ISSN 2007-9737

A Comparative Study on Thyroid Segmentation Using Filters and Transfer Learning

Christopher Gutierrez[1], Fernando Gaxiola[1,*], Patricia Melin[2], Alain Manzo-Martinez[1], Luis Gonzalez-Gurrola[1], Graciela Ramirez-Alonso[1]

[1] Autonomous University of Chihuahua, Faculty of Engineering, Mexico

[2] Tijuana Institute of Technology/TecNM, Mexico

{igaxiola, cmontiel}@uach.mx

Abstract. Ultrasound (US) images are commonly used for analyzing soft tissues because they are non-invasive and low-cost. US image quality can be compromised by low contrast and noise, potentially impairing radiologists' interpretation and affecting the accuracy of computer algorithms in segmentation and analysis characterization. Successful segmentation models are often computationally intensive, making them less practical for diagnostic applications. In this work, a U-Net model was developed that employs a CNN as the encoder for segmentation, combined with image preprocessing using border-detection filters to enhance edges and reduce noise. We compare the performance of four segmentation models, all implemented with the same data. Notably, U-Net with Xception model encoder and preprocessing with bilateral filter and CLAHE achieves high performance, with a Dice score of 0.7476 and an IoU of 0.6862.

Keywords. Segmentation, sonography, thyroid, transfer learning, filter.

1 Introduction

The thyroid gland is an endocrine organ that produces hormones responsible for regulating heart rate, blood pressure, body temperature, and various metabolic processes. Disorders related to the thyroid can present as structural symptoms, such as goiter, nodules, and carcinoma, or functional symptoms, including hypothyroidism and hyperthyroidism [1]. When the hormonal secretion process is altered, this can be linked to unwanted growth of nodules, thereby affecting the metabolic system and causing conditions such as diabetes, hypertension, or hypotension [2]. In the past three years, the incidence of thyroid nodules has significantly increased, likely due to irregular dietary habits and inconsistent work-rest schedules [3]. The presence of nodules constitutes an indicator of thyroid cancer. In 2020, approximately 590,000 individuals worldwide were diagnosed with thyroid cancer, making it the tenth most common type of cancer. Timely diagnosis and treatment are essential for enhancing survival rates and understanding the disease's progression more effectively [3]. Thyroid nodules are lumps that often develop within the thyroid gland and usually do not cause any symptoms. These nodules are commonly found during routine neck examinations. When thyroid nodules grow large and cause noticeable symptoms, imaging tests like ultrasound (US) and computed tomography (CT) are used to visualize them [4]. Ultrasound images are extensively employed for the examination of soft tissue organs owing to their non-invasive nature and cost-effectiveness [5]. Although CT images are detailed, their use is less frequent due to their high cost, exposure to ionizing radiation, and the possibility of adverse reactions to contrast media [6]. The visual quality of US images is often compromised by lower contrast and higher noise, which limits edge detection and affects the detectability of lesions during diagnosis or interpretation by radiologists [7]. Consequently,

supplementary applications are necessary in conjunction with conventional diagnostic techniques to ensure precise evaluation and prompt identification. Techniques for segmenting thyroid nodules and glands are classified into three primary categories: machine learning (ML), deep learning (DL), and methods based on region, contour, and shape analysis [8]. Although frequently employed owing to their substantial 92% overlap, also recognized as Intersection over Union (IoU), in segmentation efficacy, region-based approaches have predominantly been assessed on images derived from limited datasets [9], [10]. These techniques can produce reliable results even in noisy images. However, this approach is often highly sensitive to the quality of the images. Additionally, they require an initial manual delineation step and careful parameter tuning, which increases their computational cost and may affect their performance when working with low-resolution datasets. Region-based methods [11] administer the segmentation process using the grayscale intensity of the pixels within a designated region. Variations between high and low grayscale intensity pixels determine the boundaries of the regions to be segmented. These methods produce effective results for images with homogeneous and smooth structures. Fast segmentation processes and ease of application are significant advantages.

However, these methods often struggle to establish clear segmentation boundaries in images with pixels that have similar gray intensities [8]. In ML and DL methods ([12] and [13]), the segmentation process is based on the classification of image pixels. This process is carried out using ML methods, applying them to features obtained through traditional techniques, while DL implements them internally and automatically derives its features. DL techniques have proven effective in various fields, particularly in medical image processing. In this area, two methodologies stand out ([14] and [15]) as dominant in DL: one employs CNNs that analyze image data through a layer-by-layer feature extraction process; the other takes advantage of the Transformer model, with its capacity to manage sequential data [13]. Most methods in this category offer significant advantages, including insensitivity to the echogenicity of nodules, high-precision automatic segmentation, and efficiency. However, they also present challenges, such as the need for extensive labeled datasets and prolonged training times [8].

Recent studies [16] have focused on the development and use of DL Transformer methods to segment nodule regions from thyroid images, which have led to segmentation models that outperform existing methods. Transformer-based models require larger datasets to effectively grasp the complex patterns and relationships found in medical images. The scarcity of medical data with accurate annotations poses a significant challenge for developing self-attention mechanism-based models. This limitation prevents these models from fully leveraging their potential to learn complex patterns in images. Instead, convolutional neural network (CNN) models tend to perform better in scenarios with limited data, as they handle parameters more efficiently and generalize more easily, all at a considerably lower computational cost [17]. Taking advantage of their strengths in image classification, numerous studies have proposed innovative CNN models that use the U-Net framework for segmentation tasks [12, 18, 19]. The U-Net architecture is commonly used in this area due to its simple structural design, facilitation of generalization through symmetric training, and the ability to learn features at multiple levels thanks to skip connections [12].

However, this method struggles with the unclear and blurred boundaries of nodular regions during the segmentation process and often shows insensitivity to small nodules [13]. This study addressed the complexity of the tissue boundaries surrounding the thyroid and proposed a U-Net model for two-dimensional image segmentation tasks, taking into account the limitations of ultrasound techniques. The proposed model incorporates, as an encoder, a transfer learning model and a combination of border detection filters to apply it to various datasets. The distinct features of the proposed model, in comparison to similar studies in the literature, can be summarized in three main points:

1.- Strengthening of the coding layers through the systematic selection of a competitive transfer learning model.

2.- Improvement in the ability to select more significant features and reduction of overfitting

through the use of border detection filters that highlight edges and reduce noise.

3.- Reduction in computational cost less than that of models that use attention mechanisms when implementing a CNN model as an encoder.

This article demonstrates that accurate segmentation results can be achieved without complex neural architectures or attention mechanisms, as long as border detection filters are carefully selected and an appropriate encoder model is used for the images being segmented.

This paper is structured as follows: Section 2 reviews related work in the field. Section 3 describes the materials and methods we used in our proposed approach, including the selected border detection filters and encoders. In Section 4, we present our experimental results. Finally, we draw conclusions in Section 5.

2 Related Work

This review synthesizes recent advancements in thyroid nodule segmentation using deep learning (DL) techniques. Direct comparison across studies is challenging due to the frequent use of private datasets ([8] and [17]). To establish an unbiased comparative framework, only studies utilizing the publicly available Digital Database Thyroid Image (DDTI) dataset [20] are considered. Pan et al. [21] introduced SGUNet, a segmentation method that uses a pixel-level semantic map to guide low-level features, thereby improving the accuracy of thyroid nodule representations in ultrasound images. SGUNet achieved a Dice coefficient of 0.6290 and an IoU score of 0.4590. Gong et al. [22] developed TRFE, a network that applies guided attention to the thyroid region for precise segmentation, reporting an IoU of 0.5272 and a Dice score of 0.6904. Radhachandran et al. [23] introduced a multitask approach that integrates an anomaly detection (AD) module for the automatic detection and segmentation of thyroid nodules in ultrasound images. The AD module's effectiveness was evaluated across several state-of-the-art segmentation architectures using the UCLA, DDTI, and Stanford CINE datasets. Unlike previous studies, this research included images without nodules in the evaluation, yielding an average Dice index of 0.5760 and an IoU of 0.4380. Nguyen et al. [24] developed a segmentation method that combines nested and attention-based networks to leverage their respective advantages. This approach employs a suggestion network (SN) and an enhancement network (EN) within a nested architecture. The authors observed that segmentation complexity increased for lesions that were either very small or very large. Their method achieved a Dice index of 0.612, outperforming other approaches, although it required more computational time than alternative segmentation networks. Das et al. [8] identified data preprocessing strategies such as data augmentation, region of interest (ROI) detection, and principal component analysis (PCA) as prevalent methods to mitigate overfitting. All previously discussed studies utilized data augmentation and ROI detection, reflecting the limited size of the DDTI dataset (480 images).

