IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 15, No. 6, December 2026, pp. 5464~5473
ISSN: 2252-8938, DOI: 10.11591/ijai.v15.i6.pp5464-5473

Ensemble method for lung parenchyma segmentation from a computed tomography image

Asharani Ramadas, Chidananda Murthy Melekote Vinayakamurthy
Department of Electronics and Communication Engineering, Sri Siddhartha Academy of Higher Education, Tumkur, India

Article Info

Article history:

Received Sep 25, 2025
Revised Aug 10, 2026
Accepted Aug 31, 2026

Keywords:

Bagging
Boosting
Computed tomography image
Ensemble method
Segmentation
Stacking

ABSTRACT

Deep learning models are data-driven models that often learn features directly from the training images. The generalizability of these models decreases for small datasets. Machine learning models are understandable and capable of working effectively on small datasets. Lung parenchyma segmentation is a preprocessing stage for lung nodule segmentation. Ensemble methods such as bootstrap aggregating (Bagging), boosting, and stacking are used to segment the lung parenchyma from lung computed tomography (CT) images. The boosting ensemble method outperforms Bagging and stacking methods with an accuracy of 99.64% and 100% lung coverage.

This is an open access article under the CC BY-SA license.

Corresponding Author:

Asharani Ramadas
Department of Electronics and Communication Engineering, Sri Siddhartha Academy of Higher Education
Tumkur, India
Email: asha.srp@gmail.com

1. INTRODUCTION

Lung cancer is the second-highest diagnosed cancer in the world, with an 11.4% growth rate and a leading mortality rate of 18% [1]. Low-dose computed tomography (LDCT) is used as one of the tools to screen for lung cancer; it has been proven that the mortality rate due to lung cancer can be reduced by 24% to 33% for men and women, respectively [2]. Lung cancer detection, lung nodule localization and quantification, lung volume measurement, reconstruction of 3D lung structure from 2D slices, and treatment planning require quantitative and visual analysis of lung parenchyma; hence, parenchyma segmentation from lung computed tomography (CT) images plays a crucial role in CT image analysis.

Lung parenchyma segmentation removes the unwanted parts (bone, heart, and fat) from the lung CT image; it increases the performance of the computer-aided design (CAD) system by extracting the valid features from the image. It also reduces the computational complexity by preserving the dominant features from the lung CT image in diagnosis and treatment planning. Medical image segmentation using machine learning can classify each pixel into a particular class and is capable of handling complex pathology while specifying the abnormalities; hence, it is a more prominent method in medical image processing and contributes to diagnostic systems.

The lung dense attention network (LDANet) was proposed to segment the lung parenchyma with a dice similarity coefficient (DSC) of 98% on the lung image database consortium (LIDC)-image database resource initiative (IDRI) and COVID-19 CT image dataset [3]. An improved UNet architecture was proposed to segment CT images with small and blurry regions from the open-source imaging consortium pulmonary fibrosis progression dataset and achieved a mean DSC of 0.9329 and an accuracy of 0.9885 [4]. Recurrent residual UNet (R2UNet) segments the CT image with a dice of >0.96, a specificity of >0.98, and a

mean intersection over union (IoU) of >0.94 [5]. Attention and circular cross-feature U-Net (ACX-UNet) is a UNet variant enhanced with the convolutional block attention module to reduce the information loss and cyclic cross-spatial pyramid pooling to suppress the background noise.

Generalization is achieved by the pretrained visual geometry group 16 (VGG16), by providing more affluent semantic features [6]. Mask region-based convolutional neural network (R-CNN) with a residual network 50 (ResNet50) backbone segments the lung parenchyma with a DSC of 93.4% on the COVID-19 dataset [7]. Bidirectional encoder representation from image transformers (BEiT), densely connected convolutional network (DenseNet), and sequential convolutional neural network (Sequential CNN) are assembled for lung cancer prediction [8]. The ensemble method combines the weak learners to build a strong, accurate prediction model. Adaptive boosting (AdaBoost) performs well in noise-free data sets, while bootstrap aggregating (Bagging) can handle noisy datasets. Randomization is suitable for large data sets [9].

Bagging, boosting, rotation forest, and gradient boosting are tree-based ensemble methods. Ensemble methods improve the accuracy, reliability, and generalizability of the model compared to a single machine learning model [10], [11]. A deep learning-based ensemble architecture was proposed to segment the medical image. The first layer, consisting of multiple deep learning models trained with original images, results in probability maps, and the second layer, trained with augmented images (image + probability maps), results in the refined prediction fused by a weight-optimized linear combiner to generate the segmented image [12]. One hundred deep convolutional neural networks (DCNNs) are ensemble to perform cell segmentation. Class membership probabilities of 100 DNNs are aggregated by mean, median, alpha-stable distribution location parameter, and majority voting.

