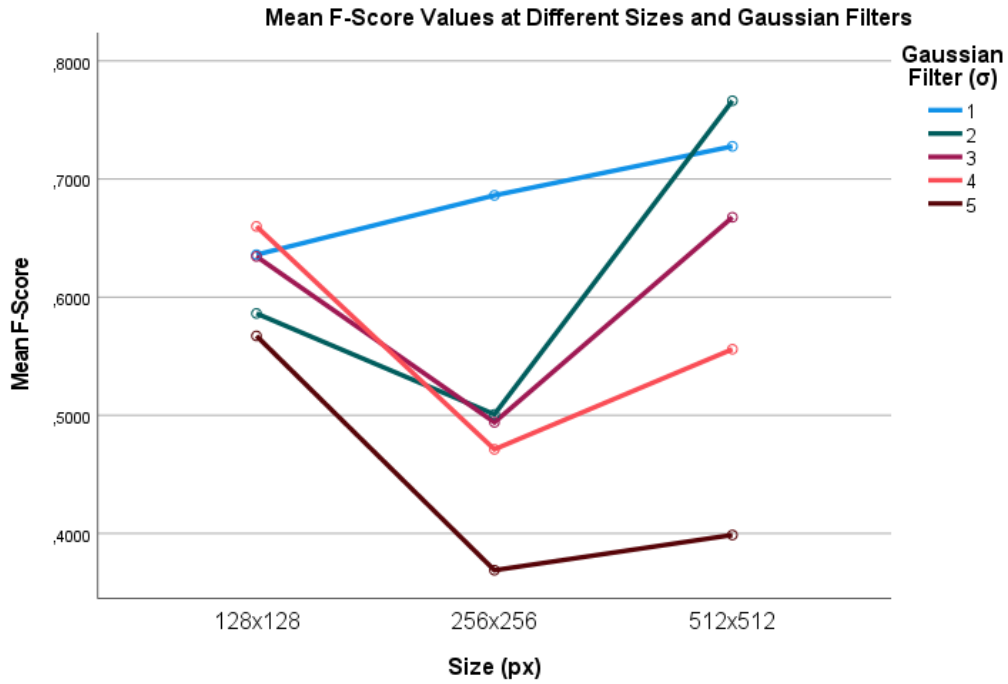


Figure 39. F-Score Average Values Graph in Different Dimensions and Filters

The effects of different image sizes (128×128, 256×256, 512×512 pixels) and Gaussian filter levels (sigma 1–5) on the F1 score were evaluated using repeated measures ANOVA. Descriptive statistics indicated that F1 scores ranged from 0.567 to 0.660 for 128×128 images, and from 0.369 to 0.686 for 256×256 images. The highest average F1 score was recorded at the 512×512 resolution with low filter levels (e.g., sigma 2: mean= 0.766).

According to multivariate analysis results, the main effect of image size was found to be statistically significant (Wilks' Lambda= 0.414, $F(2,40)= 28.321$, $p< 0.001$, $\eta^2= 0.586$). Gaussian filter levels also showed a significant main effect (Wilks' Lambda= 0.232, $F(4,38)= 31.513$, $p< 0.001$, $\eta^2= 0.768$). Additionally, the interaction effect between image size and Gaussian filter level was also strong and statistically significant (Wilks' Lambda= 0.235, $F(8,34)= 13.844$, $p< 0.001$, $\eta^2= 0.765$). These results indicate that both image size and Gaussian filter level significantly influence the F1 score and that their combination produces a strong interactive effect.

Mauchly's test of sphericity indicated that the sphericity assumption was violated for the Gaussian filter factor ($W = 0.221$, $p < 0.001$) and the image size $\times$ filter interaction ($W = 0.037$, $p < 0.001$), while the assumption was met for the image size factor ($W = 0.943$, $p = 0.309$). Therefore, Greenhouse-Geisser correction was applied in the analysis for the filter and interaction terms.

According to the within-subjects effects analysis, the main effect of image size was statistically significant ($F(2,82) = 25.053$, $p < 0.001$, $\eta^2 = 0.379$). Similarly, a significant main effect was also found for Gaussian filter levels ($F(4,164) = 66.718$, $p < 0.001$, $\eta^2 = 0.619$). The interaction between image size and filter level was also statistically significant ($F(8,328) = 17.696$, $p < 0.001$, $\eta^2 = 0.301$). These results indicate that both image size and Gaussian filter levels independently and interactively contribute significantly to model performance.

Post-hoc analyses revealed a significant difference between the 128×128 and 256×256 sizes ($p < 0.001$), while the difference between 128×128 and 512×512 was not significant ($p = 1.000$). Regarding Gaussian filter levels, differences particularly between sigma 1 and sigma 5 were statistically significant ($p < 0.001$), with a general trend showing that increasing the filter level led to a marked decrease in F1 scores.

Precision Results

Effect of Size

The analysis of image size revealed a statistically significant and substantial effect on precision ($F(2,82) = 17.152$, $p < 0.001$, $\eta^2 = 0.295$). Precision scores were notably higher for 512×512 images, while lower values were recorded for the 128×128 and 256×256 dimensions. The mean precision was 0.615 for 512×512 images, compared to 0.499 and 0.510 for 128×128 and 256×256 images, respectively. No significant difference was found between the 128×128 and 256×256 sizes. These results highlight the importance of using larger image resolutions to enhance segmentation sensitivity.

Effect of Filtering

The analysis of filter effects on precision revealed a statistically significant and substantial impact ($F(4,164)= 32.812$, $p< 0.001$, $\eta^2= 0.445$). Precision was highest at sigma 1, with a clear downward trend observed as the sigma value increased. The lowest precision scores were recorded at sigma 5. These results underscore the critical importance of selecting an appropriate sigma value to optimize precision in segmentation models.