Notably, only Radhachandran et al. [23] applied a filter to remove Gaussian noise. Additional studies have implemented border detection filters on other publicly available ultrasound datasets to further address overfitting. For example, Banerjee et al. [25] applied a bilateral filter to enhance nodule edges and used histogram equalization for contrast adjustment. Li et al.[26] employed a median filter and contrast modification to highlight edges. Poormina et al. [30] combined multiple Khuan border detection filters, histogram equalization, and a Canny filter for edge detection to support nodule classification. Yadav et al. [31] performed a comparative analysis of border detection filters to assist radiologists in evaluating nodule malignancy by emphasizing edges and reducing noise in ultrasound images. The adoption of transfer learning models in computer vision applications has increased substantially in recent years. Prochazka & Zeman [27] employed a ResNet model as an encoder within a U-Net architecture to segment nodules across three distinct ultrasound image datasets. In T. Banerjee et al. [25] developed a module that integrates MobileNetV2 and VGG16 transfer learning models with an attention mechanism for thyroid nodule segmentation. Hu et al. [28] introduced a segmentation model with a dual-branch network: one branch uses a ResNet model, while the other incorporates an attention model called Mamba.

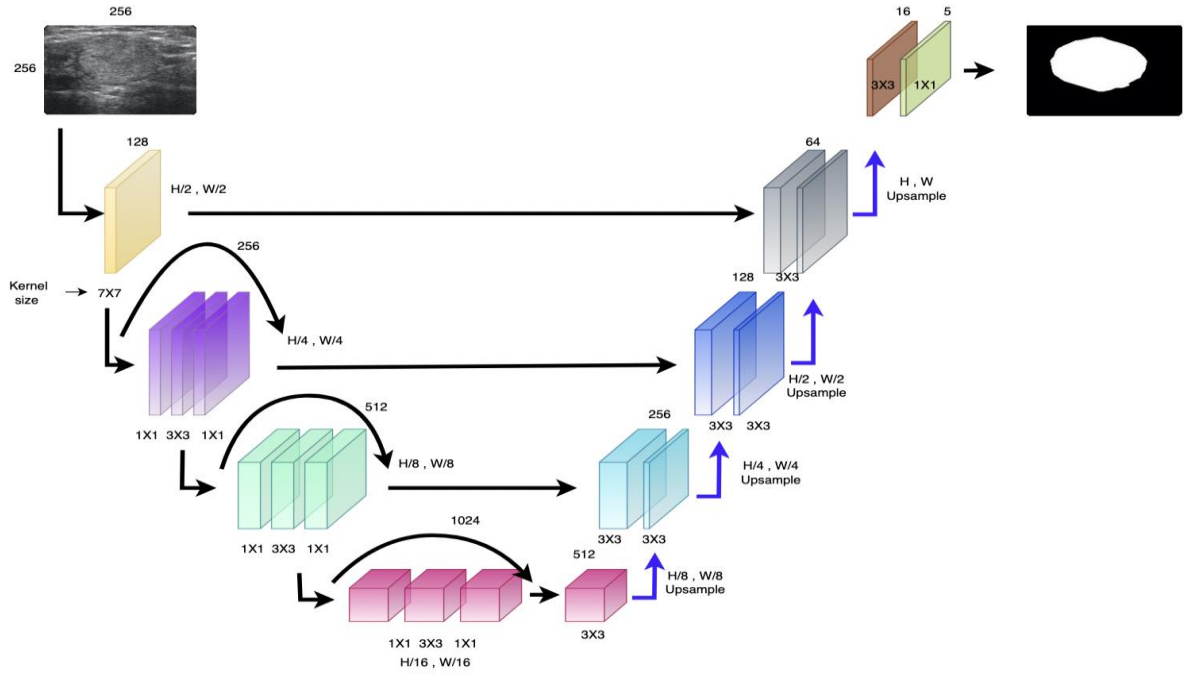

Fig. 1. U-Net Model block diagram

3 Materials and Methods

This section describes the materials and methods used in the experimental evaluation of the thyroid nodule segmentation model. The image dataset, Digital Database Thyroid Image (DDTI) [20], is introduced, with details on preprocessing and the experimental filter selection process to address overfitting. Subsequent subsections discuss the selection of the transfer learning model as the encoder for the U-Net architecture and the loss function used in the experiments. The section concludes with a description of the metrics employed to assess model performance.

3.1 Digital Database Thyroid Image (DDTI) Datasets

The DDTI dataset, curated by Pedraza et al. [20], is an open-access resource designed for training radiologists and developing algorithms for thyroid nodule analysis. It comprises 390 XML files, each corresponding to a case: 91 normal cases (thyroids without nodules), 52 benign nodules, and 247 malignant nodules. Each case includes detailed descriptions and radiologist diagnoses. Multiple ultrasound images (one to three per patient) are available per case, each with a resolution of 560 x 360 pixels.

Preprocessing steps included removing images without masks and corrupt XML files, and generating corresponding masks from the XML files. The final dataset consists of 466 images and 466 masks, each at 560 x 360 pixels. For experimental purposes, the dataset was partitioned into 80% (374 samples) for training, 10% (46 samples) for testing, and 10% (46 samples) for validation. To reduce GPU workload, images were resized to 256 x 256 pixels and pixel values were normalized to the range [0, 1]. Padding or cropping was not applied, as the minimal data loss from these adjustments was deemed negligible for model performance.

3.2 U-NET Architecture

U-Net is a convolutional neural network architecture characterized by a "U" shape, consisting of a contracting path (encoder) that captures image context and an expanding path (decoder) that enables precise localization [29]. Designed primarily for semantic image segmentation, U-Net is effective in applications requiring pixel-level predictions, such as medical, satellite, and microscopy images.

The architecture [41] includes a contraction route (left section of Figure 1) and an expansion route (right section of Figure 1). The contraction path follows the standard convolutional network design, applying two 3x3 convolutions (without padding), each followed by a rectified linear unit (ReLU) and a 2x2 max pooling operation with a stride of 2 for subsampling. At each subsampling step, the number of feature channels is doubled. Each step in the expansion path involves upsampling the feature map, followed by a 2x2 convolution (up-convolution) that halves the number of feature channels, concatenation with the corresponding cropped feature map from the contraction path, and two 3x3 convolutions, each followed by a ReLU.

The final layer uses a 1x1 convolution to assign each 64-component feature vector to the target class. The network contains 23 convolutional layers in total. In this study, a transfer learning model serves as the encoder (see Section 3.3), resulting in a variable number of convolutional layers based on the selected base architecture. The decoder mirrors the encoder path and utilizes up-convolutions.

3.3 Transfer Learning

Transfer learning is a subfield of machine learning that applies knowledge acquired from one problem to related tasks [30]. This approach allows models to leverage prior learning, leading to faster training and improved performance with less data.

A key advantage is that only part of the trained model requires further adaptation for new tasks [31]. For this study, state-of-the-art computer vision models were selected. The following subsections briefly describe each model considered.

3.3.1 MobileNet

MobileNet is a series of efficient, lightweight convolutional neural networks developed by Google, primarily designed for computer vision tasks on devices with limited resources, such as smartphones and embedded systems. Its main innovation is the employment of depthwise separable convolutions, which significantly reduce computational costs and model size compared to standard CNNs. This process splits a regular convolution into two smaller, more efficient operations:

- Depthwise convolution: Applies a single filter to each input channel independently, filtering the input.
- Pointwise convolution: A 1x1 convolution that linearly combines the outputs from the depthwise step to generate new features across channels. This factorization results in roughly 8 to 9 times fewer computations with only a slight reduction in accuracy.

The MobileNet family has evolved through several versions, each introducing further improvements:

- MobileNetV1 (2017) [32]: Introduced the core concept of depthwise separable convolutions, along with hyperparameters (width and resolution multipliers), to balance performance and efficiency.
- MobileNetV2 (2018) [33]: Introduced inverted residual blocks with linear bottlenecks. This structure helps preserve information flow and improve accuracy while maintaining efficiency.
- MobileNetV3 (2019) [34]: Utilized hardware-aware neural architecture search (NAS) and other techniques, such as the hard swish activation function, to further optimize performance and efficiency for specific mobile phone CPUs.