This proposed method is suitable for a real-time clinical environment [13]. The stacking ensemble method was constructed using a light gradient boosting machine and support vector machine as base learners and extreme gradient boosting as a meta-learner for pixel classification in image segmentation. This model is an efficient, robust, and accurate model with segmentation coverage of 0.63 [14]. Ensemble methods enhance the performance of the single models; hence, to overcome the uninterpretability of the deep learning models, this study proposed ensemble methods to segment the lung parenchyma from the lung CT image.

2. METHOD

The ensemble approach combines classifiers to improve their efficiency. In this work performed lung parenchyma segmentation from lung CT images by using boosting, Bagging, and stacking ensemble methods to recognize the strong ensemble method among them. The qualitative and quantitative study of these ensemble methods reveals the strong ensemble method used to develop the front-end application for automatic lung parenchyma segmentation.

2.1. Boosting

Boosting is one of the ensemble techniques that can average the predictions of different base learners by voting before final predictions. Gradient boosting, AdaBoost, extreme gradient boosting (XGBoost) algorithm, and light gradient boosting machine are key boosting algorithms [15]. A strong prediction is obtained by combining the predictions of base learners [16].

In this proposed work use XGBoost, a powerful ensemble learning method constructed by 150 collective decision trees. The weighted combination of all the trees is used to form the strong classifier. The preceding tree reduces the error by focusing on the misclassified pixels from the prior tree. Complex features of the CT image are captured by keeping the depth at 6 and the balanced learning rate equal to 0.1.

Figure 1 illustrates the ensemble method using the XGBoost classifier. Image features, namely pixel intensity, gradient, texture, and spatial positions as shown in (1)-(5), are used to obtain the feature matrix. Pixel intensities are used to differentiate the air, bone, and tissue structure from the CT image. Pixel gradient calculated by the Sobel operator to enhance the edges in the image. Local binary patterns are utilized to identify the recurring tissue structure within the image. Normalized row and column coordinates define the spatial positions of the pixels to reduce false positives.

A strong classifier uses this feature matrix to provide the final prediction. XGBoost makes the initial prediction by logarithmic addition of the class distribution to build a baseline model. First-order and second-order derivatives of the log loss function are computed. The decision tree with maximum depth equal to 6 is built to fit the gradients and Hessians, and then predictions are updated. Each newly added tree corrects the model's error by minimizing the log loss; all 150 trees are an ensemble. The final prediction is the additive combination of all 150 trees' predictions. Algorithm 1 makes clear the consequences of the XGBoost ensemble learning method for image segmentation.

$$Pixel\ Intensity = I(i,j) \quad (1)$$

$$Sobel\ gradient\ G(i,j) = \sqrt{(\frac{\partial I}{\partial x})^2 + (\frac{\partial I}{\partial y})^2} \quad (2)$$

Where, $\frac{\partial \mathrm{I}}{\partial \mathrm{x}}$ and $\frac{\partial \mathrm{I}}{\partial \mathrm{y}}$ are horizontal and vertical Sobel filters.

$$LBP_{P,R}(i,j) = \sum_{i=0}^{P-1} s(g_c - g_i).\, 2^p \quad (3)$$

P is the number of neighboring pixels, R is the radius from the center pixel to P^{th} neighboring pixel, step function $s(x)$, g_c is the intensity of the center pixel, $g_i$is intensity of the neighboring pixel, 2^p is weight assign to the Pth bit.

$$Normalized\ row\ r_{norm} = \frac{i}{H}, normalized\ column\ C_{norm} = \frac{i}{W} \quad (4)$$

Where H denotes the height and W denotes the width of the image.

$$Feature\ matrix\ X = \begin{bmatrix} I(0,0) & G(0,0) & LBP(0,0) & r_{norm}(0,0) & C_{norm}(0,0) \\ I(0,1) & G(0,1) & LBP(0,1) & r_{norm}(0,1) & r_{norm}(0,1) \\ . & . & . & . & . \end{bmatrix} \quad (5)$$

Algorithm 1. Segmentation algorithm for XGBoost

Step 1: model initialization with prediction constant (base constant) $F_0(x) = \log(\frac{P}{1-p})$, where p is positive class in training data.

Step 2: error = actual-prediction
Log loss function as in (6)

$$\mathcal{L}(y_i, \hat{y}_i) = -[y_i \log(p_i) + (1 - y_i) \log(1 - p_i)] \quad (6)$$

Where $\hat{y}_i$ is current prediction, y_i is actual prediction, p_i is probability of current prediction $p_i = \frac{1}{1+e^{-\hat{y}_i}}$
Find 1st derivative i.e., gradient $g_i = \frac{\partial \mathcal{L}}{\partial \hat{y}_i} = p_i - y_i$
Find second derivative i.e., hessian $h_i = \frac{\partial^2 \mathcal{L}}{\partial \hat{y}_i^2} = p_i(1 - p_i)$

Step 3: train a regression tree by fitting the computed gradient and hessians, trees create split that reduces the gradient error.

Step 4: prune the tree for max depth = 6 to avoid overfitting.