Interaction Between Size and Filter

The analysis of the interaction between image size and sigma value on precision revealed a statistically significant and strong effect ($F(8,328)= 31.310$, $p< 0.001$, $\eta^2= 0.433$). The highest precision was achieved with the combination of 512 resolution and sigma 1, yielding a maximum precision score of 0.801. In contrast, the lowest precision was observed with the 512 resolution and sigma 5 combination. These results highlight the importance of using high-resolution images along with low sigma values to enhance segmentation sensitivity.

In terms of sensitivity, the most successful combination was 512 resolution with sigma 1. While the 512 resolution generally enhanced precision, the use of sigma 5 led to a notable decline. These results (Figure 40) underscore the critical importance of selecting the appropriate resolution and sigma value combination to maximize precision.

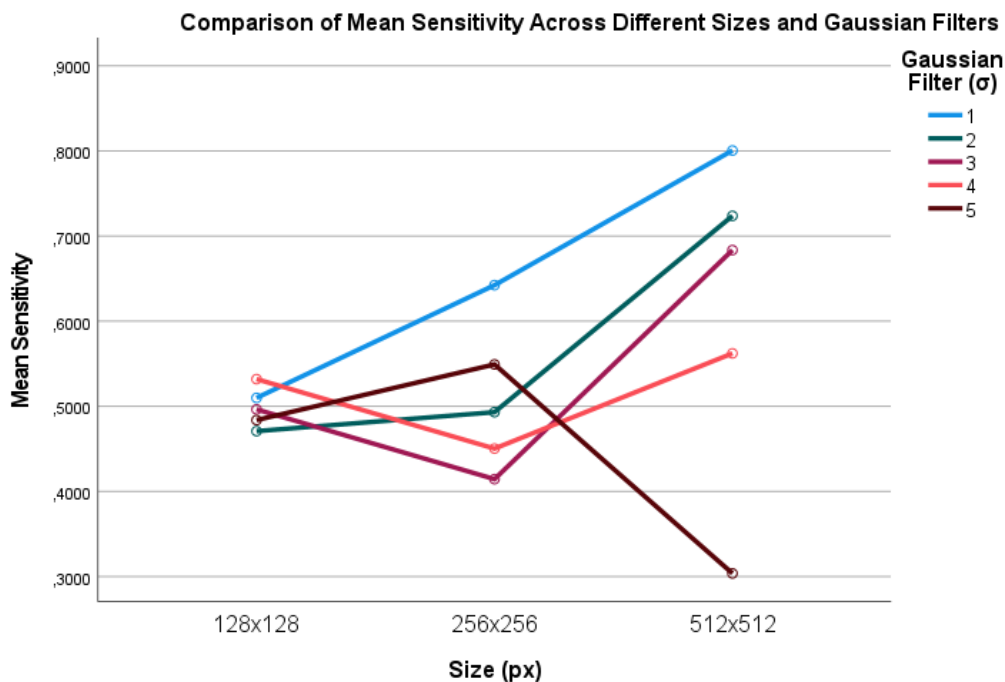


Figure 40. Graph of Precision Average Values in Different Size and Filters

The effects of different image sizes (128×128, 256×256, 512×512 pixels) and Gaussian filter levels (sigma 1–5) on precision were examined. Descriptive statistics indicated that precision scores ranged from 0.471 to 0.532 at 128×128 resolution, from 0.415 to 0.642 at 256×256, while the highest mean precision was recorded at 512×512 resolution particularly for sigma 1 (0.801).

Multivariate analysis results revealed a statistically significant main effect of image size on precision (Wilks' Lambda= 0.535, $F(2,40)= 17.364$, $p < 0.001$, $\eta^2 = 0.465$). A similarly strong main effect was found for Gaussian filter levels (Wilks' Lambda = 0.401, $F(4,38)= 14.185$, $p < 0.001$, $\eta^2 = 0.599$). In addition, the interaction between size and Gaussian filter was also highly significant (Wilks' Lambda= 0.152, $F(8,34)= 23.692$, $p < 0.001$, $\eta^2 = 0.848$). These results indicate that both resolution and filter level independently and interactively exert a meaningful influence on precision.

According to Mauchly's sphericity test, the assumption of sphericity was met for the image size factor ($W = 0.997$, $p = 0.947$), but was violated for both the Gaussian filter ($W = 0.275$, $p < 0.001$) and the size*filter interaction ($W = 0.067$, $p < 0.001$). Therefore, Greenhouse-Geisser corrections were applied for these effects.

According to the within-subjects analysis, the main effect of image size on precision was found to be statistically significant ($F(2,82)= 17.152$, $p < 0.001$, $\eta^2 = 0.295$). The Gaussian filter also showed a significant main effect ($F(4,164)= 32.812$, $p < 0.001$, $\eta^2 = 0.445$). Moreover, the interaction between image size and Gaussian filter was notably strong ($F(8,328)= 31.310$, $p < 0.001$, $\eta^2 = 0.433$), indicating that these two factors jointly exert a considerable impact on precision.

Post-hoc analyses revealed a significant difference between the 128×128 and 512×512 image sizes ($p < 0.001$), while the difference between 128×128 and 256×256 was not statistically significant ($p = 1.000$). Regarding Gaussian filter levels, significant differences were observed between sigma 1 and all other levels (e.g., sigma1 vs. sigma 5: $p < 0.001$).

These results suggest that precision tends to increase with higher image resolution, whereas higher Gaussian filter levels are associated with a decrease in precision. The highest precision

value (0.801) was obtained with the 512×512 size at sigma 1, highlighting that larger image sizes enhance precision, but this benefit diminishes as the filter strength increases.

Specificity Results

Effect of Size

The relationship between image size and specificity revealed a statistically significant and small-to-moderate effect based on one-way ANOVA results ($F(2,82)= 9.332$, $p=0.001$, $\eta^2= 0.185$). The highest specificity was observed in 128×128 images (mean= 0.955), followed by a notable decline in the 256×256 group (mean= 0.835). Specificity improved again at the 512×512 size (mean= 0.920), reaching a level similar to that of the 128×128 images and significantly higher than the 256×256 group. These results highlight the importance of choosing an optimal image size to enhance specificity in segmentation tasks.