3.3.2 VGGNET

VGGNet (Visual Geometry Group Network), introduced in 2014 [35], is a deep convolutional neural network architecture recognized for its effectiveness and simplicity in computer vision tasks such as image classification and object recognition. VGGNet reduces the spatial

dimensions of the output through a sequence of convolutional layers with small border detection filters (3x3). As the network becomes more complex, the number of border detection filters increases. Designing the architecture typically involves selecting the number of convolutional layers and the number of border-detection filters per layer. The most widely used architectures are VGG16 and VGG19, with 16 and 19 layers, respectively. VGGNet has demonstrated exceptional performance on various benchmark datasets, including ILSVRC, where it ranked among the top models

3.3.3 DenseNet

Every layer is linked to others through the Dense Convolutional Network (DenseNet) [36]. It addresses the vanishing gradient issue, significantly reduces the number of parameters, improves feature flow, encourages feature reuse, and increases efficiency. DenseNet enhances the depth, accuracy, and training efficiency of convolutional networks by establishing short connections between layers near the input and near the output.

3.3.4 ResNet

ResNet (Residual Network) is a deep learning architecture introduced in 2015 by Microsoft Research [37]. It was specifically designed to address the vanishing gradient problem in very deep neural networks. This convolutional neural network (CNN) addresses challenges in training deep networks, as increasing depth can lead to diminishing gradients that hinder training. ResNet introduces a residual block, which enables the network to bypass layers that do not significantly contribute to the final output, thereby maintaining gradient flow. A residual block in ResNet consists of two convolutional layers, with a shortcut connection that adds the input to the block's output. After this addition, a non-linear activation function, such as ReLU, is applied.

3.3.5 EfficientNet

EfficientNet is a family of highly efficient and accurate convolutional neural networks (CNNs) that employs a systematic compound scaling method to balance the network's depth, width, and input resolution [38]. The fundamental aspects of the model are as follows:

- Compound Scaling: Traditional methods scaled only one dimension (depth, width, or resolution) at a time, which often led to diminishing returns in accuracy. EfficientNet introduces a principled compound scaling method that uses a single compound coefficient (ϕ) to uniformly scale all three dimensions simultaneously:
  - Depth (number of layers).
  - Width (number of channels).
  - Resolution (input image size). This balanced approach maximizes performance for a given computational budget.
- MBConv Blocks: The architecture is based on mobile inverted bottleneck convolution (MBConv) blocks, originally from MobileNetV2. These blocks are highly memory- and computation-efficient, utilizing depthwise separable convolutions and squeeze-and-excitation (SE) modules that help the network focus on important features.

The EfficientNet family includes models from B0 (the baseline) to B7, with each larger model being more accurate and requiring more computational power. This allows users to select a model that best fits their specific hardware constraints and performance needs.

3.3.6 Xception

Xception (Extreme Inception) is a deep convolutional neural network architecture that primarily uses depthwise separable convolutions and residual connections. Proposed by François Chollet of Google in 2016 as an extension of the Inception architecture [39], its fundamental aspects are as follows:

- Depthwise Separable Convolutions: Xception replaces the traditional Inception modules with depthwise separable convolutions. This operation factorizes a standard 3D convolution into two separate steps:

- Depthwise convolution: Applies a single spatial filter to each input channel independently.
- Pointwise convolution (1x1): A 1x1 convolution then combines the outputs across the channels.
- This design choice allows the network to capture spatial and cross-channel correlations separately, which is more parameter-efficient than standard convolutions.
- Extreme Inception Interpretation: The Inception module can be viewed as an intermediate step between standard convolutions and depthwise separable convolutions. Xception extends this concept by assuming that cross-channel and spatial correlations can be entirely decoupled, using the maximum possible number of channels processed independently.
- Residual Connections: The architecture incorporates skip (residual) connections around its main modules, similar to ResNet, which helps accelerate convergence and improve gradient flow in very deep networks.
- Architecture Flow: The network is structured into three main flows: an entry flow, a middle flow repeated eight times, and an exit flow.

In summary, Xception is a powerful and efficient architecture that demonstrates the effectiveness of fully decoupling the processing of spatial and cross-channel information in deep learning.

3.3.7 NASNet-Mobile

NASNet-Mobile is a lightweight, pre-trained convolutional neural network (CNN) designed for efficient image classification on devices with limited computational power, such as mobile devices. Developed by Google researchers using Neural Architecture Search (NAS) [40], an automated machine learning technique that employs reinforcement learning to discover high-performing network structures, its architecture is built on repeating normal cells (which maintain feature map size) and reduction cells (which reduce height and width). The network is pre-trained on more than a million images from the ImageNet database. It is optimized for mobile vision tasks, balancing accuracy and computational efficiency, making it suitable for deployment on mobile devices.

Table 1. Transfer learning models

Model	Parameters	Depth
MobileNetV1	4.3 M	55
MobileNetV2	**3.5 M**	105
MobileNetV3	5.4 M	105
VGG16	138 M	**16**
VGG19	143.7 M	19
EfficientNetB0	5.3 M	132
RestNet50V2	25.6 M	103
DenseNet121	8.1 M	242
Xception	22.9 M	81
NASNetMobile	5.3 M	389

3.4 Selection of the Filter for Data Preprocessing

This stage involved selecting border detection filters for preprocessing data from the DDTI dataset. A base model, as described in Section 3.4.1, was trained. After training and validation, the base model was tested on new data. Results were analyzed qualitatively by comparing ultrasound images with ground truth masks and model predictions and quantitatively using metrics such as the Dice coefficient and IoU to identify the filter that enhances segmentation performance.

3.4.1 Base Model

For this study, less complex transfer learning models were prioritized, considering those with fewer parameters or shallower depth.

Table 2. Filter configurations for data preprocessing and the jobs that implemented them and principal function

Works	Filter	Principal Function
[42]	Kuan	Remove the speckled noise and keep the sounds.
[25]	Bilateral	Smooths without losing edges.
[25]	CLAHE	Adaptive contrast to reduce noise.
[43]	Sobel	Detects and enhances edges.
[23] [43]	Canny	Detects and enhances edges.
[23]	Gaussian Blur	Smooth the image, reduce the noise.
[42]	Histogram Equalization	Improve the contrast
[44]	Fast Bilateral	Smooths without losing edges with less computational cost than the Bilateral filter.

Number of quantized parameters in millions (M), quantized depth by number of layers.

Table 1 shows the size of the models considered for this work, where it is observable that MobileNetV2 has the fewest parameters (3.5 million), but is deep (105 layers); in contrast, VGG16 is the least deep model (16 layers), but one of the models with the most parameters (138 million).

Despite not being the model with the fewest parameters and least depth, MobileNetV1 was considered for use as the encoder in the U-Net architecture for the base model.

It is so due to its balanced configuration, ranking second in the number of parameters (4.3 million) and third in the least depth (55 layers).

3.4.2 Border Detection Filters

Our selection of border detection filters is based on state-of-the-art technology, from which the border detection filters applied in related works have been chosen (see Table 2).

To provide context for the operation of the Below border detection filters, each filter is briefly described along with its mathematical formulation.

The formula for the **Kuan filter** (Kuan [41]) estimates the true signal val ($\hat{z}_{ij}$) of a pixel based on a weighted average of the central pixel's intensity and the local mean within a filter window. The Kuan filter transforms the multiplicative noise model (common in radar images) into an additive noise model for processing.

The filtered pixel value $\hat{z}_{ij}$ at coordinates (i, j) is calculated as:

$$\hat{z}_{ij} = \bar{z} + W \cdot (z_{ij} - \bar{z}), \tag{1}$$

where:

- $\hat{z}_{ij}$ is the Kuan filtered pixel value.
- z_{ij} is the original (unfiltered) value of the central pixel.
- $\bar{z}$ is the local mean of all pixels in the moving window (kernel) centered on (i, j).
- W is the weighting function (weighting factor or coefficient), which is the core of the Kuan filter's adaptivity.

The weighting function W is calculated to balance between smoothing (using the local mean) in uniform areas and preserving edges (using the original pixel value) in high-variance areas.

