Research Paper: Machine Learning Application for Brain Tumor Localization in the Repository of Molecular Brain Neoplasia Data (REMBRANDT)

Machine Learning for Health Care (Heinz 95-845) - Lukas Mohs

1. Abstract

Within this paper, I want to present the putcome of the combined application of *Computer Vision* (CV) and *Machine Learning* (ML) to the *Repository of Molecular Brain Neoplasia Data* (REMBRANDT) dataset. This dataset contains pre-surgical *magnetic resonance* (MR) multi-sequence images of 130 patients. Based on a visual analysis and the application of several ML algorithms, I tried to predict the location of the brain tumor and compare it to the evaluation of three Radiologist, which defined the affected brain part and the potential impact on the brain functionality.i

2. Introduction and Background

2.1 Brain Cancer

Among all types of cancer, brain cancer is one of the deadliest even if it's not one of the most common ones. With a so-called *5-Year Relative Survival Rate* of 32 % for white people and 39 % for black people it ranks on place 7 out of 26 for the lowest survival rate. Specific cancer types like *Glioblastoma*, a very fast growing type of tumor, even has a a rate of 5%. Especially in the later years of life, this rate decreases heavily for all types of brain cancer. (American-Cancer-Society) The high variance in the surivial rate is given by many factors such as the type of tumor, the location and of course whether it was treated. Especially the latter ones formed a major part of scientific studies that included ways of scanning the human brain for malignant tissue as well as the way of stopping the growth of the tumor cells. (World-Health-Organization)

2.2 Magnetic Resonance Imaging (MRI)

Magnetic Resonance Imaging (MRI) addressed the challange of *scanning* the human body by using *radiology* to see different layers of the inner tissue or organs. These different layers can be combined together to construct a 3-dimensional model of any part of the body. Mainly a strong *magnetic fields* is used in combination with *radio waves* and so-called *field gradients.* In comparison to *Computer Tomography* (CT), MRI doesn't rely on *X-radiation*, which qualifies it as a less harmful method. It should be mentioned that the magnetic waves of MRI can affect *cardiac pacemakers* so that exceptions apply. (Edelman and Warach, 1993)

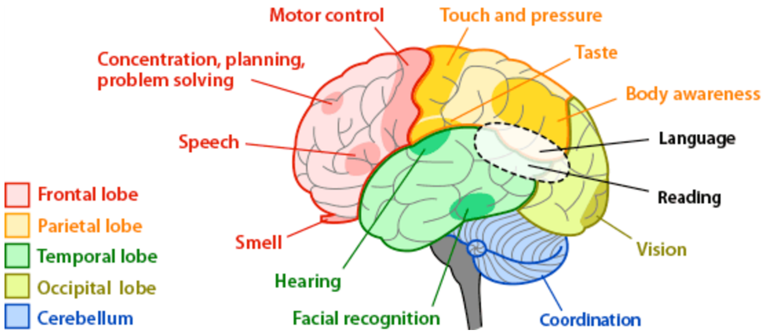


Figure 1: Brain Areas (source: Arizona State University)

2.3 The Human Brain

Especially the development and refinement of the MRI technology favored advanced studies about the structure and functionality of the human brain. In order to understand the classification task of the presented algorithm, the major parts of the human brain are shortly described and visualized in Figure 2.3:

- *Frontal Lobe:* resides in the front of the head and is the only lobe on the lateral surface and separated from the other ones. It is responsible for problem solving, planning but also processes stimuli from the nose.

- *Temporal Lobe:* caudally joins the parietal and occipital lobe without a clear boundary. Researchers could show that it hosts the function of recognizing faces.

- *Parietal Lobe:* is Clear separated from frontal lobe but merges into temporal and occipital lobe. It is important for the control of the body including the sense of touch.

- *Occipital Lobe:* has also no clear coundary to its neighbours (parietal and temporal lobe). The stimuli of the eye are for example processed by this lobe.

- *The Insula:* is a small portion of the cerebral cortex and hidden by the other outer lobes. It is believed to be responsible for consciousness.

- *The Cerebellum:* is placed on the back of the head below the occipital lobe and next to the *brain stem*. It could be shown that it is responsible for fine motoric controls and languages.