Effect of Filtering

When evaluating the effect of sigma values on specificity, one-way ANOVA results showed a statistically significant and small-to-moderate effect ($F(4,164)= 8.188$, $p<0.001$, $\eta^2= 0.166$). The highest specificity was obtained with sigma 1 (mean= 0.970), while a consistent decline in specificity was observed as the sigma value increased. Notably, the use of sigma 5 resulted in a specificity score as low as 0.828. These results suggest that selecting lower sigma values may be important for maintaining high specificity in segmentation models.

Interaction Between Size and Filter

The analysis of the interaction between image size and sigma value on specificity revealed a statistically significant effect, albeit with a limited effect size ($F(8,328)= 3.414$, $p= 0.001$, $\eta^2= 0.077$). The highest specificity was observed with the combination of 128-pixel image size and low sigma values, particularly sigma 1. In contrast, certain combinations such as 256-pixel size with sigma 5 resulted in a marked decline in specificity. These results suggest that, in order to maintain specificity during the segmentation process, both image resolution and filter parameters should be carefully considered together.

The most successful combination in terms of specificity was found to be the 128-pixel image size with a sigma value of 1. In contrast, a notable decrease in specificity was observed when using the 256-pixel image size, especially in combination with a sigma value of 5 (Figure 41). These results indicate that, rather than relying solely on high resolution, selecting smaller

image sizes in conjunction with low sigma values is more effective for enhancing specificity performance.

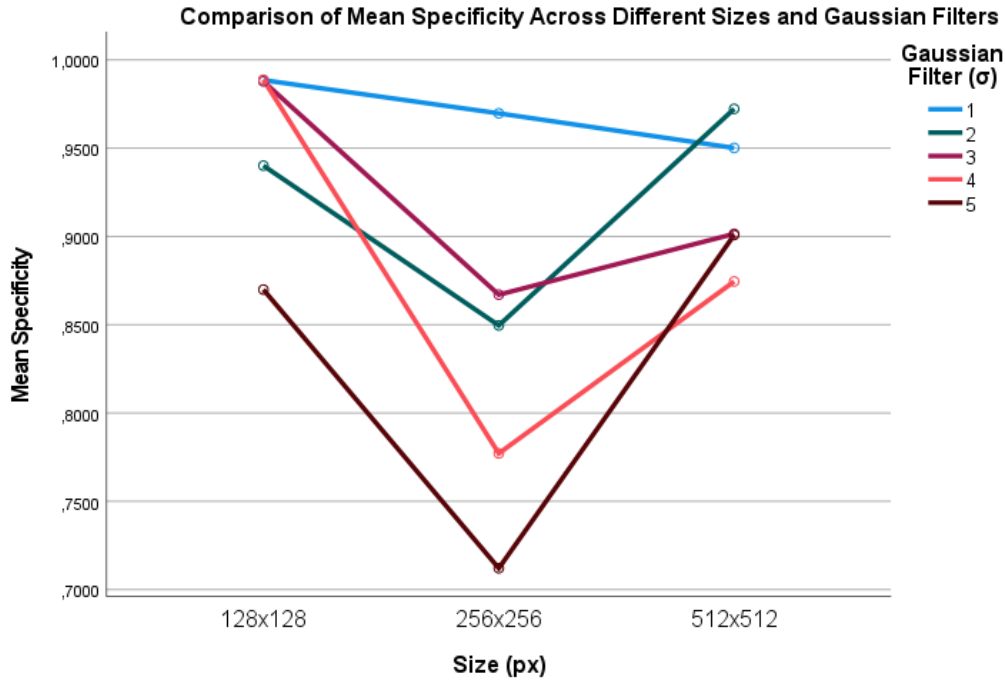


Figure 41. Graph of Specificity Average Values in Different Dimensions and Filters

The effects of different image sizes (128×128, 256×256, 512×512 pixels) and Gaussian filter levels (sigma 1–5) on specificity were examined. According to descriptive statistics, specificity values ranged from 0.870 to 0.989 for the 128×128 size, and from 0.712 to 0.970 for the 256×256 size. For the 512×512 size, the highest specificity (0.972) was recorded at sigma 2.

Multivariate analysis results showed a significant main effect of image size (Wilks' Lambda= 0.755, $F(2,40)= 6.499$, $p= 0.004$, $\eta^2= 0.245$) as well as a significant main effect of Gaussian filter level (Wilks' Lambda= 0.737, $F(4,38)= 3.397$, $p= 0.018$, $\eta^2= 0.263$). However, the interaction between image size and Gaussian filter was not statistically significant (Wilks' Lambda= 0.735, $F(8,34)= 1.529$, $p= 0.184$, $\eta^2= 0.265$). These results suggest that while both image size and Gaussian filter individually influence specificity, their combined interaction does not produce a significant effect.

According to Mauchly's test of sphericity, the assumption of sphericity was violated for image size ($W=0.686$, $p=0.001$), Gaussian filter ($W=0.147$, $p<0.001$), and their interaction ($W=0.001$, $p<0.001$). As a result, Greenhouse-Geisser correction was applied in all relevant analyses.

According to the within-subjects analyses, the main effect of image size was found to be statistically significant ($F(1.52, 62.4)=9.33$, $p=0.001$, $\eta^2=0.185$). A significant main effect was also observed for the Gaussian filter level ($F(2.21, 90.8)=8.19$, $p<0.001$, $\eta^2=0.166$). Additionally, the interaction between image size and filter level was significant ($F(3.86, 158.3)=3.41$, $p=0.011$, $\eta^2=0.077$). Post-hoc analyses revealed that the largest difference in terms of image size was between 128×128 and 256×256 , which was statistically significant ($p=0.005$). Regarding the Gaussian filter, the difference between sigma 1 and sigma 5 was particularly pronounced ($p=0.005$), and a significant difference was also noted between sigma 1 and sigma 4 ($p=0.049$). These results indicate that larger image sizes generally enhance specificity, whereas increasing the sigma level of the Gaussian filter tends to reduce specificity. In particular, specificity scores showed a marked decline at the sigma 5 level.