The formula for the weighting function W is:

$$W = \frac{1 - C_u^2 / C_i^2}{1 + C_u^2}, \tag{2}$$

or sometimes presented as

$$W = \frac{Var_k}{(Var_k + Var_{noise})}. \tag{3}$$

The components of the weight function are:

- C_u: The estimated noise variation coefficient (coefficient of variation of the speckle noise). This is often estimated as $C_u = 1/\sqrt{NLOOKS}$, where $NLOOKS$ is the number of looks for the radar image.

Ci: The image variation coefficient (local coefficient of variation) within the filter window. This is calculated as $C_i = S/\bar{z}$, where S is the local standard deviation of pixel intensities within the window and $\bar{z}$ is the local mean.

- Var_k: The variance of pixels in the window.
- Var_{noise}: The variance of the speckle noise.

The filter adapts because the weight W becomes close to 0 in homogeneous regions (where C_i is low, approaching C_u), resulting in the filtered pixel being close to the local mean (maximum smoothing).

Near edges, C_i is high, making W close to 1, and the filtered pixel value approaches the original pixel value (edge preservation).

The **bilateral filter** (bilateral [45]) replaces a pixel's value with a weighted average of nearby pixels, where weights depend on both spatial and radiometric distance. The formula is expressed as:

$$\mathrm{BF}(I)(p) = \frac{1}{w_p} \sum_{q \in S} G_s(||p-q||) G_r(|I_p - I_q|) I_q, \quad (4)$$

where G_s is a Gaussian for spatial distance, G_r is a Gaussian for radiometric (intensity) difference, I_p is the intensity of the center pixel, I_q is the intensity of a neighbor pixel, and w_p is a normalization factor.

The components of the formula function are:

- p: The center pixel coordinates.
- q: A neighboring pixel's coordinates.
- S: The set of all neighboring pixels within a defined window.
- $G_s(||p-q||)$: A Gaussian function that weights pixels based on their spatial distance from p. The farther a pixel is spatially, the lower its weight.
- $G_r(|I_p - I_q|)$: A Gaussian function that weights pixels based on the absolute difference in intensity or color between the center pixel (I_p) and the neighboring pixel (I_q). Pixels with very different intensities get a low weight, which helps preserve edges.
- I_q: The intensity value of the neighboring pixel.
- $w_p = \sum_{q \in S} G_s(||p-q||) G_r(|I_p - I_q|)$: The normalization factor. This sums all the weights to ensure the final average is correctly scaled.

The formulation of the **Histogram Equalization** [46] involves creating a new pixel value (s) by using the cumulative distribution function (CDF) of the original image's pixel intensities (r). This is mathematically expressed as:

$$s = T(r) = (L-1) * sum(p_r(x)), \quad (5)$$

$$for\ x\ from\ to\ r.$$

Here is the step-by-step formulation:

- Calculate the histogram by counting the frequency of each pixel intensity level in the original image.
- Calculate the normalized histogram (or Probability Density Function, PDF): Divide the frequency of each pixel intensity by the total number of pixels (\(N\)) in the image.
  - $p_r(x) = n_x/N$
  - Where n_x is the number of pixels with intensity x, and N is the total number of pixels.
- Calculate the Cumulative Distribution Function (CDF): Sum the normalized histogram values from the lowest intensity level up to the current one.

$$CDF(x) = sum(p_r(x)), \quad for\ x\ from\ to\ r \quad (6)$$

- Map to new pixel values: Multiply the CDF by the maximum possible pixel intensity level $(L-1)$ and round to the nearest integer to get the new pixel value s.

$$s = round((L-1) * CDF(r)), \quad (7)$$

  - For an 8-bit grayscale image, $L = 256$, so the formula is
  - $s = round((255) * CDF(r))$(8)

Contrast Limited Adaptive Histogram Equalization (CLAHE) [47] is a complex algorithm involving several steps rather than a single simple mathematical formula applied globally to an image. The core formulation relies on local histogram equalization and bilinear interpolation with a contrast limit. The process for a single channel (e.g., grayscale or luminance channel in a color image) can be broken down as follows:

- Image Tiling: The input image is divided into a grid of small, non-overlapping rectangular regions called "tiles" (e.g., 8x8 grid is common).
- Contrast Limiting (Clipping): For each tile, a histogram of pixel intensities is computed. To prevent over-amplification of noise in uniform areas, a "clip limit" is applied to the histogram bins.
  - Any pixel count in a histogram bin that exceeds this limit is clipped to the limit value.
  - The excess pixels are then redistributed evenly across all other bins of the histogram, which ensures the total number of pixels in the tile remains constant.
- Histogram Equalization and CDF Calculation: A cumulative distribution function (CDF) is calculated from the modified (clipped and redistributed) histogram for each tile. This CDF then serves as the transformation function for the pixels within that specific tile.

  For a given pixel intensity value s in a tile with a clipped histogram h, the transformation function (CDF) $T(s)$ is generally formulated as:

$$T(s) = round\left(\frac{C(s) - C_{min}}{M \times N - C_{min}} \times (L-1)\right). \quad (9)$$

  Where:
  - $C(s)$ is the cumulative count of pixels up to intensity s in the clipped histogram.
  - C_{min} is the minimum cumulative count (often 0 after normalization).
  - $M \times N$ is the total number of pixels in the tile (or region size).
  - L is the number of possible gray levels (e.g., 256 for an 8-bit image).

  The result is scaled to the full dynamic range (0 to $L-1$) and rounded to the nearest integer.
- Bilinear Interpolation: To remove artificial boundaries between adjacent tiles (a common artifact of simple adaptive histogram equalization), bilinear interpolation is used. Just one tile's transformation function does not determine the final intensity value for a pixel, but rather the weighted average of the transformation functions of the four nearest tile centers. For a pixel at position P, its final value s' is interpolated using the gray-level mappings of the surrounding tile centers P1, P2, P3, and P4 based on its relative position within the grid:

$$\begin{aligned} s' = {} & (1-\alpha)(1-\beta)g_{P1}(s) \\ & +\alpha(1-\beta)g_{P2}(s) \\ & +(1-\alpha)\beta g_{P3}(s) \\ & +\alpha\beta g_{P4}(s). \end{aligned} \quad (10)$$

  Where α and β are the normalized horizontal and vertical distances of the pixel from the corner of the tile region, and $g_{Pi}(s)$ is the mapping function (CDF) of the respective tile center P_i.

The **Sobel filter** (Sobel [48] or Sobel operator) is a discrete differentiation operator that computes an approximation of the image intensity function's gradient to emphasize edges. It is formulated by applying two 3x3 convolution kernels to a grayscale image, one for the horizontal direction (G_x) and one for the vertical direction (G_y).

- The standard 3x3 Sobel kernels, G_x and G_y, are defined as follows:

$$G_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix}, \quad (11)$$

$$G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix}. \quad (12)$$

  These kernels are designed to combine Gaussian smoothing (the 1, 2, 1 weighting) with differentiation, making the filter more resistant to noise than a simple derivative. The kernels are separable, which can improve computational efficiency. At each pixel (x, y) in the input image I, the Sobel operator computes two derivatives:
  - Horizontal Gradient (G_x): This is calculated by convolving the image patch with the G_x

kernel. It measures changes in intensity along the x-direction (vertical edges).

- Vertical Gradient (G_y): This is calculated by convolving the image patch with the G_y kernel. It measures changes in intensity along the y-direction (horizontal edges).
- The convolution operation at a specific pixel can be expressed as (where * denotes the convolution operation):

$$G_x = kernel_x * I, \tag{13}$$

$$G_y = kernel_y * I. \tag{14}$$

The final output is the magnitude of the gradient at that pixel, which represents the strength of the edge. This magnitude is calculated by combining $G_x$and G_y using one of two common formulas:

- L2 Norm (Euclidean Distance): This is the most common and accurate method:

$$G = \sqrt{G_x^2 + G_y^2}. \tag{15}$$

The result G is the overall edge strength at the pixel.

- L1 Norm (Manhattan Distance): This is a computationally cheaper approximation, often used when performance is critical (e.g., in hardware implementations), as it avoids the square root operation:

$$G = |G_x| + |G_y|, \tag{16}$$

Additionally, the direction of the gradient (the angle of the edge) can be calculated:

$$\Theta = \arctan(G_y, G_x), \tag{17}$$

where Θ is the angle of orientation.