Step 5: updating of prediction = old prediction + η. Trees output (η = 0.1 is learning rate), as in (7).

$$F_t(x) = F_{t-1}(x) + 0.1\ h_t(x) \quad (7)$$

Step 6: repeat step 2 to 4 for 150 decision trees.

Step 7: final prediction passes through the sigmoid function, as in (8).

$$y_i = \sigma\big(F_t(x)\big) = \frac{1}{1+e^{-F_t(x)}} \quad (8)$$

2.2. Bagging

Bagging is an ensemble machine learning method used to improve model accuracy by reducing the variance. Bootstrap samples are created by sampling with replacement; these samples are used to train the different base learners independently [17]. Predictions from the different learners are subjected to maximum voting for classification to obtain the final prediction from the model.

In this study use Algorithm 2 to illustrate the Bagging ensemble method used for segmentation. Each decision tree is exposed to the different data samples. For each split it samples different features; hence, these trees are diverse. As shown in Figure 1, the final prediction is obtained by maximum voting among the different predictions from 200 diverse decision trees.

Algorithm 2. Segmentation algorithm for Bagging

Step 1: D represents the provided training data; produce the N dataset by bootstrap sampling of the original dataset, as in (9) and (10).

$$D = \{(x_i, y_i)\}_{i=1}^{200} \; original\ data\ set\ used\ to\ generate\ 200\ data\ sets \tag{9}$$

$$N = D_1, D_2, D_3, D_4, \ldots D_{200} \; by\ bootstrap\ sampling\ method \tag{10}$$

Step 2: Build decision tree with max depth = 20.

- Select a feature $m \ll N$
- Split criteria are to maximize the information gain by considering Gini impurity and entropy
- For binary classification $Gini = 1 - (p_0^2 + p_1^2)$, where p_o is sample belongs to class 0 and p_1 is sample belongs to class 1
- Entropy $(H) = -p_0 \log(p_0) - p_1 \log(p_1)$
- Calculate the information gain, as in (11).

$$IG = H(P) + \left[\frac{|L|}{|P|} H(L) + \frac{|R|}{|P|} H(R)\right] \tag{11}$$

Where $|P|$ is number of samples in the parent node, $|L|$ is number of samples in the left node, $|R|$ is number of samples in the right node, and $H(.)$ is impurity measure (consider either entropy or Gini index).

Step 3: each decision tree learns from bootstrap sample.

- $h_n(t)$ is the nth decision tree prediction trained by D_n data samples.
- $h_m(t)$ is the mth decision tree prediction trained by D_m data sample.
- By reducing the correlation between $h_n(t)\ and\ h_m(t)$ reduces the variance.

Step 4: majority voting will be the final prediction

- Final prediction is given by (12) and (13).

$$\hat{y} = \sum_{t=1}^{T} 1.(h_t(x) = c) \tag{12}$$

Where $\hat{y}$ is final prediction, C is number of classes (0,1 for binary classifier), T is 200 (total number of trees), $h_t(x)$ is t^{th} tree prediction, $1.(h_t(x) = c)$ is indicator function.

$$Var(\frac{1}{T}\sum_{t=1}^{T} h_t(x)) = \rho\sigma^2 + \frac{1-\rho}{T}\sigma^2 \tag{13}$$

Where ρ is correlation between trees, if ρ is high then limits the variance reduction, and if ρ is low then variance decreases as T increases.

Step 5: results the class probability.

2.3. Stacking

The stacking ensemble method consists of a meta-learner that provides the final prediction by considering the predictions of all the base learners [18]. Diverse base learners are trained on the same data samples. All the predictions from diverse base learners are fed to the meta-learner to obtain the final prediction. Figure 1 illustrates the stacking used for lung segmentation.

The proposed stacking model combines the Bagging and boosting ensemble methods in the first stage. A hundred random forest (RF) trees are ensembled with a maximum depth of 10 to reduce overfitting, and a hundred gradient boosting trees with a maximum depth of 6 are united in the first stage to obtain diverse predictions. The second stage consists of a meta-learner, preferably a logistic regression model, for better interpretation of the weighted contribution from the base learners in the first stage. Stacking increases the generalizability of the prediction model.

Algorithm 3 is employed as the underlying procedure for constructing the proposed stacking ensemble model. The overall architecture is designed with two heterogeneous base learning algorithms, namely RF and XGBoost, which function as the first-level learners and independently generate prediction outputs from the input data. These prediction outputs are then collected and integrated as new features for the second learning stage. Subsequently, a logistic regression model is utilized as the meta-learner to combine and learn from the outputs produced by the base learners. Through this process, the meta-learner is able to capture complementary information from both RF and XGBoost, thereby generating the final prediction of the stacking ensemble model.