In conclusion, metrics such as Dice, IoU, Pixel Accuracy, Recall, F-score, Sensitivity, and Specificity vary depending on image size and filter levels. In general, smaller sigma values (particularly sigma 1 and sigma 2) yielded higher performance, whereas performance decreased significantly at sigma 5. Notably, substantial improvements were observed in Dice and IoU scores at the 512×512 resolution, suggesting that higher image resolutions enhance model performance. Pixel Accuracy remained consistently high across all image sizes, although a drop was observed at sigma 5. Similarly, the negative impact of sigma 5 was evident in Recall and Sensitivity scores, while the overall trend indicated that larger image sizes were associated with improved metric values. These results suggest that combining high-resolution images with lower sigma values is generally advantageous for optimizing segmentation performance.

4.4. Discussion

MRI, one of the most widely used and effective imaging methods in the diagnosis of liver tumors, is commonly preferred in clinical practice due to its high contrast advantage in differentiating focal liver masses, detecting malignant lesions such as HCC, and distinguishing between benign and malignant results (Edelman et al., 1996; Efe, 2003). Although the recent literature frequently examines liver lesion imaging through multi-modality approaches such as PET-MRI, PET-CT, SPECT-CT, or ultrasound-CT, there is a limited number of studies that provide a detailed analysis of MR images based on resolution, sequence type, and preprocessing techniques (e.g., filtering, histogram equalization).

This study aims to contribute to the early and accurate diagnosis of HCC by systematically evaluating the accuracy of automatic segmentation performed on T1- and T2-weighted MR images under varying image resolutions and preprocessing conditions. Unlike prior studies that mostly focus on a single sequence type, fixed resolution, or limited preprocessing steps, this work addresses the heterogeneous nature of real clinical data. In this context, segmentation model performance is comparatively analyzed using both the publicly available T1-sequence ATLAS dataset and T2-sequence images obtained from Çukurova University Faculty of Medicine Hospital. The evaluation includes three different resolution levels (128×128 , 256×256 , 512×512) and multiple preprocessing techniques, including Gaussian filtering and histogram equalization.

The results not only identify which combinations achieve the highest accuracy but also statistically demonstrate the impact of image type and parameter configurations on segmentation performance. Thus, the study highlights the importance of data and method selection in clinical diagnostic processes within machine learning applications.

The results indicate that metrics such as Dice, IoU, Pixel Accuracy, Recall, Sensitivity, and Specificity are significantly influenced by data type, contrast characteristics, resolution, and filtering parameters.

The T1-sequence ATLAS dataset, with its relatively large and balanced sample structure, demonstrated more stable and higher performance across most metrics; particularly at the

512×512 resolution, Dice (0.853) and IoU (0.7437) scores were achieved. This indicates the model's ability to successfully distinguish large and structured regions. A similar trend was reported by Vivanti et al. (2017), where the performance of automatic tumor tracking models improved as the data volume increased (Vivanti et al., 2017).

In contrast, the T2-sequence dataset developed in this study, despite its limited sample size, exhibited remarkable performance in terms of precision and minimizing false negatives. At 128×128 resolution, Recall (0.969) and Precision (0.6959) scores suggest that the lesion contrast in T2-weighted images enhances the model's capability to detect positive classes. Dai et al. (2025) similarly reported that T2 data provided superior precision compared to radiologists, especially for small lesions (Dai et al., 2025). Moreover, Fallahpoor et al. (2024) stated that although T2-based models were effective in liver segmentation, they remained limited in tumor detection (Fallahpoor et al., 2024). In our study, the high Specificity (0.9936) score achieved with 512×512 resolution T2 images demonstrates the model's ability to reduce false positives, indicating that such limitations can potentially be overcome.

In addition to the differential effects of imaging sequences on performance, another notable aspect of this study is the multidimensional analysis of the impact of Gaussian filtering on segmentation performance in relation to both resolution and sequence type. According to multivariate analysis results, both resolution ($p = 0.004$) and sigma level ($p = 0.018$) have a statistically significant main effect on specificity. Moreover, the interaction between resolution and filtering level was also found to be statistically significant ($p = 0.011$), indicating that the effect of filtering varies depending on image resolution, and that optimal performance cannot be achieved through uniform parameter settings.

At 128×128 resolution, low sigma values (sigma 1–2) supported segmentation performance, whereas a significant decline was observed in all metrics at sigma 5. While T1-sequence images at 256×256 resolution remained more robust against filtering, T2-sequence images showed a loss of precision as sigma increased. At 512×512 resolution, the most favorable filtering effect was observed at sigma 2, with specificity reaching as high as 0.972. This suggests that moderate smoothing enhances the model's ability to reduce false positives. Results by Shanbhogue et al. (2021) regarding the susceptibility of artifacts and edge

sharpness in T2-sequence images to filtering are consistent with the results of this study (Shanbhogue et al., 2021).

At the high-resolution level of 512×512 , the positive effect of Gaussian filtering became more pronounced. Specifically, at sigma 2, specificity reached notably high values (e.g., 0.972) in both T1 and T2 datasets. This indicates that moderate filtering can help reduce false positives while enhancing segmentation accuracy with minimal information loss. However, at higher sigma levels such as sigma 5, especially in T2 sequences, segmentation boundaries became blurred and accuracy decreased.

In terms of histogram equalization techniques, the CLAHE method generally improved segmentation performance at high resolution across both datasets. However, at low and medium resolutions, the HE method yielded more consistent results than CLAHE on T2 images. These results suggest that histogram equalization may not be universally beneficial and should be carefully evaluated in relation to data type and resolution.