Gaussian blur [49] is an image filtering technique that uses a Gaussian function to create a soft, hazy, and smooth effect by averaging a pixel's value with its neighbors. It is a type of low-pass filter that reduces image noise and sharp details by applying a weighted average to each pixel, with the pixel's value having the heaviest weight and the influence decreasing with distance.

This technique is used to soften images, reduce noise, guide the viewer's eye, and prepare images for downsampling. Use a 2D Gaussian function to generate a convolution kernel, which is then applied to an image to blur it. The formula for the 2D Gaussian function is:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}, \tag{18}$$

where x and y are the pixel coordinates and σ is the standard deviation that controls the amount of blur. To apply the filter, this function calculates the weights for a kernel matrix, and each pixel's new value is a weighted average of its neighbors, with the center pixel receiving the highest weight.

The **Canny filter** [50] is not a single formula, but a multi-stage algorithm that mathematically formulates edge detection based on several steps to optimize for low error rate, good localization, and minimal response to noise. The core mathematical formulations involved are:

- Noise Reduction (Gaussian Smoothing): The image is smoothed using a Gaussian filter to reduce noise, as described by the formula (18).
- Gradient Calculation [51]: The intensity gradients of the smoothed image are calculated using the formulas (11), (12), (13), (14), (15), (16), and (17).

Fast Bilateral filter [52] formulations often approximate the original filter [45] by reinterpreting it as a linear convolution. This is achieved by linearizing the range-dependent kernel, computing the convolution for a sparse set of intensity values using the Fast Fourier Transform (FFT), and then interpolating the results to obtain the final filtered image.

Other fast approaches include using 3D FFT with a downsampling strategy or rearranging the filter into a set of functional operators to simplify computations.

3.5 Selection of Transfer Learning Model as an Encoder

The process of selecting a transfer learning model as an encoder parallels the filter selection

approach, except that models are trained on data preprocessed with the chosen filter to mitigate overfitting.

Results were evaluated qualitatively by comparing ultrasound images with ground truth and predicted masks, and quantitatively using metrics such as the Dice coefficient and Intersection over Union (IoU). This methodology enabled the identification of the filter-encoder combination that maximizes segmentation performance while minimizing overfitting.

3.6 Performance Metrics

Model performance in medical image segmentation is typically assessed by comparing predicted results to expert-generated accuracy masks.

Common evaluation metrics include precision, sensitivity, Dice coefficient, and Intersection over Union (IoU).

In this study, performance was specifically evaluated using the Dice coefficient and IoU metrics.

3.6.1 Sørensen-Dice Index (Dice Score, Dice)

The Dice coefficient [53] is a statistical measure that quantifies the overlap between two sets. In the context of image segmentation, it assesses segmentation accuracy by comparing the model's output to the ground-truth segmentation:

$$Dice = \frac{2 \cdot |A \cup B|}{|A| + |B|}, \tag{19}$$

A: Set of pixels segmented by the model.
B: Set of real pixels (ground truth).
$A \cap B$: Intersection (correctly classified pixels).

3.6.2 Intersection over Union (IoU)

The Intersection over Union (IoU) [54] metric is widely used to evaluate the spatial accuracy of segmentation or object detection models.

It quantifies the extent of overlap between the predicted segmentation and the ground truth:

$$IoU = \frac{|A \cap B|}{|A \cup B|}, \tag{20}$$

A: Set of pixels segmented by the model.
B: Set of real pixels (ground truth).
$A \cap B$: Intersection (correctly classified pixels).
$A \cup B$: Union (all predicted pixels are correct or real).

3.7 Loss Function

Training segmentation models for medical images involves unique challenges compared to natural image segmentation, primarily due to data imbalance. This imbalance often occurs between foreground (diseased) and background (non-diseased) samples, as well as between simple and complex cases.

The difficulty of classifying thyroid nodules can result in segmentation failures, and the presence of indistinct nodule edges further complicates the task. These issues are significant in clinical practice and necessitate targeted solutions.

To address these challenges, a Dice loss [55] function is employed to balance foreground and background samples and to reduce the bias present in the cross-entropy loss function during network training, as shown in Formula 21:

$$\mathcal{L}_{DICE} = 1 - \frac{2 \cdot |A \cup B|}{|A| + |B|}. \tag{21}$$

3.8 Setup

Training and testing were performed on the Google Colab platform utilizing NVIDIA Tesla V100 GPUs with 16 GB of memory.

The frameworks used included PyTorch 2.0.1 and TensorFlow 2.19. Model weights were initialized with pre-trained ImageNet encoders.

The Adam optimization algorithm was applied, and training was conducted for 100 epochs.

4 Results

This section presents the experimental results obtained using the DDTI dataset. The analysis is divided into two parts.

The first part compares the performance of the base model with different border filters applied as preprocessing.

The second part evaluates various transfer learning models using the most effective preprocessing method identified in the first part.

ISSN 2007-9737

Table 4. Comparison of the base model's performance along with filter preprocessing

Filter	Time train	Dice	IoU
None	9 min.	0,1251	0,086
Bilateral + CLAHE	52 min.	**0,6500**	**0,5482**
Sobel	62 min.	0,5313	0,4254
Canny	12 min.	0,5879	0,4850
Gaussian Blur	65 min.	0,5207	0,4130
Kuan, Histogram, Canny	**10 min.**	0,5759	0,4635
Fast Bilateral	65 min.	0,6387	0,5280

Highlight the best results in bold

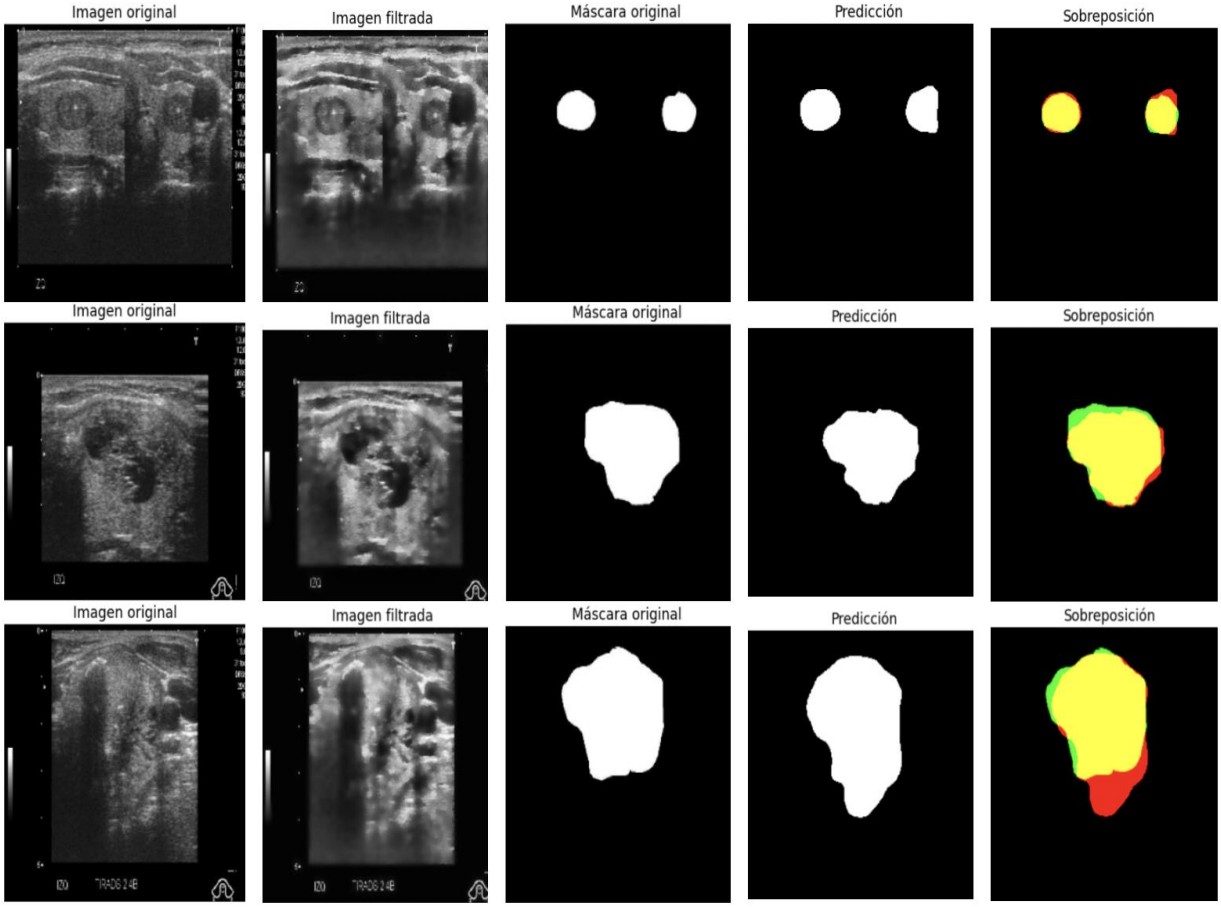

Fig. 2. Nodule segmentation results using the base model (U-Net with MobileNet encoder) and the most successful filter configuration in performance (Bilateral + CLAHE)