(Duvernoy et al., 2012)

2.4 Computer Vision

Computer Vision (CV) is a field of *Computer Science* that focusses on the way of how to digitally process, analyze and understand images. Driven by the development of cheap cameras and computing power, CV is applied in many cirsumstances to support or replace the human's richest sense: the eye. To provide a highlevel understanding of a CV identification or classification approach, one has to consider the digital representation of an image on the

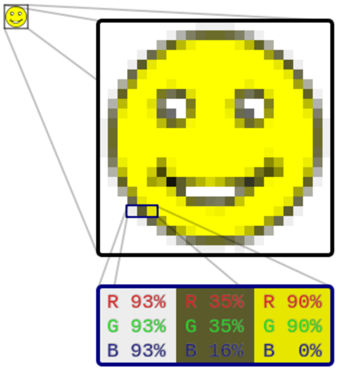


Figure 2: Raster Graphics (source: Wikipedia)

computer. This can be decomposed to a two-dimensional matrix, where each entry holds a tupel of three values: the ***R**ed*, ***G**reen* and ***B**lue* (RGB) channel. This tupel identifies how the color of one specific *pixel* is composed. These matrix of pixels is often referred to as *raster graphic*. Figure 2.4 demonstrates this representation for three specific pixels within an image. In comparison to colorful images, *grayscale images* just has a single channel, which indicates the brightness between black (0 %) and white (100 %). Most common file types represent the channel values in a range between 0 and 255. The brightness of a colorful image is therefore determined by the average of the sum of each pixel's RGB values. The contrast of an image is referring to the variance of the sum of each pixel's RGB values: the higher the difference, the stronger the contrast. These concepts are important to later understand the image processing algorithm.

2.5 Machine Learning

In the recent years, *Machine Learning* (ML) genereated special attention due to it's successful application in many fields of industry. Generally, ML can be understood as an automized approach of analyzing and learning specific patterns within a big series of data. For this purpose, statistical principles are implemented on a given *training set* afterwards to predict the class or value of new instances. ML is used in this scenario to evaluate the output of the image processing step and locate the brain tumor within a given sequence of MRI scans. In particular, the probabilistic *Naive Bayes* classifier will be used in combination with a *Decision Tree* due to their simplicity, whcih allows a solid interpretabiliby.

3. Implementation: ML Prediciton based on CV Pattern Matching Outcomes

Within this section I want to describe the setup and implementation of the *ensemble* of CV and ML that is used to locate the brain tumor within the image sequence retrieved from the MRI scan. Ensemble means that a two-step process is used to realize the final prediction:

- First, the images fo the REMBRANDT dataset are retrieved, preprocessed and analyzed by the application of several pattern matching methods.

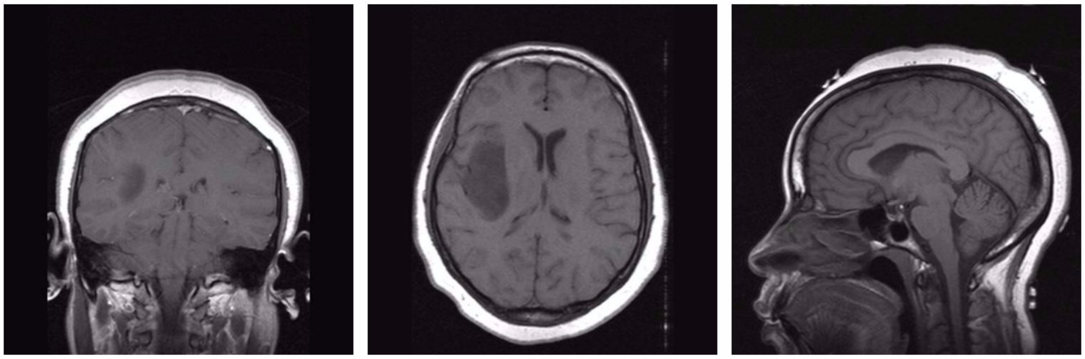


Figure 3: Sequence Comparison (back to front, top to bottom, left to right)

- Then, the output of the analysis is written to *CSV* file to be taken as an input for the ML learning and prediction approach.