In conclusion, this study presents a unique multidimensional approach that not only evaluates the performance of segmentation models but also systematically investigates influential image properties (T1/T2), resolution, filtering, and histogram equalization as preprocessing factors. While similar studies in the literature tend to focus primarily on image types or model architectures (Fallahpoor et al., 2024; Dai et al., 2025; Vivanti et al., 2017), the parametric assessment of image resolution and preprocessing settings constitutes a noteworthy contribution of this work.

Although T1-weighted images (despite a limited sample size) demonstrated more stable performance overall, T2-weighted images yielded more favorable results in specific clinical objectives, particularly in Recall, Sensitivity, and Specificity. Regarding Gaussian filtering, sigma levels of 1–2 were generally associated with better performance, whereas higher values such as sigma 5 clearly led to a decline in segmentation quality across both T1 and T2 images. These results highlight that the performance of medical image segmentation models is not solely dependent on algorithmic architecture, but also significantly influenced by data characteristics and preprocessing strategies. A combined evaluation of sequence type,

resolution, contrast profile, and filtering parameters has the potential to enhance both the accuracy of model outputs and their reliability in clinical applications.



5. CONCLUSIONS AND FUTURE WORKS

This study systematically evaluates the impact of T1- and T2-weighted MR images on segmentation performance in HCC by employing different resolutions and preprocessing techniques, thereby addressing a critical gap in the literature. Factors influencing segmentation success are not limited to model architecture alone; sequence type, resolution level, and preprocessing parameters also exert a direct influence on performance. Accordingly, this work not only assesses accuracy metrics but also offers a holistic analysis of the entire image processing pipeline.

The T1-weighted ATLAS dataset, with its large sample size and well-established annotations, yielded consistent and high scores in metrics such as Dice, IoU, Pixel Accuracy, and Specificity. Notably, Dice (0.853) and IoU (0.7437) scores achieved with 512×512 resolution demonstrate the model's capability to accurately identify macroscopic anatomical structures. In contrast, CU-HCC T2-weighted dataset although limited in sample size showed particularly high Recall (0.969) and Sensitivity (0.6959) scores at the 128×128 resolution. This suggests that T2 images may offer an advantage in identifying small or ill-defined lesions due to their richer lesion-to-background contrast.

Another factor as crucial as resolution in segmentation performance is the parametric tuning of preprocessing steps, particularly Gaussian filtering and histogram equalization. In this study, the impact of Gaussian filtering on segmentation was evaluated across various sigma values, demonstrating that its effect varied significantly depending on image resolution and sequence type. At 128×128 resolution, lower sigma values (sigma 1–2) improved segmentation accuracy, whereas a substantial degradation was observed at sigma 5. At 512×512 resolution, sigma 2 yielded the highest Specificity (0.972) scores in both T1- and T2-weighted datasets. These results indicate that moderate smoothing filters can reduce false positives while preserving lesion boundaries.

A notable and often overlooked contribution of this study is the use of statistical comparisons of performance metrics across groups, rather than relying solely on descriptive reporting. While most segmentation studies in the literature report metrics such as Dice, IoU, or Recall without evaluating statistical significance, this study applied multivariate analyses

(MANOVA) to test the effects of Gaussian filtering and resolution levels. The analyses revealed that both main effects and interactions had statistically significant impacts on critical metrics such as Specificity. This approach not only identifies which configurations yield better results but also determines whether these differences are scientifically valid, thereby enhancing the academic rigor and clinical reliability of the study.

Similarly, in histogram equalization methods, the CLAHE technique positively influenced segmentation performance at high resolution across both datasets. However, at low and medium resolutions, the HE method produced more consistent results on T2 images. These results highlight that preprocessing techniques are not universally optimal and should be tailored according to data type and image resolution.

The clinical and systemic implications of these technical results are also noteworthy. When the appropriate resolution, sequence type, and optimal filtering parameters (e.g., sigma value) are selected, not only does model accuracy improve, but computational load is also reduced. This is particularly beneficial for clinical AI systems operating with large datasets, as it decreases computational costs and enhances system efficiency, enabling faster performance of decision support mechanisms. Achieving high accuracy with low-resolution images through appropriate filtering parameters also makes it possible to train models with lighter architectures in a shorter time.

One limitation of this study is the relatively small sample size in the T2 dataset. However, this does not undermine the methodological and clinical contributions of the study. Future analyses involving different organ systems, a wider range of imaging sequences, and multicenter datasets are expected to support these results.

In conclusion, this study clearly demonstrates that data type, resolution, preprocessing methods, and parametric settings must be considered together within the image processing pipeline to optimize medical image segmentation performance. While previous literature has addressed these variables in isolation, this work provides an integrated, multidimensional evaluation, thereby contributing to the development of accurate, reliable, and efficient AI solutions for clinical decision support systems.

The results of this study demonstrate that preprocessing steps, image sequence types, and resolution levels exert significant influence on model performance in the field of medical image segmentation. However, to enable the generalization of these results to broader clinical applications and to enhance the effectiveness of AI-assisted imaging systems, certain aspects require further development in future research. The key areas for improvement are outlined below:

This study focused exclusively on T1- and T2-weighted MRI sequences. However, magnetic resonance imaging offers a wide range of modalities, including diffusion-weighted imaging (DWI), apparent diffusion coefficient (ADC), and hepatobiliary contrast-enhanced sequences. Future studies should comparatively evaluate each of these sequences in terms of segmentation accuracy, contrast advantages, and sensitivity to preprocessing techniques. Identifying which sequence is most appropriate for specific clinical scenarios would enable sequence-specific model optimization.

In this study, three different resolutions (128×128 , 256×256 , 512×512) were used. However, testing medium-sized (such as 192×192 , 384×384) and non-square (such as 320×256) sizes is important to evaluate the adaptation capacity of the model to different image ratios. In addition, the image sizes on which the model is trained and tested should be compatible with the output resolutions of clinical imaging systems, which will pave the way for real-time clinical use. Future studies should increase this diversity and analyze the size-optimization relationship in more detail.

