# IMAGE TUMER LEVEL IDENTIFICATION USING MACHINE LEARNING

*Ankur Chadha, Sudhanshu Mehta*

ankurchadha01@gmail.com, sudhanshu123_mehta@yahoo.co.in

**ABSTRACT**

Mechanism has been developed to identify the tumor stage of the patient from the patient data. For this, training data has been used to train the machine learning algorithm for tumor and non-tumor areas of image. This data has been used for tumor area identification in the test images. Later from the tumor blocks of test image, tumor stage of patient has been identified.

## 1. INTRODUCTION

This document describes the procedure with which tumor region and tumor stage identification has been done. This document also refers to the steps for the environment setup.

## 2. ENVIRONMENT SETUP

For tumor region and tumor stage identification, total 2 laptops/computers has been used. From here, we will call it machine 1 and machine 2. Machine 1 related data has been given in "Machine1Data" folder and Machine 2 related data has been given in "Machine2Data" folder. These two folders are given in the solution folder.

### *2.1 Machine 1 responsibilities:*

a) Image extraction from zipped file.
b) Image splitting procedure (Given .tif file has been splitted into 128x128 multiple frames).
c) Machine learning.
d) Image Testing.

### *2.2 Machine 2 responsibilities:*

a) Feature extraction from training and testing images.

### *2.3 Follow following steps for Environment setup of machine 1:*

a) Take 64 bit window 7 laptop/computer.
b) Install 64-Bit Python installer "python-2.7.13.amd64.msi"
c) Place openslide-win64-20160717 folder in C:\
d) Mention the following in the Path environment variable :
C:\Python27\Scripts;C:\Python27\;C:\openslide-win64-20160717\openslide-win64-20160717\bin
e) Open Command Prompt and install openslide library using following command:
pip install openslide-python
f) Copy the Packages folder in the local machine.
g) Open Command Prompt and navigate to the Packages folder where whl library files are available.
h) Install numpy and scipy libraries using following commands using whl files:
pip install numpy-1.11.3+mkl-cp27-cp27m-win_amd64.whl
pip install scipy-0.18.1-cp27-cp27m-win_amd64.whl
i) Copy and Paste all the files from the DLL folder to C:\Python27\DLLs
j) Paste multiresolutionimageinterface.py file in C:\Python27\Lib folder

### *2.4 Follow following steps for Environment setup of machine 2:*

a) Install Python(x, y), version 2.7.10.0 32-bit on windows 7 machine.
b) While installation, installer will ask for custom installation or full/complete installation. Select the full/complete installation and for rest of installation, install it using default steps on laptop/computer.
c) Python(x, y) Home window will open.
d) In "Options" drop down menu, select none and click on "black and red colored" circular icon to open spyder IDE.
e) From the right top corner of spider window, for browsing to working directory, navigate to Machine2Data folder (given in folder submitted).
f) Navigate to File → Open→ select "Ztesting1.4.py" file in Machine2Data folder.
g) Search for "threshold_otsu" function call and select "Go to definition" option.
h) At the start of function definition, write following instruction and save it.

```
if (image.min() == image.max()):
    return 200
```

**Note**: This is a change in the python library file.

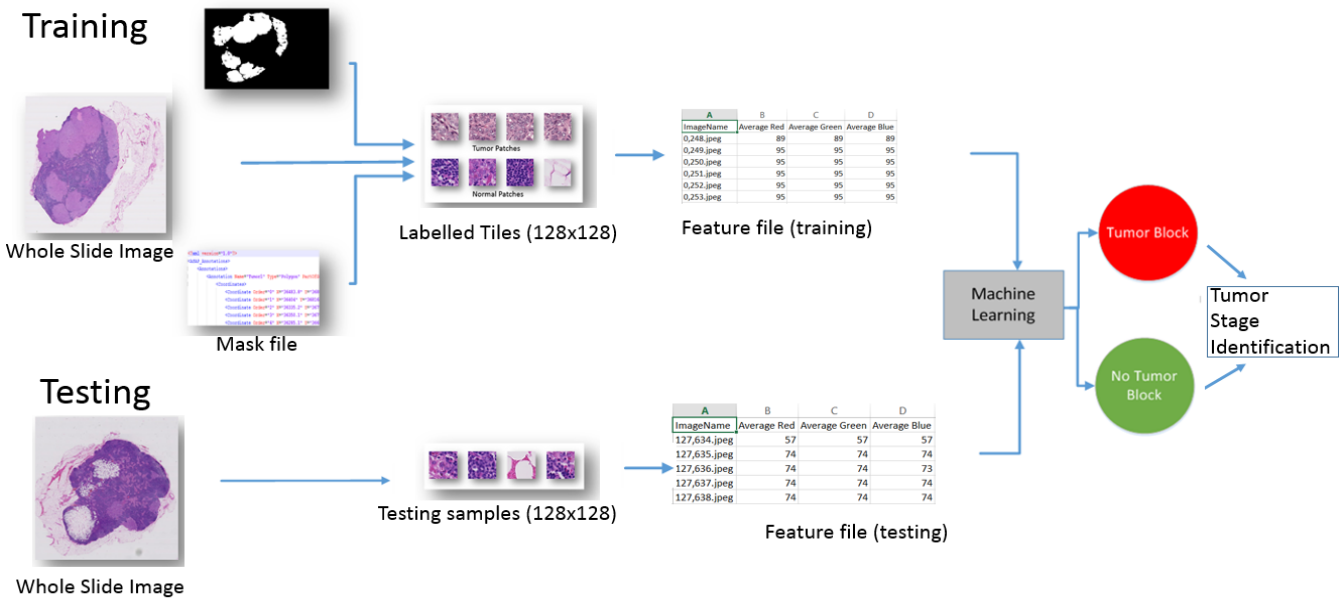


**Figure 1**

i) Close the spider IDE, python(x,y) Home window and restart machine 2.

