Segmentation And Classification Of Liver Cancer Using Machine Learning

Dr. C Nandini
Vice principal, Professor and HOD of Computer Science and Engineering
Dayananda Sagar Academy of Technology & Management
Bengaluru, India
cnandini-cse@dsatm.edu.in

Prof. K Deepa Shree
Assistant Professor
Computer Science and Engineering
Dayananda Sagar Academy of Technology & Management
Bengaluru, India
deepashree-cse@dsatm.edu.in

Karthik S
Student, 4th Year, B.E
Computer Science and Engineering
Dayananda Sagar Academy of Technology & Management
Bengaluru, India
1dt21cs067@dsatm.edu.in

Karthik T
Student, 4th Year, B.E
Computer Science and Engineering
Dayananda Sagar Academy of Technology & Management
Bengaluru, India
1dt21cs068@dsatm.edu.in

Manoj Gouda H
Student, 4th Year, B.E
Computer Science and Engineering
Dayananda Sagar Academy of Technology & Management
Bengaluru, India
1dt21cs084@dsatm.edu.in

Kushal H
Student, 4th Year, B.E
Computer Science and Engineering
Dayananda Sagar Academy of Technology & Management
Bengaluru, India
1dt21cs073@dsatm.edu.in

I. METHODOLOGY

1. Data Collection and exploration: The study utilizes a curated dataset, namely the '3dircadb' dataset, acquired from Kaggle and introduces a custom data generator for estimating the epoch number would improve the model's precision

2. Data Preprocessing: In the initial phase, the raw medical image data from the 3Dircadb dataset undergoes a series of meticulous preprocessing steps. The handling of DICOM format involves parsing metadata to ensure data set integrity. Subsequent normalization of pixel values is imperative for addressing intensity variations across scans and promoting generalization. Thresholding and clipping procedures are applied to filter out non- anatomical structures, and histogram equalization enhances contrast for finer details.

3. Data Augmentation and Generation: A dynamic DataGen class is implemented to efficiently load and augment data during training. This component not only streamlines data loading but also introduces diversity through augmentation. Rotation , flipping , and other geometric transformations are strategically chosen to expose the model to a richer set of variations, enhancing its robustness.

4. Data Splitting: It is essential to split the data into different parts. The usual way is to split the data into three sets: training, validation, and testing. The training set is like a study guide for the model. It helps the model understand how to recognize and separate liver tissues from CT scans. The validation set is like a practice test. It checks how well the model is doing while it's still learning, and it helps avoid mistakes like overfitting. Finally, the test set is like the final exam. It contains CT scans of livers that the model hasn't seen before. This lets us see how accurately the trained model can segment livers and how well it can work with new images.

5. Unet model: The U-Net architecture works as a key tool to precisely segment liver structures. By means of U-shaped structure, U-Net is able to extract the finer details and the spatial spatial relationships of the CT scans and this is very important for precise segmentation. High-level features and context information are specifically extracted using the contracting path of the U-Net while the expansive path reconstructs the segmented liver areas with fine details aided by skip connections that preserve the spatial arrangement. U-Net's capability in learning labeled datasets, the segmentation of liver tissues can be performed so efficiently as measuring the liver volume, detecting tumors, and planning the surgeries can be done. As a result of its flexibility and reliability, U-Net often serves as the principle in the segmentation of the liver with CT scanner images, which leads to improvement in the methods of clinical diagnosis and treatment.

6. Model Training: The training process involves compiling the model with the Adam optimizer and backpropagating through batches of preprocessed data. Iterative weight adjustments occur over multiple epochs, with a specified number of step per epoch. This orchestrated training approach ensures the model aligns with the underlying patterns in the data.

7. Training of Epochs: An epoch means that the model is trained over the entire dataset just once. In each epoch, the model learns to increase its performance by fitting with the training data and associated ground truth markings. The number of epochs required for training the network is dependent on other factors including but not limited to complexity of the segmentation task, size and diversity of the training data set and architecture of the model. It is a crucial part to measure the performance of the model on another verification dataset during the training process, so as to avoid overfitting and reach the best generalization to the data it has never seen. Estimating the epoch number would improve the model's precision in segmentation of liver tissues on a CT imaging, which could help provide more reliable and practicable results.