The limited number of T2 sequence images naturally limits the learning capacity of the model against this data type. This affects both the stability of the segmentation outputs and the power of statistical analyses. Therefore, in future studies, the number of samples for both T1 and T2 sequence images should be increased, and if possible, a more representative data structure should be created by using multi-center data pools. This expansion will directly increase the generalizability of the model and its performance on real-world data.

The results obtained in the study revealed that preprocessing steps can have different effects depending on the sequence type. In particular, operations such as Gaussian filtering and histogram equalization produced different results in T1 and T2 sequences. In the future, these

processes should not be designed as “uniform” but rather dynamic and adaptable with sequence-specific parameters. Thus, each image type can be optimized with the most appropriate preprocessing method for its structure.

As shown in the study, when the appropriate resolution, filter type and sigma value are determined, not only the accuracy but also the computational load of the model can be optimized. This provides a great advantage especially in systems with limited computational capacity in the clinical environment. Future studies should focus on developing parametric models that balance performance and computational load. Thus, both accuracy is preserved and the executable of the model is increased.

Finally, correlating segmentation outputs with clinical outcomes (e.g., pathology reports, patient follow-up data) is necessary to evaluate the real clinical utility of the developed model. Future studies should focus on holistic patient-based outcomes, not just image-based ones, so that the impact of AI on patient management can be more clearly demonstrated.

6. REFERENCES

- Alawya, B., & Constantinou, C. (2023). Hepatocellular carcinoma: A narrative review on current knowledge and future prospects. *Current Treatment Options in Oncology*, 24(7), 711–724.
- Almotairi, S., Kareem, G., Aouf, M., Almutairi, B., & Salem, M. A.-M. (2020). Liver tumor segmentation in CT scans using modified SegNet. *Sensors*, 20(5), 1516.
- Altaş, İ. (2023). Performance evaluation of Jaccard–Dice coefficient on building segmentation from high resolution satellite images. *Balkan Journal of Electrical & Computer Engineering*, 11(1), 1–6.
- Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., ... & Farhan, L. (2021). Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. *Journal of Big Data*, 8(1), 83.
- Anstee, Q. M., Reeves, H. L., Kotsiliti, E., Govaere, O., & Heikenwalder, M. (2019). From NASH to HCC: Current concepts and future challenges. *Nature Reviews Gastroenterology & Hepatology*, 16(7), 411–428.
- Anter, A. M., & Hassenian, A. E. (2019). CT liver tumor segmentation hybrid approach using neutrosophic sets, fast fuzzy C-means and adaptive watershed algorithm. *Artificial Intelligence in Medicine*, 97, 105–117.
- Araújo, J. D. L., da Cruz, L. B., Diniz, J. O. B., Ferreira, J. L., Silva, A. C., de Paiva, A. C., & Gattass, M. (2022). Liver segmentation from computed tomography images using cascade deep learning. *Computers in Biology and Medicine*, 140, 105095.
- Ben-Cohen, A., Diamant, I., Klang, E., Amitai, M., & Greenspan, H. (2016). Fully convolutional network for liver segmentation and lesion detection. In *Deep Learning and Data Labeling for Medical Applications* (pp. 77–85). Springer.
- Bhan, I. (2020). Weight loss can help head off lasting damage caused by fatty liver. Retrieved from <https://www.health.harvard.edu/blog/weight-loss-can-help-head-off-lasting-damage-caused-by-fatty-liver-2020043019664>
- Bhatt, D., Patel, C., Talsania, H., Doshi, N., & Ghayvat, H. (2021). CNN variants for computer vision: History, architecture, application, challenges and future scope. *Electronics*, 10(20), 2470.
- Bousabarah, K., Letzen, B., Tefera, J., Savic, L., Schobert, I., Schlachter, T., ... & Lin, M. (2021). Automated detection and delineation of hepatocellular carcinoma on

- multiphasic contrast-enhanced MRI using deep learning. *Abdominal Radiology*, 46(1), 216–225.
- Bray, F., Ferlay, J., Soerjomataram, I., Siegel, R. L., Torre, L. A., & Jemal, A. (2018). Global cancer statistics 2018: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: A Cancer Journal for Clinicians*, 68(6), 394–424.
- Campadelli, P., Casiraghi, E., & Esposito, A. (2009). Liver segmentation from computed tomography scans: A survey and a new algorithm. *Artificial Intelligence in Medicine*, 45(2–3), 185–196.
- Christ, P. F., Elshaer, M. E. A., Ettlinger, F., Tatavarty, S., Bickel, M., Bilic, P., ... & Menze, B. H. (2016). Automatic liver and lesion segmentation in CT using cascaded fully convolutional neural networks and 3D conditional random fields. In S. Ourselin, L. Joskowicz, M. R. Sabuncu, G. Unal, & W. Wells (Eds.), *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016* (pp. 415–423). Springer.
- Dai, H., Xiao, Y., Fu, C., Grimm, R., von Busch, H., Stieltjes, B., ... & Zeng, M. (2025). Deep learning-based approach for identifying and measuring focal liver lesions on contrast-enhanced MRI. *Journal of Magnetic Resonance Imaging*, 61(1), 111–120.
- Das, A., Acharya, U. R., Panda, S. S., & Sabut, S. (2019). Deep learning based liver cancer detection using watershed transform and Gaussian mixture model techniques. *Cognitive Systems Research*, 54, 165–175.
- Drozdzal, M., Chartrand, G., Vorontsov, E., et al. (2018). Learning normalized inputs for iterative estimation in medical image segmentation. *Medical Image Analysis*, 44, 1–13.
- Ergüney, S. (2019). Primary liver tumors. Retrieved January 1, 2025 <https://kanservakfi.com/wp-content/uploads/2019/02/Volume-53-Primer-Karacig%CC%86er-Tu%CC%88mo%CC%88rleri.pdf>
- Fallahpoor, M., Nguyen, D., Montahaei, E., Hosseini, A., Nikbakhtian, S., Naseri, M., ... & Abbasi, M. (2024). Segmentation of liver and liver lesions using deep learning. *Physical and Engineering Sciences in Medicine*, 47(2), 611–619.
- Fang, X., Xu, S., Wood, B. J., & Yan, P. (2020). Deep learning-based liver segmentation for fusion-guided intervention. *International Journal of Computer Assisted Radiology and Surgery*, 15(6), 963–972.