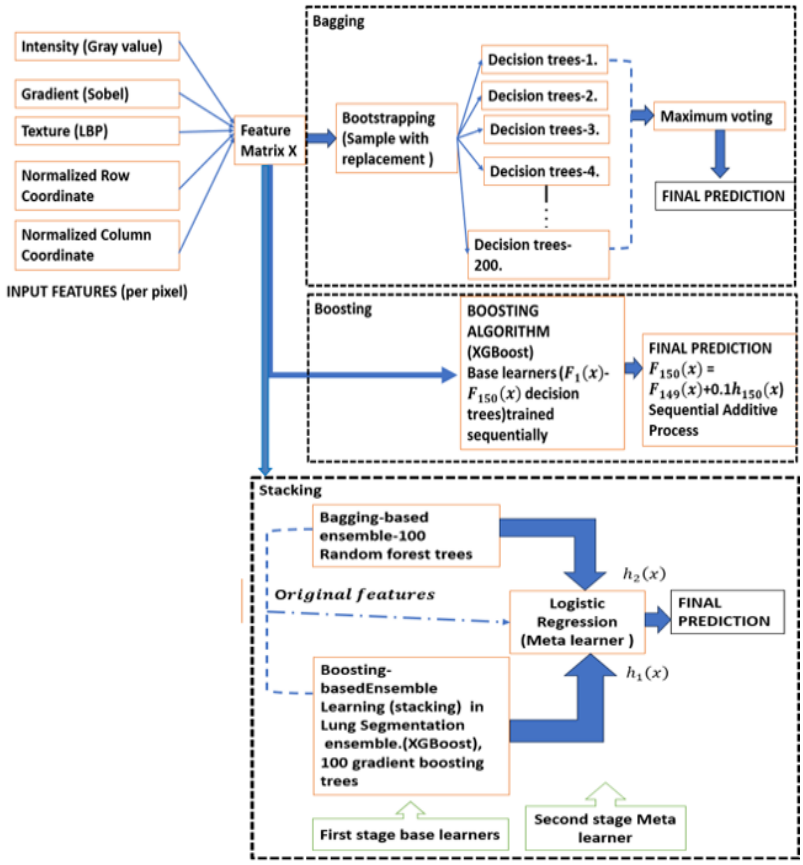

Figure 1. Ensemble methods for image segmentation

Algorithm 3. Segmentation algorithm for stacking
Step 1: initialize the base learners (RF and XGBoost)
– For XGBoosting $D = \{(x_i, y_i)\}_{i=1}^{100}$ original data set used to generate 100 data sets.
– $N = D_1, D_2, D_3, D_4, \ldots, D_{100}$ by bootstrap sampling method
– Learning rate = 0.1, Log loss is the evaluation metric loss, Max depth = 6
– $h_1(x)$ is the prediction form XGBoost
– For RF classifier number of trees = 200, Max depth = 10, random_state = 42
– $h_2(x)$ is the prediction form random forest classifier
Step 2: defining meta learner-logistic regression
Step 3: configure the stacking classifier
– Define pre and final estimators
– Perform 5-fold cross validation for base learners
– Z^i is the input to the meta learner $Z^i = [h_1(x).h_2(x), x^i]$, x^i is input vectors
Step 4: train the meta learner
Step 5: final prediction from the meta learner for lung region is (14)

$$\hat{y} = \sigma(W^T x + b) \quad (14)$$

Where b is bias term, x is the features, W^T is transpose of weight vector, $\sigma(t) = \frac{1}{1+e^{-t}}$ is sigmoid function.

3. RESULTS AND DISCUSSION

3.1. Data set and working platform

In this study utilized COVID-19 CT scan lesion segmentation, a publicly available dataset on Kaggle. This dataset consists of 2,729 CT images. Lung portions are extracted manually by using segment

anything from the Meta AI tool and then subjected to thresholding and morphological operations to obtain the ground truth. The simulation was carried out in Google Collaboratory, a cloud-based computational Jupyter notebook platform. Python 3.11.13 version running on Ubuntu 22.04.4 LTS, including preinstalled libraries OpenCV, NumPy, Matplotlib, scikit-image, and scikit-learn.

3.2. Image segmentation using Bagging, boosting, and stacking

In the proposed work, a machine learning approach is used for pixel-based image segmentation to segment the lung CT image. Each pixel in the given CT image is classified either as a lung region or a background. RF is a base learner used for pixel classification. Figure 2 illustrates the flowcharts of the CT lung segmentation methods using Bagging, boosting, and stacking. Figure 2(a) describes the flowchart of the CT lung segmentation using an ensemble of RF trees. The CT image and its corresponding mask are loaded to the system; the mask is converted to binary to ensure proper classification by the proposed machine learning model. The original CT image underwent resizing to the 256 × 256-pixel format and was then subjected to multimodal feature extraction. The spatial context of each pixel is used to understand the spatial constraints of anatomical structure. Lung CT images consist of imbalanced lung and non-lung regions; this class imbalance makes the classifier more biased towards the majority classes. This problem is overcome by downsampling the non-lung regions. Decision trees of 200 with a maximum depth of 20 are assembled to obtain the strong classifier, which is capable of handling the mixed image features in a noniterative method. The morphological post-processing operation removes the artifacts by filling holes smaller than 100 pixels.