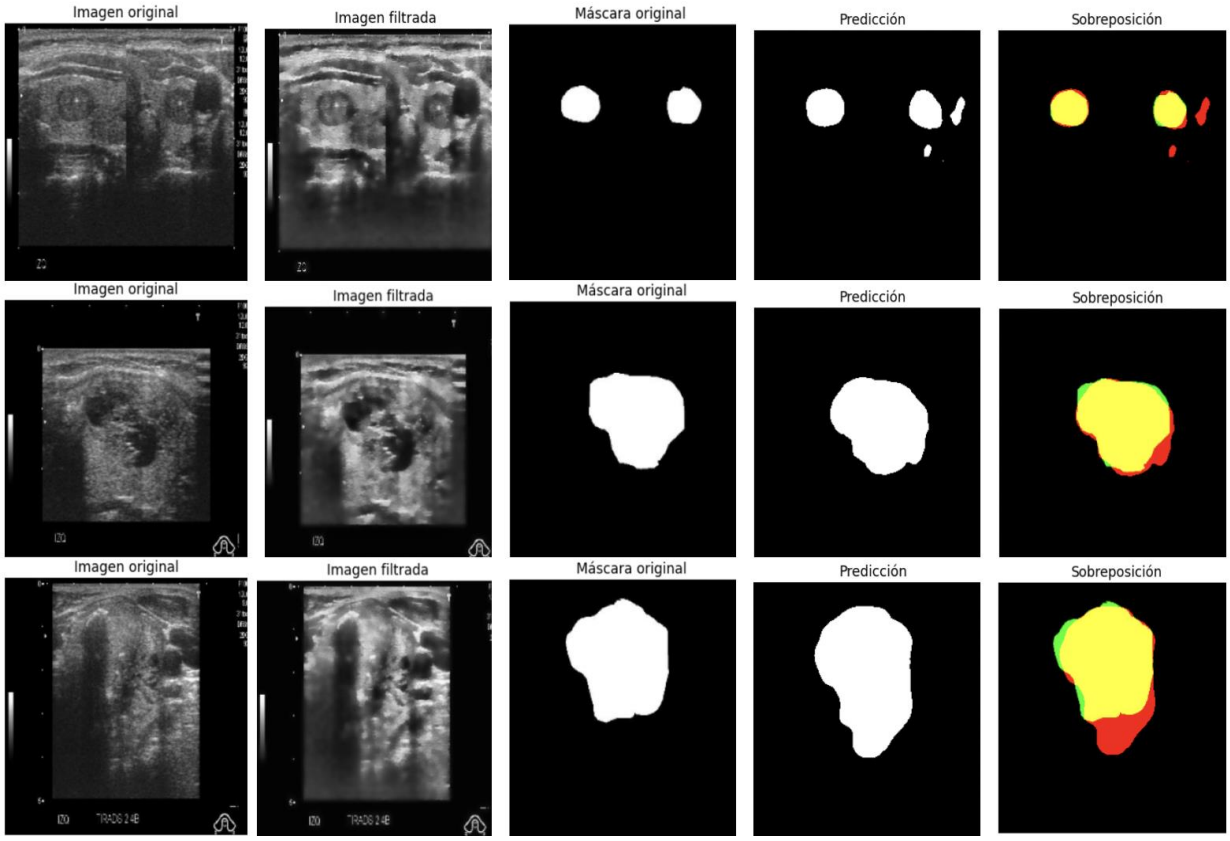

Fig. 3. Nodule segmentation results using the model with the best-performing encoder (U-Net with Xception encoder) and the preprocessing setup (Bilateral + CLAHE) deliver more effective treatment. Diagnostic accuracy may decrease with prolonged work due to fatigue. In this study, an automatic segmentation algorithm based on the U-Net architecture is proposed to delineate thyroid nodule edges from ultrasound images, thereby facilitating detection. The approach provides a comprehensive segmentation method that requires no user interaction and is broadly applicable. The entire image is processed by the segmentation model, and results are generated as a binary mask. The algorithm's segmentation results closely approximate those of clinical experts, supporting its potential application in clinical practice. However, the study is limited by challenges in acquiring and labeling ultrasound images. The dataset is relatively small due to the difficulty of obtaining and annotating ultrasound images. Additionally, unlike other studies that trained and tested on multiple datasets to improve generalizability, only the DDTI dataset was used. The approach also has limitations related to high training time, necessitating the use of a high-performance GPU. Once training is complete, however, the model's predictions are rapid and accurate. Finally, in contrast to the models used for comparison, this approach does not include cropping of the region of interest or data augmentation

4.1 Comparison of Performance among Different Border Detection Filters

The base model, consisting of a U-Net architecture with MobileNetV1 as the encoder, was evaluated on the test set to facilitate objective performance comparison. Each filter configuration listed in Table 2 was applied during training and testing with the dataset.

Experimental results are summarized in Table 4. The baseline model achieved optimal performance when the Bilateral + CLAHE filter configuration was used for data preprocessing, reaching an IoU of 0.5482 and a Dice coefficient of 0.6500, with an average training time of 52

minutes. Figure 2 displays segmentation results for three nodules of different sizes, with the final column showing the overlay (yellow) between the ground truth mask (green) and the model's prediction (red). This visualization demonstrates that the model's segmentation results closely approximate the ground truth across various nodule sizes.

4.2 Performance Comparison between Different Models Applying the Best Preprocessing

To objectively compare the benefits of using the Bilateral + CLAHE filter setup as preprocessing, we trained additional transfer learning models were evaluated as encoders within the U-Net architecture.

Each model listed in Table 1 was trained and tested using the DDTI dataset, with results summarized in Table 5. The Xception model, used as an encoder and combined with Bilateral + CLAHE border detection filters for preprocessing, achieved an IoU of 0.5824 and a Dice score of 0.6938, with an average training time of 88 minutes.

Figure 3 presents segmentation results for three nodules of varying sizes, demonstrating the most effective encoder and preprocessing configuration for nodule segmentation in the DDTI dataset.

Table 6 compares the performance of models from the literature with the results obtained using the most successful configuration in these experiments.

The Dice and IoU values are comparable to those of TRFE (Dice = 0.6904 ± 0.0334, IoU = 0.5272 ± 0.0170), and surpass those of SGUNet (Dice = 0.6290 ± 0.0415, IoU = 0.4590 ± 0.0212), U-Net Multiscale AD module (Dice = 0.5270 ± 0.325, IoU = 0.4120 ± 0.290), and Fusion CNNs + highlight network (Dice = 0.6120).

These results indicate that the segmentation performance of this model exceeds that of other models that did not incorporate border-detection filters or transfer-learning-based encoders within their U-Net architectures.

Table 5. Comparison of the performance of all encoders applying the Bilateral+CLAHE filter configuration as a preprocessing method

Model	Time train	Dice	IoU
MobileNetV1	12 min.	0.5879	0.4849
MobileNetV2	11 min.	0.4723	0.3941
MobileNetV3	4.5 min.	0.1115	0.0105
VGG16	11 min.	0.5800	0.4690
VGG19	16 min.	0.5776	0.4808
EfficientNet B0	8 min	0.2410	0.1658
RestNet50 V2	22 min.	0.5505	0.5661
DenseNet121	359 min.	0.6331	0.5804
Xception	88 min.	**0.6938**	**0.5824**
NASNetMobile	31 min.	0.5016	0.5057

Highlight the best results in bold

5 Discusion

Segmentation of thyroid nodules represents the initial step in evaluating thyroid disorders. Accurate diagnosis is possible only when the location, size, and other relevant information about the nodules are identified.