- Gul, S., Khan, M. S., Bibi, A., Khandakar, A., Ayari, M. A., & Chowdhury, M. E. (2022). Deep learning techniques for liver and liver tumor segmentation: A review. *Computers in Biology and Medicine*, 147, 105620.
- Hadi, M. U., Khan, M. A., Rehman, A., Saba, T., & Mehmood, Z. (2023). A lightweight CORONA-NET for COVID-19 detection in X-ray images. *Expert Systems with Applications*, 225, 120023.
- Hamet, P., & Tremblay, J. (2017). Artificial intelligence in medicine. *Metabolism*, 69(Supplement), S36–S40.
- Herek, D., & Karabulu, N. (2010). Magnetic resonance imaging. *TTD Thoracic Surgery Bulletin*, 1(3), 214–233.
- Hossain, M. S. A., Gul, S., Chowdhury, M. E. H., Khan, M. S., Sumon, M. S. I., Bhuiyan, E. H., ... & Alqahtani, A. (2023). Deep learning framework for liver segmentation from T1-weighted MRI images. *Sensors*, 23(21), 8890.
- Isensee, F., Petersen, J., Klein, A., et al. (2018). nnU-Net: Self-adapting framework for U-Net-based medical image segmentation. *arXiv preprint, arXiv:1809.10486*.
- Jiang, H., Shi, T., Bai, Z., & Huang, L. (2019). AHCNet: An application of attention mechanism and hybrid connection for liver tumor segmentation in CT volumes. *IEEE Access*, 7, 24898–24909.
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436–444.
- Li, X., Chen, H., Qi, X., Dou, Q., Fu, C.-W., & Heng, P.-A. (2018). H-DenseUNet: Hybrid densely connected UNet for liver and tumor segmentation from CT volumes. *IEEE Transactions on Medical Imaging*, 37(12), 2663–2674.
- Li, X., Luo, S., & Li, J. (2013). Liver segmentation from CT image using fuzzy clustering and level set. *Journal of Signal and Information Processing*, 4(3), 36–42.
- Low RN. (2007). Abdominal MRI advances in the detection of liver tumours and characterisation. *Lancet Oncol.* 8:525- 533.
- Mathew, A., Amudha, P., & Sivakumari, S. (2021). Deep learning techniques: An overview. In H. A. El-Mousa, A. M. Al-Saleh, & M. A. Al-Zoubi (Eds.), *Advanced Machine Learning Technologies and Applications* (pp. 599–608). Springer.
- Marrero, J. A., Kulik, L. M., Sirlin, C. B., Zhu, A. X., Finn, R. S., Abecassis, M. M., ... & Heimbach, J. K. (2018). Diagnosis, staging, and management of hepatocellular carcinoma: 2018 practice guidance by the American Association for the Study of Liver Diseases. *Hepatology*, 68(2), 723–750.

- Müller, D., Soto-Rey, I., & Kramer, F. (2022). Towards a guideline for evaluation metrics in medical image segmentation. *BMC Research Notes*, 15(1), 210
- Nasu, K., Kuroki, Y., Tsukamoto, T., Nakajima, H., Mori, K., & Minami, M. (2009). Diffusion-weighted imaging of surgically resected hepatocellular carcinoma: Imaging characteristics and relationship among signal intensity, apparent diffusion coefficient, and histopathologic grade. *AJR. American Journal of Roentgenology*, 193(2), 438–444.
- Nishida, N., Yamakawa, M., Shiina, T., & Kudo, M. (2019). Current status and perspectives for computer-aided ultrasonic diagnosis of liver lesions using deep learning technology. *Hepatology International*, 13(4), 416–421.
- Özcan, F., Uçan, O. N., Karaçam, S., & Tunçman, D. (2023). Fully automatic liver and tumor segmentation from CT image using an AIM-Unet. *Bioengineering*, 10(215), 1–15.
- Pang, Y., Hu, D., & Sun, M. (2019, December). A modified scheme for liver tumor segmentation based on cascaded FCNs. In *Proceedings of the International Conference on Artificial Intelligence, Information Processing and Cloud Computing (AIIPCC)* (pp. 1–6). Sanya, China.
- Pérez, M. J., & Grande, G. R. (2020). Application of artificial intelligence in the diagnosis and treatment of hepatocellular carcinoma: A review. *World Journal of Gastroenterology*, 26(37), 5617–5628.
- Peterson, M. S., & Baron, R. L. (2001). Radiologic diagnosis of hepatocellular carcinoma. *Clinics in Liver Disease*, 5(1), 123–144.
- Purwono, P., Nugroho, A. S., Setyawan, R., Sari, R. F., & Ardiyanto, I. (2023). Understanding of convolutional neural network (CNN): A review. *International Journal of Robotics and Control Systems*, 2(4), 739–748.
- Quinton, F., Lu, P., Zhang, Y., & Ruan, S. (2023). A Tumour and Liver Automatic Segmentation (ATLAS) dataset on contrast-enhanced magnetic resonance imaging for hepatocellular carcinoma. *Data*, 8(5), 79.
- Radtke, A., Nadalin, S., Sotiropoulos, G. C., Molmenti, E. P., Schroeder, T., Valentin-Gamazo, C., ... & Malagó, M. (2007). Computer-assisted operative planning in adult living donor liver transplantation: A new way to resolve the dilemma of the middle hepatic vein. *World Journal of Surgery*, 31(1), 175–185.
- Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional networks for biomedical image segmentation. *arXiv*. <https://arxiv.org/abs/1505.04597>

