Automatic Detection and Segmentation of Brain Tumor Using Random Forest Approach

Abstract by Leonard Brenk

October 26, 2020

The idea is to enhance and improve the detection and segmentation of brain tumors whereas deciding about its presence is the primary goal of this study. Data used for training and testing decision trees and random forests were obtained from the MICCAI BRATS database. It consists of multiple volumes - meaning three-dimensional images of brains - in different types of images resulting from magnetic resonance imaging: T1, T2, T1 post-Gladinium and FLAIR. These types are called features. The images were generated from 30 glioma patients in different stages. In this study 12 records each containing 4 MRI scans were used. Seeing that four volumes of a record show the same brain differently a way of addressing a single voxel - a 3D-pixel - is needed. Therefore a feature-vector has been designed to point out one specific coordinate in each of the four feature volumes:

$$\vec{x} = \Big[T1\,[x,y,z],\, T2\,[x,y,z],\, T1C\,[x,y,z],\, FLAIR\,[x,y,z] \Big]$$

Thereby we can address one voxel and compare and process it directly. This is required for training and testing binary decision trees. Before the data can be used it has to undergo certain preprocessing steps like the normalization of intensity values. The intensity of a pixel is its brightness. To normalize a histogram, it is changed in order to locate 50% of the data between the intensity values of 600 and 800. Also, values smaller than 200 and larger than 1200 are being replaced by their limit. Another preliminary adaption of the data is the computation of location information. Since the feature vector does not carry any information regarding this, eight more features are included in the feature vector, two for each channel. So now the feature vector contains 12 elements. The first added feature is computed within a 10-element neighborhood which contains 8 elements out of the current slice and the two closest ones from neighboring slices. The second feature is based on all neighbors of the pixel in a 3 x 3 x 3 cube.

For a system to recognize a pattern, binary decision trees (BDT) are being trained and tested. The intension of using such BDTs is to be able to attribute a given data vector to a class after having trained the tree. In this study, the feature vectors serve as the input data where at each node of the tree exactly one feature is being considered. While training the tree a vector that is already classified into tumor, edema, or negative by humans, is fed to the root node. Since 12 features can be used at a node to divide the data into two subgroups, the feature with the highest information gain is being selected. For us to discover which features to pick we use the following equation:

$$\sum_{i=1}^{n} P(x_i) log P(x_i)$$

This equation is used to calculate the information gain: Information without split - Information of two separately computed levels of information after the split. The feature with the highest gain will split the data most efficiently.

Having selected a feature at a new node a threshold is being created which will either forward the vector to the right or to the left child node. At some point, it won't be possible to divide the set of vectors any further, which leads to the creation of a leaf node. As we train supervised we already know the label of a pixel - whether the input pixel is a tumor pixel. Now we can attribute that leaf node to a certain class.

A tree's classification ability is optimized for that specific set of training data. Unknown images will probably result in vast errors. As an approach to solve that problem, Bagging is used to reduce variance. Thereby several subsets are created out of the training data. each of which generated its own decision tree. The result is an ensemble of different models which average leads to a majority based decision on that input data vector. However one tree uses all features, which can be further optimized by random forest. It is also an ensemble method which uses - additional to many randomly chosen trees - randomly chosen subsets of features to train these trees.

Although a random forest can detect most of the tumor pixel in an image we still can achieve a significant improvement through post-processing. Thereby we count how many neighbors of that pixel in a 250-pixel radius are also labeled the same way and improve regardingly.

As a result it should be stated that the sample size can strongly influence the classification accuracy of a forest.