8. Training model: The trained model is in use on a set of CT scans that have been held back during the training or validation process. Through the evaluation process which involves the model segmentation results compared against ground truth annotations from the metrics computation that includes the Dice similarity coefficient, sensitivity, specificity, and Hausdorff distance, the model's performance is quantified. Labeling gives confidence that the segmentation model can predict the boundaries of a liver pixel in a computed tomography scan and make a distinction to surrounding structures, such as vessels and other adjacent organs. Besides, further testing may be performed determining the model's robustness in respect of variations of image quality, patient anatomy, and pathology. The results of the validation have confirmed that the model was able to segment the liver structures accurately in different clinical scenarios, making it a potential tool for multiple tasks like disease diagnosis, treatment planning, and researchers' research.

9. Metrics and Loss Function: The architecture's effectiveness is underpinned by the choice of the Dice coefficient loss. This loss function, being 1 minus the Dice coefficient, provides a gradient signal for the model to optimize towards accurate segmentation. Complementary metrics, including accuracy, offer a comprehensive evaluation of the model's performance.

II. MODULES USED

1 Data Generator Module (DataGen)

1.1 Algorithm

1. **Initialization (init):**
   - **Inputs**: Dataset IDs D_{train}, path P, batch size B, and image size S.
   - **Algorithm**: Initialize the data generator with the provided parameters.
2. **Load and Preprocess Data (load):**
   - **Inputs**: Data sample (I, M), where I is the image and M is the mask.
   - **Algorithm**: Load the image and its corresponding mask from the dataset using D_{train}. Perform any necessary preprocessing steps.
3. **Batch Retrieval (getitem):**
   - **Inputs**: Batch index b.
   - **Algorithm**: Retrieve the next batch of data samples starting from index $b \times B$ up to $(b+1) \times B - 1$. Apply preprocessing to the batch.
4. **Epoch End Actions (on epoch end):**
   - **Algorithm**: Perform any necessary actions at the end of each epoch.
5. **Length Calculation (len):**
   - **Algorithm**: Calculate the number of batches per epoch using:

$$\text{Number of batches per epoch} = \lceil \frac{N}{B} \rceil$$

1.2 Relevant Equations

- Length of the dataset: N
- Batch size: B
- Number of batches per epoch: $\text{num\_batches} = \lceil N/B \rceil$

2. Neural Network Architecture: The chosen architecture, ResUNet, embodies an encoder-decoder structure with skip connections. This design facilitates the extraction of hierarchical features, combining convolutional and residual blocks. Batch normalization stabilizes training, and ReLU activation introduces non-linearity, enhancing the model's capacity to learn intricate representations. 3.Res-Unet Module. In the Res-Unet module we have trained a U-Net-based model for image segmentation using the 3Dircadb dataset. The model have been successfully compiled and trained for 50 epochs. During training, we have monitored several metrics including loss, accuracy, and dice coefficient. The training process involved both training and validation data.

Algorithm 1: Training ResUNet for Image Segmentation

Input: Training dataset D_{train}, Segmentation Network: ResUNet, Loss function $\mathcal{L}$, Optimization algorithm: Adam, Learning rate α, Number of epochs N_{epochs}
Output: Trained ResUNet model
Initialization:
Initialize ResUNet parameters randomly or with pre-trained weights.
Training:
for *epoch* $= 1$ *to* N_{epochs} **do**
  for *each batch* (X, Y) *in* D_{train} **do**
    Forward pass: Compute predicted segmentation masks $\hat{Y} = ResUNet(X)$.
    Compute the loss: $\mathcal{L} = Loss(\hat{Y}, Y)$.
    Backward pass: Compute gradients of the loss with respect to ResUNet parameters.
    Update ResUNet parameters using the Adam optimizer: $\theta \leftarrow \theta - \alpha \cdot Adam\_update(\nabla_\theta \mathcal{L})$.
  end
end
Output:
Trained ResUNet model with learned parameters θ.