- Saifullah, S., & Dreżewski, R. (2024). Enhanced medical image segmentation using CNN based on histogram equalization. In *Proceedings of the Second International Conference on Applied Artificial Intelligence and Computing (ICAAIC 2023)*. IEEE. ISBN: 978-1-6654-5630-2
- Samat N.A, Ali H., Salleh M.N.B.M. The Comparison of Pooling Functions in Convolutional Neural Network for Sentiment Analysis Task. *Advances in Intelligent Systems and Computing*. 2020.
- Seo, H., Huang, C., Bassenne, M., Xiao, R., & Xing, L. (2020). Modified U-Net (mU-Net) with incorporation of object-dependent high level features for improved liver and liver-tumor segmentation in CT images. *IEEE Transactions on Medical Imaging*, 39(5), 1316–1325.
- Shanbhogue, K., Tong, A., Smereka, P., Nickel, D., Arberet, S., Anthopolos, R., & Chandarana, H. (2021). Accelerated single-shot T2-weighted fat-suppressed (FS) MRI of the liver with deep learning-based image reconstruction: Qualitative and quantitative comparison of image quality with conventional T2-weighted FS sequence. *European Radiology*, 31(11), 8447–8457.
- Skwirczyński, M., Tabor, Z., Lasek, J., Schneider, Z., Gibała, S., Kucybała, I., ... & Obuchowicz, R. (2023). Deep learning algorithm for differentiating patients with a healthy liver from patients with liver lesions based on MR images. *Cancers*, 15(12), 3142.
- Srivastava, G., & Rawat, T. K. (2013). Histogram equalization: A comparative analysis and a segmented approach to process digital images. In *Proceedings of the 2013 Sixth International Conference on Contemporary Computing (IC3)*.
- Sun, C., Guo, S., Zhang, H., Li, J., Chen, M., Ma, S., Jin, L., Liu, X., Li, X., & Qian, X. (2017). Automatic segmentation of liver tumors from multiphase contrast-enhanced CT images based on FCNs. *Artificial Intelligence in Medicine*, 83, 58–66.
- The NiBabel development team. (2025). NiBabel: Python package to access a cacophony of neuro-imaging file formats. Retrieved January 1, 2025, from <https://github.com/nipy/nibabel>
- Tang, A., Bashir, M. R., Corwin, M. T., Cruite, I., Dietrich, C. F., Do, R. K. G., ... & Sirlin, C. B. (2018). Evidence supporting LI-RADS major features for CT- and MR imaging-based diagnosis of hepatocellular carcinoma: A systematic review. *Radiology*, 286(1), 29–48.

- Taşdoğan, B. E. (2021). What kind of organ is the liver and what does it do in the body? Retrieved January 1, 2025 <https://www.tkad.org.tr/wp-content/uploads/2021/09/karaciger-nasil-bir-organdır-vucutta-ne-yapar.pdf>
- Tran, S.-T., et al. (2021). Multiple Layer U-Net (Un-Net) for liver and liver tumor segmentation in CT. *IEEE Access*, 9, 3047861.
- Trefts, E., Gannon, M., & Wasserman, D. H. (2017). The liver. *Current Biology*, 27(21), R1147–R1151.
- Wang, K., Mamidipalli, A., Retson, T., Bahrami, N., Hasenstab, K., Blansit, K., ... & Hsiao, A. (2019). Automated CT and MRI liver segmentation and biometry using a generalized convolutional neural network. *Radiology: Artificial Intelligence*, 1(2), 180022.
- Wang, J., Peng, Y., Jing, S., Han, L., Li, T., & Luo, J. (2023). A deep-learning approach for segmentation of liver tumors in magnetic resonance imaging using UNet+. *BMC Cancer*, 23(1), 1060.
- Willatt, J. M., Hussain, H. K., Adusumilli, S., & Marrero, J. A. (2008). MR imaging of hepatocellular carcinoma in the cirrhotic liver: Challenges and controversies. *Radiology*, 247(2), 311–330.
- Wu, W., Wu, S., Zhou, Z., Zhang, R., & Zhang, Y. (2017). 3D liver tumor segmentation in CT images using improved fuzzy C-means and graph cuts. *BioMed Research International*, 2017, 1–11.
- Wuthrich, M., Trimpe, S., Kappler, D., & Schaal, S. (2015). A new perspective and extension of the Gaussian filter. In *Proceedings of Robotics: Science and Systems (RSS)*, 11.
- Venkatesh, S., et al. (2022). Image Enhancement and Implementation of CLAHE Algorithm and Bilinear Interpolation. *Cybernetics and Systems: An International Journal*.1-13.
- Villanueva, A. (2019). Hepatocellular carcinoma. *The New England Journal of Medicine*, 380(15), 1450–1462.
- Vivanti, R., Szeskin, A., Lev-Cohain, N., Sosna, J., & Joskowicz, L. (2017). Automatic detection of new tumors and tumor burden evaluation in longitudinal liver CT scan studies. *International Journal of Computer Assisted Radiology and Surgery*, 12(11), 1945–1957.
- Vorontsov, E., Cerny, M., Régnier, P., Di Jorio, L., Pal, C. J., Lapointe, R., ... & Tang, A. (2019). Deep learning for automated segmentation of liver lesions at CT in patients

- with colorectal cancer liver metastases. *Radiology: Artificial Intelligence*, 1(2), 180014.
- Yamashita, R., Nishio, M., Do, R. K. G., & Togashi, K. (2018). Convolutional neural networks: an overview and application in radiology. *Insights Imaging*, 9(4), 611-629.
- Yasaka, K., Akai, H., Abe, O., & Kiryu, S. (2018). Deep learning with convolutional neural network for differentiation of liver masses at dynamic contrast-enhanced CT: A preliminary study. *Radiology*, 286(3), 887–896.
- Zafar A, Aamir M, Mohd Nawi N, et al. (2022). A Comparison of Pooling Methods for Convolutional Neural Networks. *Applied Sciences*.12(17):8643.
- Zheng, S., Fang, B., Li, L., Gao, M., Wang, Y., & Peng, K. (2019). Automatic liver tumour segmentation in CT combining FCN and NMF-based deformable model. *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, 1–10.