XGBoost is used to build the trees sequentially by correcting the residual errors of the previous trees. It can fit the model on a balanced feature set and predict the label for the new pixels. Each pixel consisting of feature vectors is fed to the 150 ensemble decision trees to classify that pixel as either lung or background pixels and then aggregate the decisions of all the trees to produce the final prediction. A morphological post-processing operation was performed to fine-tune the results. XGBoost builds the optimized decision trees for classification. Figure 2(b) represents the flow of CT image segmentation using XGBoost. The first decision tree is a weak learner; 150 cascading decision trees are added iteratively to build the strong classifier. The final prediction model classifies the pixels into lung and non-lung regions by assembling the results of all weak decision trees.

The stacking classifier combines the multiple base learners (200 decision trees and 100 boosted trees) and the high-level meta learner. As shown in Figure 2(c), stacked data is generated by fivefold cross-validation (CV) performed on base learners and the original image features. A meta-learner was trained on the stacked dataset to generate optimal weights that produce the final prediction for lung segmentation. Stacking reduces the variance and bias and prevents the system from overfitting. A logistic regression model is the ideal meta-learner since it can detect features that the base learner missed. The model can be modified to include new learners for further experimental analysis. Because their decision boundaries were designed to favor the majority classes, the base learners were biased towards the majority non-lung pixel classes. The well-balanced classes enable the foundational learners to acquire the meaning of both lung and non-lung regions. Without prejudice, out-of-fold validation and fivefold CV prevent the meta-learner from training on the dominant class pixel features by using pixels that represent both classes. The meta-learner's performance improved, and a diverse decision boundary developed by the base learner with balanced learning [19], [20]. Post-processing operations remove the suspicious tiny objects. The Sobel operator is sensitive to any intensity variations, and LBP can locate the small texture irregularities that make the edge handcrafted. The fine-scale smoothing of the boundaries can be improved by using higher-order morphological filters after segmentation.

3.3. Performance evaluation

Performance of the ensemble methods was evaluated by using the ground truth and the predicted mask. The segmentation performance was evaluated using accuracy, specificity, Dice coefficient, precision, recall, F1-score, IoU, Hausdorff distance, and lung coverage [21]. Simulation results of Bagging, boosting, and stacking ensemble methods for lung segmentation in Table 1 reveal that all three methods achieved satisfactory results.

Boosting outperformed in all the evaluation metrics compared with Bagging and stacking. A precision of 97.96% indicates it can effectively reduce the false positives. Recall of 100% indicates that all the lung pixels are segmented; hence, the model is highly sensitive. An F1-score of 98.97% represents that the model is highly reliable (minimal over-segmentation and zero under-segmentation). The highest score of the intersection of the union, 97.96%, indicates accurate intersection of ground truth and the predicted mask. Lung coverage of 100% makes this method an option for an effective approach for the lung segmentation task.

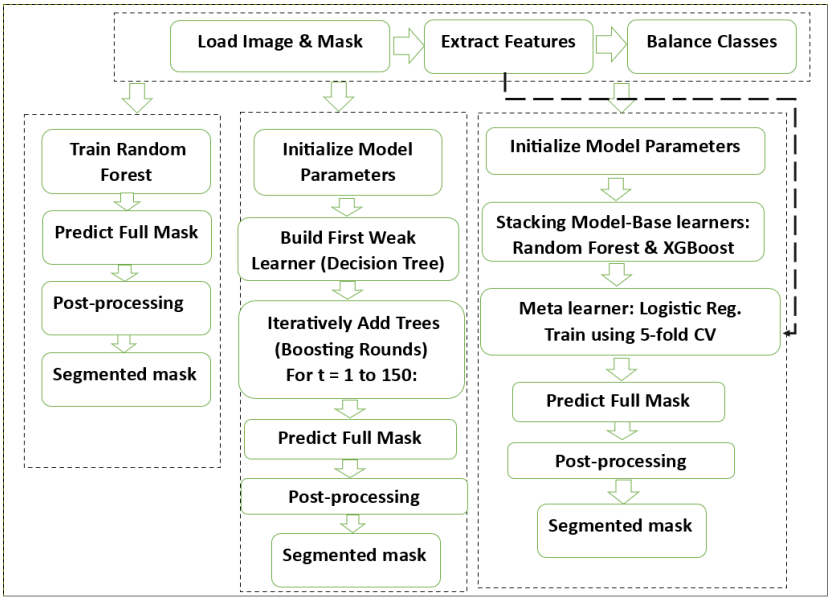

Figure 2. Flow chart of the CT lung segmentation using (a) Bagging, (b) boosting, and (c) stacking

Table 1. Performance metric comparative table

Metric	Bagging	Boosting	Stacking
Accuracy	0.9958	0.9964	0.9857
Specificity	0.9949	0.9956	0.9825
Dice coefficient	0.9880	0.9897	0.9600
Hausdorff dist	2.8284	0.2235	4.4721
Precision	0.9763	0.9796	0.9240
Recall	1.0000	1.0000	0.9999
F1-score	0.9880	0.9897	0.9604
IoU	0.9763	0.9796	0.9239
Lung coverage	100.00%	100.00%	99.99%

3.4. Comparison with state of art method

The consistent improvement across all the performance metrics considered for the evaluation metric was achieved by the boosting ensemble method among the tested approaches in lung parenchyma segmentation; hence, this method is compared to the prevailing lung parenchyma segmentation methods. As indicated in Table 2, the proposed method significantly exceeded the present method with dice values of 0.9897, 100% recall, 0.9956 specificity, and 0.9796 IoU. The visual assessment in the further section supports these findings, showing more precise lung parenchyma boundary delineation and better shape preservation.