This process enables clinicians to conduct more comprehensive diagnoses and deliver more effective treatment. Diagnostic accuracy may decrease with prolonged work due to fatigue. In this study, an automatic segmentation algorithm based on the U-Net architecture is proposed to delineate thyroid nodule edges from ultrasound images, thereby facilitating detection.

The approach provides a comprehensive segmentation method that requires no user interaction and is broadly applicable.

Table 6. Comparison of the performance between literature works and the configuration of the best encoder, as well as preprocessing for thyroid nodule segmentation, applied to the DDTI dataset

	Model	Dice	IoU
[22]	TRFE	0.6904 ± 0.0334	0.5272 ± 0.0170
[21]	SGUNet	0.6290 ± 0.0415	0.4590 ± 0.0212
[23]	U-Net Multiscale AD module	0.5270 ± 0.325	0.4120 ± 0.290
[24]	Fusion CNNs and highlighting network	0.6120	NA
Ours	Xception + Filter	**0.6938**	**0.5824**

The entire image is processed by the segmentation model, and results are generated as a binary mask. The algorithm's segmentation results closely approximate those of clinical experts, supporting its potential application in clinical practice. However, the study is limited by challenges in acquiring and labeling ultrasound images. The dataset is relatively small due to the difficulty of obtaining and annotating ultrasound images.

Additionally, unlike other studies that trained and tested on multiple datasets to improve generalizability, only the DDTI dataset was used. The approach also has limitations related to high training time, necessitating the use of a high performance GPU. Once training is complete, however, the model's predictions are rapid and accurate. Finally, in contrast to the other models used for comparison, this approach does not include cropping of the region of interest or data augmentation.

6 Conclusions

A model for segmenting thyroid nodules in two-dimensional ultrasound images using a U-Net architecture is presented. This model incorporates a transfer-learning-based encoder, which improves segmentation accuracy even with a limited dataset. A filter-based preprocessing step is also introduced to reduce overfitting and enhance model performance.

Experimental results demonstrate that this model outperforms most referenced models across all evaluated metrics. In future work, precise segmentation of thyroid nodules may serve as a foundation for computer-aided diagnosis, including automatic assessment of nodule status and classification. Further research will explore additional preprocessing strategies, the development of more robust loss functions, and the integration of more training examples, including other public datasets in addition to the DDTI dataset.

Author Contributions

All authors contributed to the study's conception and design. Material preparation, data processing, model implementation, and analysis were conducted by Christopher Gutierrez and Fernando Gaxiola. All authors participated in analyzing the results and developing new ideas. Christopher Gutierrez drafted the first version of the manuscript, and all authors provided feedback on previous drafts. All authors reviewed and approved the final manuscript.

Acknowledgments

The authors express their gratitude for the support received in various areas from Faculty of Engineering and Faculty of Medicine and Biomedical Sciences of Universidad Autonoma de Chihuahua. Christopher Gutierrez received a scholarship provided by Conahcyt.

References

1. **Baimukashev, R., Kadyrov, S., Turan, C. (2024).** Systematic Survey of Deep Fuzzy Computer Vision in Biomedical Research. Fuzzy Information and Engineering, Vol.

16, No. 3, pp. 220–243. doi: 10.26599/FIE.2024.9270043.

2. **Banerjee, T., Singh, D.P., Swain, D. (2025).** A Novel Hybrid Deep Learning Approach Combining Deep Feature Attention and Statistical Validation for Enhanced Thyroid Ultrasound Segmentation. Scientific Reports, Vol. 15, No. 1. doi: 10.1038/s41598-025-12602-6.
3. **Biradar, N., Dewal, M.L., Rohit, M. (2016).** Blind Source Parameters for Performance Evaluation of Despeckling Filters. doi: 10.1155/2016/3636017.
4. **Canny, J. (1986).** A Computational Approach to Edge Detection. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-8, No. 6, pp. 679–698. doi: 10.1109/TPAMI.1986.4767851.
5. **Carson, P.L., Fenster, A. (2009).** Anniversary Paper: Evolution of Ultrasound Physics and the Role of Medical Physicists and the AAPM and Its Journal in That Evolution. doi: 10.1118/1.2992048.
6. **Zhang, C., Deng, X., Ling, S.H. (2024).** Next-Gen Medical Imaging: U-Net Evolution and the Rise of Transformers. Sensors. doi: 10.3390/s24144668.
7. **Chollet, F. (2017).** Xception: Deep Learning with Depthwise Separable Convolution. IEEE Conference on Computer Vision and Pattern Recognition (CVPR). doi: 10.1109/CVPR.2017.195.
8. **Das, D., Iyengar, M.S., Majdi, M.S. (2024).** Deep Learning for Thyroid Nodule Examination: A Technical Review. Artificial Intelligence Review, pp. 1–20. doi: 10.1007/s10462-023-10635-9.
9. **Diáz, J.E.M. (2020).** Deep artificial vision applied to the early identification of non-melanoma cancer and actinic keratosis. Computación y Sistemas, Vol. 24, No. 2, pp. 751–766. doi: 10.13053/CyS-24-2-2901.
10. **Dice, L.R. (1945).** Measures of the amount of ecologic association between species. Ecology, Vol. 26, No. 3, pp. 297–302. doi: 10.2307/1932409.
11. **Gavaskar, R.G., Chaudhury, K.N. (2018).** Fast Adaptative Bilateral Filterering. IEEE Transactions on Image Processing, pp. 1–12. doi: 10.1109/TIP.2018.2871597.
12. **Bulusu, G., Vidyasagar, K.E.C., Mudigondal, M. (2025).** Cancer Detection Using Artificial Intelligence: A Paradigm in Early Diagnosis. Archives of Computational Methods in Engineering, pp. 2365–2403. doi: 10.1007/511831-024-10209-0.
13. **Gesing, A. (2015).** The Thyroid Gland and the Process of Aging. Thyroid Research, Vol. 8, No. Suppl. 1, pp. A8. doi: 10.1186/1756-6614-8-S1-A8.
14. **Gong, H., Chen, G., Wang, R. (2021).** Multi-Task Learning for Thyroid Nodule Segmentation with Thyroid Region Prior. Proceedings of the International Symposium on Biomedical Imaging, Vol. 2021-April, pp. 257–261. doi: 10.1109/ISBI48211.2021.9434087.
15. **Haribabu, K., Prasath, R., Praveen, J.I. (2025).** MLRT-UNet: An Efficient Multi-Level Relation Transformer Based U-Net for Thyroid Nodule Segmentation. Computer Modeling in Engineering & Sciences, Vol. 143, pp. 414–448. doi: 10.32604/cmes.2025.059406.
16. **He, K., Zhang, X., Ren, S. (2015).** Deep Residual Learning for Image Recognition. doi: 10.48550/arXiv.1512.03385.
17. **Hernández-Herrera, P., Abonza, V., Sánchez-Contreras, J. (2023).** Deep learning-based classification and segmentation of sperm head and flagellum for image-based flow cytometry. Computación y Sistemas, Vol. 27, No. 4, pp. 1133–1145. doi: 10.13053/CyS-27-4-4772.
18. **Howard, A.G., Zhu, M., Chen, B. (2017).** MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications.
19. **Hu, M., Zhang, Y., Xue, H. (2024).** Mamba- and ResNet-Based Dual-Branch

Network for Ultrasound Thyroid Nodule Segmentation. Bioengineering, Vol. 11, No. 10. doi: 10.3390/bioengineering11101047.