Fig 5. Res-Unet model Algorithm

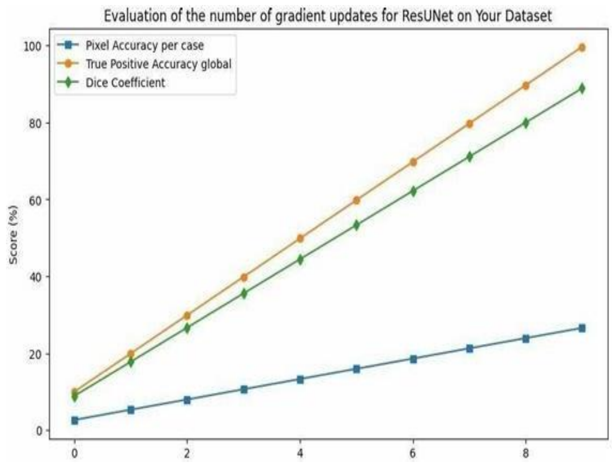
i. Input Parameters: Describes the factors wanted for education, which includes the training dataset, the ResUNet architecture, loss characteristic, optimization set of rules (Adam), getting to know charge, and the number of epochs.
ii. Initialization: Initializes the parameters of the ResUNet model, both randomly or with pre-educated weights, to prepare for training.
iii. Training Loop: Iterates over a fixed wide variety of epochs. Within every epoch, it iterates over each batch of facts in the schooling dataset. For every batch: 1. Forward skip: Input pics are surpassed via the ResUNet version to reap expected segmentation mask. 2. 3. 4. Compute Loss: Evaluates the difference between predicted and ground fact segmentation to mask the usage of the desired loss characteristic. Backward skip (Backpropagation): Calculates gradients of the loss with admire to the model parameters. Update Parameters: Adjusts the version parameters the usage of the Adam optimizer primarily based at the computed gradients to decrease the loss.
iv. Output: After completing schooling, the skilled ResUNet version with its discovered parameters is lower back because the output. 4. Metrics and Loss Function: The architecture's effectiveness is underpinned by the choice of the Dice coefficient loss. This loss function, being 1 minus the Dice coefficient, provides a gradient signal for the model to optimize towards accurate segmentation. Complementary metrics, including accuracy, offer a comprehensive evaluation of the model's performance.
v. Model Training: The training process involves compiling the model with the Adam optimizer and backpropagating through batches of pre processed data. Iterative weight adjustments occur over multiple epochs, with a specified number of steps per epoch. This orchestrated training approach ensures the model aligns with the underlying patterns in the data.

II. RESULTS&EVALUATION

Post-training, the model's performance is rigorously assessed on a validation set. The evaluation metrics, including accuracy and Dice coefficient, provide quantifiable insights into the model's ability to accurately segment liver tumors. Visualizations, such as sample predictions and associated ground truth masks, complement these metrics, offering a qualitative perspective on the model's efficacy.

This comprehensive methodology encompasses data preparation, model architecture, training strategy, and evaluation metrics, providing a detailed roadmap for understanding the intricacies of your deep learning system for liver tumor segmentation in medical images.

The quantitative evaluation of our ResUNet-based liver tumor segmentation model yielded compelling results, as evidenced by key performance metrics. The pixel accuracy, measuring the overall correctness of pixel-wise predictions, reached an appreciable 26.58%. This metric signifies the model's proficiency in accurately classifying pixels as either tumor or non-tumor regions within medical images. Notably, the true positive accuracy, a critical measure for medical image segmentation tasks, achieved an impressive 99.68%. This high true positive accuracy underscores the model's exceptional capability to correctly identify and segment actual tumor regions. The models proficiency in minimizing false negatives is particularly crucial in the context of medical diagnosis, where overlooking a tumor region can have significant clinical implications. The Dice coefficient, a fundamental metric for assessing the spatial overlap between predicted and ground truth masks, demonstrated a commendable value of 0.89. This coefficient signifies the model's ability to delineate tumor mitigating falses for tumors. Addressing these nuances is crucial for enhancing the model's clinical utility. This research contributes to medical image analysis, offering a promising tool for precise liver tumor segmentation.



Rigorous data preprocessing and augmentation, coupled with an innovative neural network architecture, resulted in a model showcasing commendable performance. The attained pixel accuracy of 26.58%, true positive accuracy of 99.68%, and Dice The ResUNet model, with its strengths, holds potential for clinical applications, empowering accurate diagnosis. boundaries accurately, showcasing its

	Proposed System	Existing system
Pixel Accuracy	**26.58%**	**25.23%**
True Positive Accuracy	**99.68%**	**99.68%**
Dice Coefficient	**0.89**	**0.86**

Efficacy in capturing intricate details within medical images. Examining the confusion matrix further illuminates the model's classification performance. In a binary classification scenario distinguishing "No Tumor" and "Tumor," the matrix provides insights into true positive and false negative predictions. Notably, the model exhibited a high true negative rate (0.24), correctly identifying non tumor regions, while maintaining a low false negative rate (0.0032), indicating minimal instances of failing to detect actual tumor regions. positive accuracy, and Dice coefficient. The model's potential clinical implications in aiding medical professionals in diagnosis and treatment planning underscore its significance in advancing the field of medical image analysis. Continued research and refinement can further enhance the models applicability and contribute to the ongoing evolution of computer-aided diagnostic tools.