Table 2. Comparison with existing method

Method	Dice	Sensitivity	Accuracy	Recall	Specificity	Precision	F1-score	IoU
LDANet [3]	0.9843	0.9829	0.9952	----	----	----	----	----
U-Net fused with shape stream [4]	0.9329	----	0.9885	----	----	----	0.9805	0.8658
Recurrent residual UNet [5]	0.9795	----	----	----	0.9893	----	----	0.9581
Mask R-CNN [7]	0.934	0.8956	0.9879	----	0.9954	0.9110	----	----
3D attention ResUNet [22]	0.9361	----	----	----	----	----	----	0.9103
FusionLungNet [23]	----	----	0.9979	0.9941	----	0.9865	0.9901	0.9812
Proposed method (boosting)	0.9897	----	0.9964	1.0000	0.9956	0.9796	0.9897	0.9796

3.5. Qualitative analysis of proposed method

Simulation results of all the ensemble methods are illustrated in Figure 3. Figure 3(a) shows the original CT image, while the foreground represents the lung region and the background represents the non-lung parenchyma regions in Figure 3(b). The Bagging and boosting ensemble methods in Figures 3(c)

and 3(d), respectively, resemble the ground truth. Stacking in Figure 3(e) results in hard segmentation with rough edges. The visual comparison confirms that the XGBoost ensemble method achieves smoother contours than the other ensemble methods.

The proposed ensemble (XGBoosting) method is compared with pretrained deep learning models, as shown in Figure 4. Figure 4(a) presents the input CT image, while the ground truth in Figure 4(b) is manually annotated using the segment anything annotation tool and post-processing operations. R231CovidWeb (UNet) [24] in Figure 4(c) and DeepLabV3 with ResNet50 [25] in Figure 4(d) were used to compare the qualitative performance of the proposed model in Figure 4(e). The results demonstrate that the proposed method can segment different pathological structures in CT images and achieve competitive performance compared with the deep learning architectures.



Figure 3. Comparative visualization of CT image segmentation using ensemble methods: (a) original image, (b) ground truth, (c) Bagging, (d) boosting, and (e) stacking

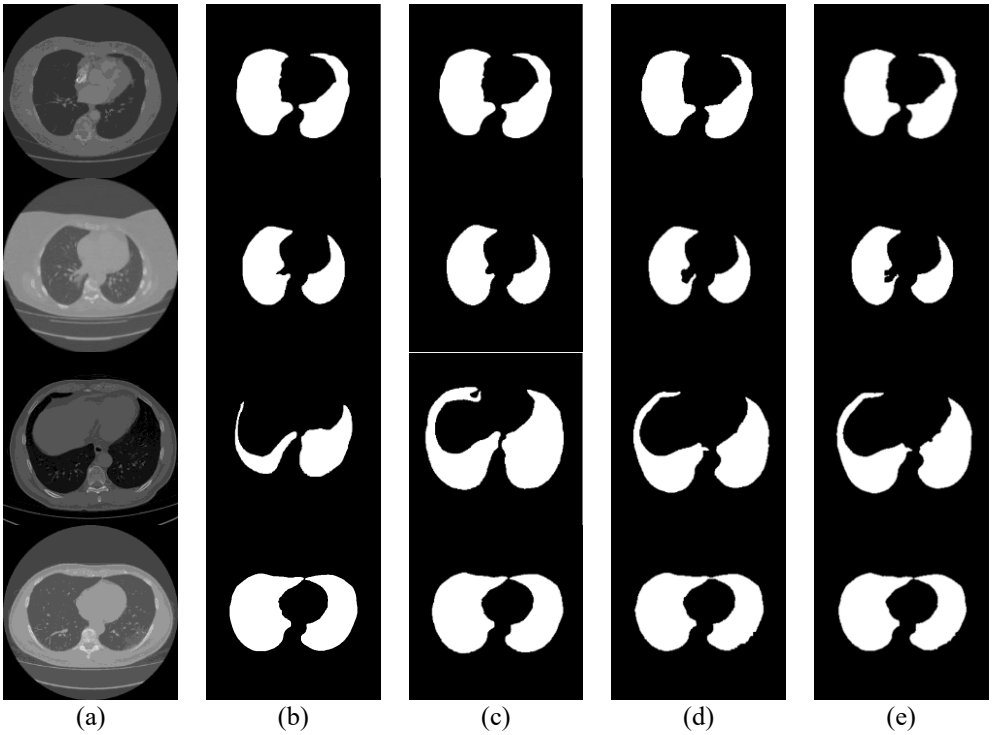

Figure 4. Visual comparison of proposed method with deep learning methods: (a) input image, (b) ground truth, (c) UNet, (d) Deeplabv3-ResNet50, and (e) proposed method