The most recent version of the source code of this project, which is based on *Python* for the image processing and *R* for the ML part, can be found on Github:
https://github.com/lukasmohs/TumorDetection .

3.1 REMBRANDT Image Data Set

At this point, the input data in form of a series of image sequences is described: The data of REMBRANDT was generated through the Glioma Molecular Diagnostic Initiative and contains the pre-surgical MR multi-sequence images from 130 patients. This image dataset is enriched by a professional classification of three differen radiologists for each individual stating the tumor location within the brain. Each sequence can be defined as one specifc direction of slicing, which was used in order to scan the patient's head. This means that each patient was scanned either horizontally or vertically from different angles. In figure 3.1, one image of each of these sequnces for one patient is provided. (Kenneth Clark)

Comparing the images provided from each sequence, we can see that the image on the left (*back to front*) and the one on the right (*left to right*) both contain more *noise* in terms of head parts that don't belong to the actual brain. The automatic localization within the image is additionally more complex than for the one in the middle (*top to bottom*) due to the fact that images might not be cropped similarily so that a localization would need to be realized relatively to the shape of the skull, which would be more error-prone. Therefore, the *top-down* sequence of each patient was chosen to perform the analysis. It should be mentioned that the sequences of each patient are not of similar length meaning that they contain a different numbers of single images.

3.2 Image Preprocessing

labelimage-preprocessing As the first step of the CV part, the top-down image sequences for the patients were retrieved from the *Cancer Image Archive*:
https://wiki.cancerimagingarchive.net/display/Public/REMBRANDT .

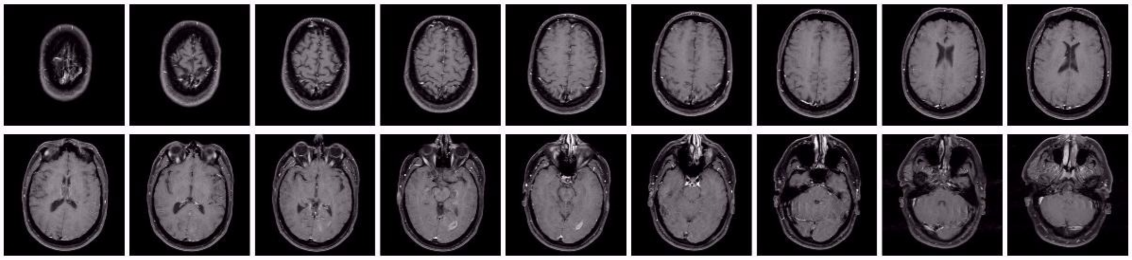


Figure 4: Full Top-down Sequence of Individual

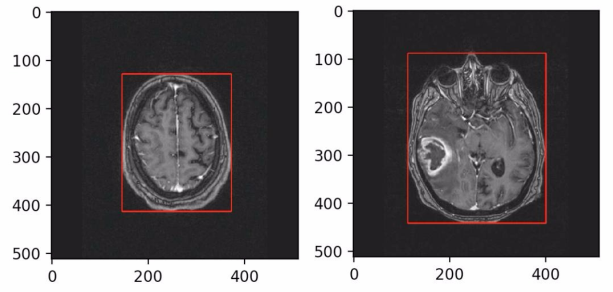


Figure 5: Image Cropping

An entire sequence of scans is shown in Figure 3.2. In order to better determine the location within one single image and to improve the pattern matching performance, the images should b cropped to the skull in the middle of the image. Hence, the algorithm contained in the *cropImages.py* file was developed that iterates over the *x-axis* and *y-axis* of each image and identifies the important parts of the image based on the brightness (see: secttion 2.4). It's performance is vidualized in Figure 3.2. A new image is written to a file and used for the next processing step.

3.3 Pattern Matching

After preprocessing the images, the main part of the algorithm that extracts features out of the image sequences for the following ML approach can be executed. The major idea of identifying a tumor within each image is based on a *similarity measure* between a so-called *template* and the *target image*. For the actual matching, the *OpenCV* library version *2.4.13.2* for Python was used (Opencv-Dev-Team, a). In the following, I will explain the development of the matching template as well as the exact matching application.

