**3D Segmentation Model on CT Abdomen Organs**

**Overview**

3D Segmentation Model on CT Abdomen Organs is built and trained to segment the abdominal organs from CT scans. The primary goal is to segment the Liver, Right and Left Kidneys, and Spleen. The dataset used is the FLARE22 dataset, which consists of 3D MRI scans and corresponding ground truth segmentation masks. The project includes data extraction, preprocessing, model building, training, evaluation, and data visualization of results.

**Dataset:** [**MICCAI FLARE22 Challenge Dataset (50 Labeled Abdomen CT Scans) (zenodo.org)**](https://zenodo.org/records/7860267)

Setup Instructions

Python 3.7 or higher

• TensorFlow 2.x

• Nibabel

• NumPy

• Matplotlib

• Scikit-learn

• SciPy

**!pip install tensorflow nibabel numpy matplotlib scikit-learn scipy**

**Model Architecture**

**VNet Architecture**

VNet is a deep learning model designed for volumetric (3D) segmentation. It uses a similar approach to U-Net but is tailored for 3D image data. Below are key architectural details:

**Input Layer:**

Takes a 3D input image with shape (128, 128, 64, 1).

**Encoder Path:**

Convolutional Layers: Several 3D convolutional layers with ReLU activation functions to extract features.

MaxPooling Layers: Reduce the spatial dimensions of the feature maps**.**

**Bottleneck:**

The deepest part of the network with several convolutional layers without downsampling.

**Decoder Path:**

Transpose Convolutional Layers: Upsample the feature maps to the original spatial dimensions.

Concatenation: Concatenate features from the encoder path to preserve spatial details.

**Output Layer:**

A 3D convolutional layer with a softmax activation function to predict the probability of each class for every voxel.

The model is built using TensorFlow/Keras and consists of multiple convolutional and pooling layers arranged in an encoder-decoder fashion. The output of the model is a 4D tensor where each voxel in the input image is classified into one of the predefined organ classes.

**Dice Coefficient**

The Dice coefficient is a metric used to evaluate the performance of the segmentation model. It measures the overlap between the predicted segmentation and the ground truth. The formula is:

**Dice = 2 × Intersection**

**------------------**

**Sum of the sizes of two sets**

Training Process

Data Preprocessing

The preprocessing steps include:

**Loading Data:**

The data is loaded using the nibabel library, which reads .nii.gz files.

**Resizing:** Images and labels are resized to a fixed target dimension (128, 128, 64) using scipy.ndimage.zoom. This step ensures that all input data is of the same size.

**Filtering Labels:** Labels are filtered to include only the target organ classes (liver, right kidney, left kidney, spleen). Non-target classes are excluded or set to zero.

**Normalization:**

Images are normalized to the range [0, 1].

**Training Procedure**

**Data Generators:**

A custom DataGenerator class is used to load and preprocess data on-the-fly during training. This class handles batching, shuffling, and data augmentation.

**Model Compilation:**

The model is compiled with the Adam optimizer, categorical crossentropy loss, and accuracy and Dice coefficient as metrics.

**Model Training:**

The model is trained for 10 epochs. During training, the loss and accuracy are monitored for both training and validation data.

**Validation and Inference**

**Validation Process**

**Validation Data:**

The validation set is used to evaluate the model's performance after training. Predictions are generated for the validation set.

**Dice Coefficient Calculation:**

The Dice coefficient is calculated for each organ class separately. This metric evaluates the model's accuracy in segmenting each organ.

**3D Visualization:**

The model's predictions are compared with the ground truth labels to visualize the segmentation results. Example slices from the validation set are plotted to illustrate the model's performance.

**Video Link**

[**Video - Google Drive**](https://drive.google.com/drive/folders/16D13Trodc9piSM1gDyJ5mgWzWMbgqOPL)