4. CONCLUSION

The comparative study of Bagging, boosting, and stacking ensemble methods for CT image segmentation was performed. Boosting (XGBoost) emerges as the most effective and reliable ensemble method for segmenting lung CT images by achieving remarkable accuracy, precision, recall, F1 score, IoU, and lung coverage. Bagging is most closely related to boosting in performance, but stacking lags relatively in performance. Findings in this work highlight that boosting is a highly reliable and robust ensemble method for CT image segmentation. The performance of the boosting ensemble method on CT images with different

pathologies, like asymmetric lungs and bound and top slices, will be studied and compared with the deep learning architectures. The proposed method surpasses the deep learning segmentation models.

ACKNOWLEDGMENTS

This work is carried out as a part of my Ph.D. work at the Department of Electronics and Communication Engineering, Sri Siddhartha Academy of Higher Education, Tumkur, India.

FUNDING INFORMATION

This research received no specific grant from any funding agencies.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Asharani Ramadas	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓			
Chidananda Murthy Melekote Vinayakamurthy	✓	✓			✓	✓			✓	✓		✓	✓	

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

The authors state they have no rival interests.

INFORMED CONSENT

The data set used in this work is a publicly available database consisting of anonymized medical images provided for research purpose, hence No consent is required for the secondary analysis.

ETHICAL APPROVAL

The dataset utilized in this study is publicly accessible and created for research purposes; hence, no ethical approval was taken.

DATA AVAILABILITY

Data availability does not apply to this paper because this study did not create any new data.

REFERENCES

[1] H. Sung *et al.*, "Global cancer statistics 2020: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries," *CA: A Cancer Journal for Clinicians*, vol. 71, no. 3, pp. 209–249, May 2021, doi: 10.3322/caac.21660.
[2] R. E. Ledda, G.-C. Funk, and N. Sverzellati, "The pros and cons of lung cancer screening," *European Radiology*, vol. 35, no. 1, pp. 267–275, Jul. 2024, doi: 10.1007/s00330-024-10939-6.
[3] Y. Chen *et al.*, "LDANet: automatic lung parenchyma segmentation from CT images," *Computers in Biology and Medicine*, vol. 155, Mar. 2023, doi: 10.1016/j.compbiomed.2023.106659.
[4] L. Zhu, Y. Cai, J. Liao, and F. Wu, "Lung parenchyma segmentation based on U-Net fused with shape stream," *IEEE Access*, vol. 12, pp. 29238–29251, 2024, doi: 10.1109/access.2024.3365577.
[5] G. S. Nandeesh, M. Nagabushanam, S. Pramodkumar, and S. Nandini, "Lung parenchyma segmentation and nodule detection using deep learning," *Journal of Optics*, vol. 53, no. 1, pp. 635–642, Feb. 2024, doi: 10.1007/s12596-023-01187-w.
[6] H. Wu, Z. Zhang, Y. Zhang, B. Sun, and X. Zhang, "ACX-UNet: a multi-scale lung parenchyma segmentation study with improved fusion of skip connection and circular cross-features extraction," *Signal, Image and Video Processing*, vol. 18, no. 1, pp. 525–533, Feb. 2024, doi: 10.1007/s11760-023-02770-1.