3.3.1 Template Development

To generate the template as an input for the matching process, I implemented an algorithm that develops a general pattern in form of an image that represents the average of a *training image set*. As explain in section: 2.4, a matrix can be used to represent each pixel of an image. Therefore, I manually selected some input tumor images and applied the algorithm provided in the *generateTemplate.py* file. Each image is streched to the same size and to

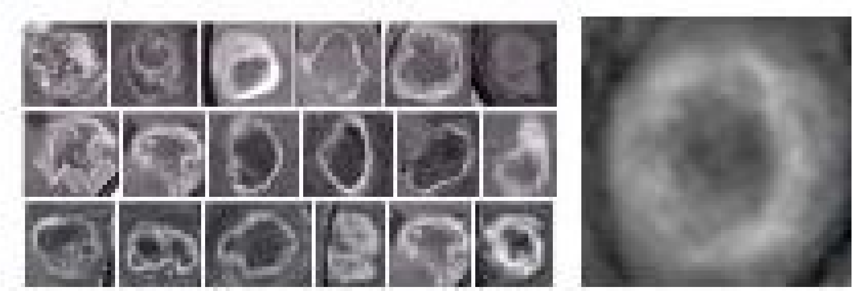


Figure 6: Template Generation (left: input, right: output)

then calculate the average template matrix, the algorithm iterates over each pixel of each image and adds together all partial RGB values. This can be formulated as follows:

May $A^i_{j,k,c} \in \mathbb{N}$ and $0 < i < I, 0 < j < J, 0 < k < K, 0 < l < L$

where I is the number of input pictures, j is the number of rows, k is the number of columns and c refers to the color index. Then, the average template image can be formulated as follows:

$$\not\phi_{j,k,c} = \frac{\sum_{i=1}^{I} A^i_{j,k,c}}{I}$$

A^i can then be expressed as:

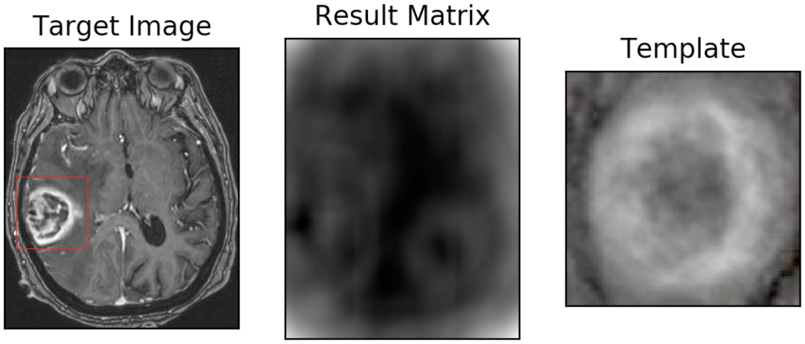
$$(A)^i = \begin{bmatrix} a_{1,1} & a_{1,2} & ... & a_{1,k} \\ a_{2,1} & a_{2,2} & ... & a_{2,k} \\ ... & ... & ... & ... \\ a_{j,1} & a_{j,2} & ... & a_{J,K} \end{bmatrix}$$

and $a_{j,k} \in \mathbb{N}^3$ is refering to one pixel consisting of 3 color channels

Figure 3.3.1 demonstrates the input and result of this template creation process.

3.3.2 Matching Algorithm

The core function of the data extraction from the input files is realized within the *processImages.py* file. As mentioned before, OpenCV was used for matching the previously generated template (see: section 3.3.1). The corresponding method in the OpenCV library is called *cv2.matchTemplate()* and it basically slides the template image over the target image and computes how simililar it is to the current frame. The output is a new image in grayscale, where each pixel that ranges from 0 to 255 indicates how well the template matched the specific frame. It needs to be mentioned that depending on the chosen method, this result matrix can also be inverted meaning that the brightness can also indicate the output of the *error term*. As an argument, the function takes a specific method out of a set of six possible ones. Since all methods for creating the *result matrix* are based on a similar mathematical approach, I want to discuss the best performing ones, which are called