## 3. PROCESS FOR TUMOR IDENTIFICATION AND TUMOR STAGE DETECTION

The overall process of Tumor Stage identification can be divided into two stages: 1) Image Training 2) Tumor Prediction. Figure 1 explains the process/method used for tumor identification and tumor stage detection. Each stage follows a series of steps in order to train and predict pN-stage of the Tumor in Whole Slide tiff files. We have used multiple libraries of Python programming language in order to build our Algorithm. Few of the significant ones are listed below:

- scikit-learn
- multiresolutionimageinterface
- openslide

### Image Training

**Step 1 (Splitting and classifying WSI file into jpeg image blocks based on XML mask)**

WSI files for Training are split into 128 X 128 pixel image blocks in the jpeg format. These blocks are classified as Tumor and Non-Tumor using the annotated xml mask file.

**Step 2 (Feature Extraction)**

Each image block is processed to extract 46 features per block. CSV Feature file is prepared for each patients slide with labelled flag indicating Tumor/Non-Tumor categories.

**Step 3 (Training the model)**

CSV Feature file is used as an input data for Machine Learning Algorithm. Random Forest classifier is used to train the model.

### Tumor Prediction

**Step 1 (Splitting WSI file into jpeg image blocks)**

Background segmentation of WSI test files is performed to segregate relevant areas in the WSI tiff file from non-relevant ones. Image split algorithm is performed only on the relevant areas to split them into 128 X 128 pixel image blocks.

**Step 2 (Feature Extraction)**

Each non-labelled image block is processed to extract 46 features. CSV Feature file for each patient slide is prepared without any labelled flag.

**Step 3 (Predicting the Tumor Stage using trained model)**

CSV Feature file is used as an input for Trained Machine Learning model. An output file is generated with the predictions Labelling 1 for Tumor and 0 for No Tumor.

**Metastasis Classification:**
Algorithm will calculate the number of Tumor and Non-Tumor files predicted for each patient slide.
From the number we will calculate the percentage of Tumor area with the following formula:

**Tumor %** = [Predicted number of Tumor Files / Predicted number of Tumor Files + Predicted number of Non-Tumor Files)]*100

**Note**: Predicted Number of Tumor files could contain false positives as well. As a result, most of the times the number will be greater than zero even for the Negative patient slide.

**Table 1**

| % Range | Classification |
|---|---|
| 0-5 | Negative |
| 5-10 | ITC |
| 10-20 | Micro |
| >20 | Macro |

*3.1 On machine 1, execute following for training:*

a) Open Image Split(test).py file in notepad++
b) Replace the image variable with the file name and save
   Ex –
   image='patient_109_node_0'
c) Following steps are used in splitting image using Image Split(train).py file:
   - Create a tiff mask using annotated xml file
   - Split the tiff mask file into 128 X 128 jpeg images
   - Split and segregate an Image file(.tif format) into 128 X 128 size Tumor/Non- Tumor regions using jpeg mask images
   - Create a Preview jpeg image to verify the image split.
d) Double click on "Image Split(train).py" file. It will start generating 128x128 sized images in a folder.

*3.2 On Machine 2 follow following steps.*

a) Take the folder which has 128x128 files and copy it in Machine 2.
b) Copy "Ztesting1.4.py" file from Machine2Data folder to the folder copied in previous step.
c) Double click on "Ztesting1.4.py" file. This will start feature extraction of the image. This will take several steps. Number of files processed in the folder will be shown on screen.
d) At the end, ztest.csv file will be generated in the same folder with all the features.

*3.3 On machine 1, execute following for testing:*

a) Following steps are used in splitting image using Image Split(test).py file:
   - Perform Background segmentation.
   - Split the Test Image(.tif file) into 128 X 128 jpeg images.
b) Double click on "Image Split(test).py" file. It will start generating 128x128 sized images in a folder.

*3.4 On Machine 2 follow following steps:*

a) Take the folder in which has 128x128 files and copy it in Machine 2.
b) Copy "Ztesting1.4.py" file from Machine2Data folder to the folder copied in previous step.
c) Double click on "Ztesting1.4.py" file. This will start feature extraction of the image. This will take several steps. Number of files processed in the folder will be shown on screen.
d) At the end a ztest.csv file will be there in the folder. Having all the features.

*3.5 Following steps are used to train and test machine learning algorithm:*

a) Rename the ztest.csv file of Training Data as train.csv
b) Rename the ztest.csv file of Test Data as test.csv
c) Place both the above mentioned files along with output-rf.csv and rf.py in the same folder
d) Execute rf.py file in command line
e) Wait for the execution to complete and note down the predicted count of Tumor and Non-Tumor files.
f) Based on Predicted count values the percentage of Tumor is calculated and Metastasis Classification is made in accordance with the Table 1

*3.6 Identify the pN tumor stage:*

After calculating the Metastasis Classification for each patient slide, lymph node classification (pN-stage) is determined based on the Algorithm mentioned on the Grand Challenge website and stage_labels.csv is prepared.

## 4. REFERENCES

[1] Camelyon 2017: https://camelyon17.grand-challenge.org/

[2] Veta, M., et al.: Assessment of algorithms for mitosis detection in breast cancer histopathology images. Medical image analysis 20(1), 237–248 (2015)