[7] W. A. P. Llacho, E. C.-Gutierrez, and L. D. H. Tapia, "Lung parenchyma segmentation using mask R-CNN in COVID-19 chest CT scans," *International Journal of Advanced Computer Science and Applications*, vol. 16, no. 2, 2025, doi: 10.14569/ijacsa.2025.0160214.
[8] S. R. Quasar, R. Sharma, A. Mittal, M. Sharma, D. Agarwal, and I. de La T. Díez, "Ensemble methods for computed tomography scan images to improve lung cancer detection and classification," *Multimedia Tools and Applications*, vol. 83, no. 17, pp. 52867–52897, Nov. 2023, doi: 10.1007/s11042-023-17616-8.
[9] T. G. Dietterich, "Ensemble methods in machine learning," in *Multiple Classifier Systems*, Berlin, Heidelberg: Springer, 2000, pp. 1–15, doi: 10.1007/3-540-45014-9_1.
[10] S. Ardabili, A. Mosavi, and A. R. V.-Kóczy, "Advances in machine learning modeling reviewing hybrid and ensemble methods," in *Engineering for Sustainable Future (INTER-ACADEMIA 2019)*, Cham, Switzerland: Springer, 2020, pp. 215–227, doi: 10.1007/978-3-030-36841-8_21.
[11] T. N. Rincy and R. Gupta, "Ensemble learning techniques and its efficiency in machine learning: a survey," in *2nd International Conference on Data, Engineering and Applications (IDEA)*, Feb. 2020, pp. 1–6, doi: 10.1109/IDEA49133.2020.9170675.
[12] T. Dang, T. T. Nguyen, J. McCall, E. Elyan, and C. F. M.-García, "Two-layer ensemble of deep learning models for medical image segmentation," *Cognitive Computation*, vol. 16, no. 3, pp. 1141–1160, May 2024, doi: 10.1007/s12559-024-10257-5.
[13] S. Göb *et al.*, "Comparing ensemble methods combined with different aggregating models using micrograph cell segmentation as an initial application example," *Journal of Pathology Informatics*, vol. 14, 2023, doi: 10.1016/j.jpi.2023.100304.
[14] Z. Faska, L. Khrissi, K. Haddouch, and N. El Akkad, "A robust and consistent stack generalized ensemble-learning framework for image segmentation," *Journal of Engineering and Applied Science*, vol. 70, no. 1, Dec. 2023, doi: 10.1186/s44147-023-00226-4.
[15] R. Sibindi, R. W. Mwangi, and A. G. Waititu, "A boosting ensemble learning based hybrid light gradient boosting machine and extreme gradient boosting model for predicting house prices," *Engineering Reports*, vol. 5, no. 4, Apr. 2023, doi: 10.1002/eng2.12599.
[16] S. M. Ganie, P. K. D. Pramanik, M. B. Malik, A. Nayyar, and K. S. Kwak, "An improved ensemble learning approach for heart disease prediction using boosting algorithms," *Computer Systems Science and Engineering*, vol. 46, no. 3, pp. 3993–4006, 2023, doi: 10.32604/csse.2023.035244.
[17] L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24, no. 2, pp. 123–140, Aug. 1996, doi: 10.1007/BF00058655.
[18] R. Dey and R. Mathur, "Ensemble learning method using stacking with base learner, a comparison," *Proceedings of International Conference on Data Analytics and Insights*, 2023, pp. 159–169, doi: 10.1007/978-981-99-3878-0_14.
[19] T. Zhou and H. Jiao, "Exploration of the stacking ensemble machine learning algorithm for cheating detection in large-scale assessment," *Educational and Psychological Measurement*, vol. 83, no. 4, pp. 831–854, Aug. 2023, doi: 10.1177/00131644221117193.
[20] A. A. Khan, O. Chaudhari, and R. Chandra, "A review of ensemble learning and data augmentation models for class imbalanced problems: combination, implementation and evaluation," *Expert Systems with Applications*, vol. 244, Jun. 2024, doi: 10.1016/j.eswa.2023.122778.
[21] D. Müller, I. S.-Rey, and F. Kramer, "Towards a guideline for evaluation metrics in medical image segmentation," *BMC Research Notes*, vol. 15, no. 1, Dec. 2022, doi: 10.1186/s13104-022-06096-y.
[22] M. Abdullah and F. Shaukat, "Attention-ResUNet and EfficientSASM-UNet: UNet based frameworks for lung and nodule segmentation," 2025. *arXiv: 2505.17602*.
[23] S. Rezvani, M. Fateh, Y. Jalali, and A. Fateh, "FusionLungNet: multi-scale fusion convolution with refinement network for lung CT image segmentation," *Biomedical Signal Processing and Control*, vol. 107, Sep. 2025, doi: 10.1016/j.bspc.2025.107858.
[24] N. Enshaei *et al.*, "COVID-rate: an automated framework for segmentation of COVID-19 lesions from chest CT images," *Scientific Reports*, vol. 12, no. 1, Feb. 2022, doi: 10.1038/s41598-022-06854-9.
[25] M. Murugappan, A. K. Bourisly, N. B. Prakash, M. G. Sumithra, and U. R. Acharya, "Automated semantic lung segmentation in chest CT images using deep neural network," *Neural Computing and Applications*, vol. 35, no. 21, pp. 15343–15364, Jul. 2023, doi: 10.1007/s00521-023-08407-1.

BIOGRAPHIES OF AUTHORS



Asharani Ramadas is a research scholar at Sri Siddhartha Academy of Higher Education, Tumkur, Karnataka, India. She graduated in telecommunication engineering and a Master of Technology in Digital Electronics and Communication Engineering from Vishweshvariah Technological University, Karnataka, India. She is an assistant professor at Ramaiah College of Arts, Science and Commerce, Bengaluru, India. Her research includes medical image processing, artificial intelligence, and machine learning. She can be contacted at email: asha.srp@gmail.com.



Chidananda Murthy Melekote Vinayakamurthy is an associate professor at Sri Siddhartha Academy of Higher Education in the Department of Electronics and Communication Engineering. He received his B.E., M.Tech., and Ph.D. from Visvesvaraya Technological University in 2005, 2007, and 2019, respectively. In addition to giving presentations at conferences, he has 26 papers published in journals. His areas of interest in study are computer vision, machine learning, and high-resolution images. He can be contacted at email: chidanandamurthymv@ssit.edu.in.