20. **Huang, G., Liu, Z., van der Maaten, L. (2018).** Denselye Connected Convolutional Networks. doi: 10.48550/arXiv.1608.06993.
21. **Huérfano, Y., Vera, M., Del Mar, A. (2016).** Imagenología médica: fundamentos y alcance resumen. AVFT Archivo Venezolano de Farmacología y Terapéutica, Vol. 35, pp. 71–76.
22. **Huisman, M., Plaat, A., van Rijn, J.N. (2024).** Understanding Transfer Learning and Gradient-Based Meta-Learning Techniques. Machine Learning, Vol. 113, No. 7, pp. 4113–4132. doi: 10.1007/s10994-023-06387-w.
23. **International Agency for Research on Cancer. (2025).** Global Cancer Observatory.
24. **Jaccard, P. (1912).** The Distribution of the Flora in the Alpine Zone. The New Phytologist, Vol. 11, No. 2, pp. 37–50.
25. **Ahmed, J., Soomrani, M.A.R. (2016).** TDTD: Thyroid Disease Type Diagnostics. 2016 International Conference on Intelligent Systems Engineering (ICISE). doi: 10.1109/INTELSE.2016.7475160.
26. **Koundal, D., Gupta, S., Singh, S. (2016).** Automated Delineation of Thyroid Nodules in Ultrasound Images Using Spatial Neutrosophic Clustering and Level Set. Applied Soft Computing Journal, Vol. 40, pp. 86–97. doi: 10.1016/j.asoc.2015.11.035.
27. **Kuan, D.T., Sawchuk, A.A., Strand, T.C. (1987).** Adaptive Restoration of Images with Speckle. IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. 35, No. 3.
28. **Li, C., Du, R., Luo, Q. (2023).** A Novel Model of Thyroid Nodule Segmentation for Ultrasound Images. Ultrasound in Medicine and Biology, Vol. 49, No. 2, pp. 489–496. doi: 10.1016/j.ultrasmedbio.2022.09.017.
29. **Mahamkali, N., Vadivel, A. (2015).** OpenCV for Computer Vision Applications. Proceedings of National Conference on Big Data and Cloud Computing (NCBDC'15), pp. 52–56.
30. **National Institution Health. (2022).** What Cancer Screening Tests Check for Cancer? NCI.
31. **Nguyen, D.T., Park, K.R. (2022).** Thyroid Nodule Segmentation in Ultrasound Image Based on Information Fusion of Suggestion and Enhancement Networks. Mathematics, Vol. 10, pp. 2–20. doi: 10.3390/math10193484.
32. **Pan, H., Zhou, Q., Latecki, L.J. (2021).** SGUNET: Semantic Guided UNET for Thyroid Nodule Segmentation. Proceedings of the International Symposium on Biomedical Imaging, Vol. 2021-April, pp. 630–634. doi: 10.1109/ISBI48211.2021.9434051.
33. **Paris, S., Kornprobst, P., Tumblin, J. (2009).** Bilateral Filtering: Theory and Applications. Foundations and Trends in Computer Graphics and Vision, Vol. 4, No. 1, pp. 1–73. doi: 10.1561/0600000020.
34. **Patel, O., Maravi, P.S.Y., Sharma, S. (2013).** A Comparative Study of Histogram Equalization Based Image Enhancement Techniques for Brightness Preservation and Contrast Enhancement. Signal & Image Processing: An International Journal, Vol. 4, No. 5, pp. 11–25. doi: 10.5121/sipij.2013.4502.
35. **Pedraza, L., Vargas, C., Narváez, F. (2015).** An Open Access Thyroid Ultrasound Image Database. 10th International Symposium on Medical Information Processing and Analysis, Vol. 9287, pp. 92870W. doi: 10.1117/12.2073532.
36. **Pizer, S.M., Johnston, R.E., Ericksen, J.P. (1990).** Contrast-Limited Adaptive Histogram Equalization: Speed and Effectiveness. Proceedings of the First Conference on Visualization in Biomedical Computing, pp. 337–345. doi: 10.1109/VBC.1990.109340.

37. **Poornima, D., Karegowda, A.G., Pushpalatha, K.R. (2021).** Design of a Fuzzy Inference Based Ultrasound Image Analysis System for Differential Diagnosis of Thyroid Nodules. 3rd International Conference on Integrated Intelligent Computing Communication & Security (ICIIC 2021), pp. 269–277. doi: 10.2991/ahis.k.210913.034.
38. **Prochazka, A., Zeman, J. (2025).** Thyroid Nodule Segmentation in Ultrasound Images Using U-Net with ResNet Encoder: Achieving State-of-the-Art Performance on All Public Datasets. AIMS Medical Science, Vol. 12, No. 2, pp. 124–144. doi: 10.3934/medsci.2025009.
39. **Purwono, Ma'arif, A., Rahmaniar, W. (2022).** Understanding of Convolutional Neural Network (CNN): A Review. International Journal of Robotics and Control Systems, Vol. 2, No. 4, pp. 739–748. doi: 10.31763/ijrcs.v2i4.888.
40. **Qian, S., Hu, Y., Ning, C. (2021).** MobileNetV3 for Image Classification. IEEE Xplore, pp. 490–498.
41. **Radhachandran, A., Kinzel, A., Chen, J. (2024).** A Multitask Approach for Automated Detection and Segmentation of Thyroid Nodules in Ultrasound Images. Computers in Biology and Medicine, Vol. 170. doi: 10.1101/2023.01.31.23285223.
42. **Sandler, M., Howard, A., Zhu, M. (2018).** MobileNetV2: Inverted Residuals and Linear Bottlenecks. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4510–4520. doi: 10.1109/CVPR.2018.00474.
43. **Savelonas, M.A., Iakovidis, D.K., Legakis, I. (2009).** Active Contours Guided by Echogenicity and Texture for Delineation of Thyroid Nodules in Ultrasound Images. IEEE Transactions on Information Technology in Biomedicine, Vol. 13, No. 4, pp. 519–527. doi: 10.1109/TITB.2008.2007192.
44. **Simonyan, K., Zisserman, A. (2015).** Very Deep Convolutional Networks for Large-Scale Image Recognition. ICLR 2015, pp. 1–14. doi: 10.48550/arXiv.1409.1556.
45. **Song, K., Feng, J., Chen, D. (2024).** A Survey on Deep Learning in Medical Ultrasound Imaging. Frontiers in Physics, Vol. 12, pp. 1–21. doi: 10.3389/fphy.2024.1398393.
46. **Tan, M., Le, Q.V. (2020).** EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. 36th International Conference on Machine Learning. doi: 10.48550/arXiv. 1905.11946.
47. **Vaswani, A., Brain, G., Shazeer, N. (2017).** Attention Is All You Need. 31st Conference on Neural Information Processing Systems (NIPS 2017).
48. **Vincent, O., Folorunso, O. (2009).** A Descriptive Algorithm for Sobel Image Edge Detection. Proceedings of the 2009 InSITE Conference. doi: 10.28945/3351.
49. **Xu, Y., Xu, M., Geng, Z. (2025).** Thyroid Nodule Classification in Ultrasound Imaging Using Deep Transfer Learning. BMC Cancer, Vol. 25, No. 1. doi: 10.1186/s12885-025-13917-3.
50. **Yadav, N., Dass, R., Virmani, J. (2022).** Despeckling Filters Applied to Thyroid Ultrasound Images: A Comparative Analysis. Multimedia Tools and Applications, Vol. 81, No. 6, pp. 8905–8937. doi: 10.1007/s11042-022-11965-6.
51. **Yang, W.T., Ma, B.Y., Chen, Y. (2024).** A Narrative Review of Deep Learning in Thyroid Imaging: Current Progress and Future Prospects. Quantitative Imaging in Medicine and Surgery, Vol. 14, No. 2, pp. 2069–2088. doi: 10.21037/qims-23-908.
52. **Yeung, M., Sala, E., Schönlieb, C.B. (2022).** Unified Focal Loss: Generalising Dice and Cross Entropy-Based Losses to Handle Class Imbalanced Medical Image Segmentation. Computerized Medical Imaging and Graphics, Vol. 95. doi: 10.1016/j.compmedimag.2021.102026.
53. **Zhao, J., Zheng, W., Zhang, L. (2012).** Segmentation of Ultrasound Images of Thyroid Nodule for Assisting Fine Needle

Aspiration Cytology. Health Information Science and Systems, Vol. 1.

54. **Zheng, Y., Xu, Z., Wang, X. (2022).** The Fusion of Deep Learning and Fuzzy Systems: A State-of-the-Art Survey. IEEE Transactions on Fuzzy Systems, Vol. 30, No. 8, pp. 2783–2799. doi: 10.1109/TFUZZ.2021.3062899.
55. **Zoph, B., Vasudevan, V., Shlens, J. (2018).** Learning Transferable Architecture for Scalable Image Recognition. doi: 10.48550/arXiv. 1707.07012.

Article received on 31/10/2025; accepted on 15/12/2025.
**Corresponding author is Fernando Gaxiola.*