Target Image

Result Matrix

Template

Figure 7: Target Image, Result Matrix and Template

CV_TM_SQDIFF_NORMED and *CV_TM_CCORR_NORMED*. The underlying mathematical model for CV_TM_SQDIFF_NORMED can be formulated as:

$$R(x,y) = \frac{\sum_{x',y'}(T(x^{'},y^{'}) - I(x+x^{'},y+y^{'}))^2}{\sqrt{\sum_{x',y'} T(x',y')^2 * \sum_{x',y'}(I(x+x',y+y')}}$$

where T is the template, which is matched against the Image I resulting in the output of this metric. Hence, the value of R at position (x,y) refers to the goodness of the fit of T on I at this point.
The metric of CV_TM_CCORR_NORMED is analogously defined as:

$$R(x,y) = \frac{\sum_{x',y'}(T(x^{'},y^{'}) * I(x+x^{'},y+y^{'}))^2}{\sqrt{\sum_{x',y'} T(x',y')^2 * \sum_{x',y'}(I(x+x',y+y')}}$$

(Opencv-Dev-Team, b)

Since the result matrix is also interpretable as an image, one can visually see the results of the matching algorithm. Figure 3.3.2 visualizes the result matrix for a given matching problem. The red square marks on the target image marks the corresponding position, which will be discussed again.

3.3.3 Match Processing

In this subsection, the focus is directed to the selection of the best match out of a sequence of images. When the result matrix for one image has been determined, it is required to further analyze the obtained result matrix in order compare it to the ones of other images. This is given by the fact, that a similar overall brightness level of the template favors a good match within an image sequence just due to the small distance in the color range. To overcome this problem, I invented a score that deals with this issue by computing a relative score for each retrieved result matrix. A specific characteristic of a good match is that the contrast between the matching position and its sorrpounding is quite high. This observation can be explained by the fact that the developed template his ring-shaped which lets it fit pretty well at a certain points but relatively bad in the area around it. To take

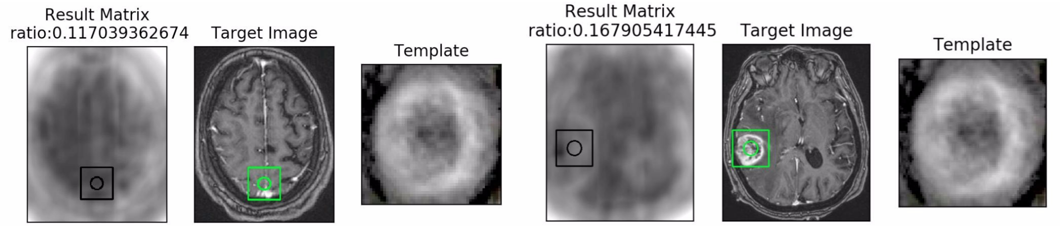


Figure 8: Application of Matching Ratio

advantage of this characteristic, the score basically evaluates the difference between the spot of the best matching point and the area. Hence, the template window size is taken around the matching point and a small circle is placed in the middle. This score, which can be uderstood as a ratio, can be formulated as follows:

IConsider X as the set of pixels within the template frame, where each $w \in X$ is defined by its two coordinates $w = (x, y)$. The set of pixels within that inner circle with radius $R \in \mathbb{R}$ around the centre $w_1 = (x_1, y_1)$ is defined as:

$$W_1 = \{(x, y) \in x : (x_1 - x)^2 + (y_1 - y)^2 \leq R^2\}$$

The set of pixels around the circle is then

$$W_2 = X \setminus W_1$$

Now, consider the function :

$$h : X \to [0, 255] \in \mathbb{N}$$

which maps each pixel to its brightness (the quality of the fit for this position). Then the total brightness of $K_1 \in \mathbb{N}$ within the circle evaluates to:

$$K_1 = \sum_{w_1 \in W_1} h(w_1)$$

The total brigtness outside the circle is accordingly:

$$K_2 = \sum_{w_2 \in W_2} h(w_2)$$

Then, the final ratio r can be computed as:

$$r = K_1 \setminus K_2$$

A visualization of this approach is shown in Figure 3.3.3, where the ratio identifies the best matching image and position in the sequence.

3.3.4 Iteration steps and Output

At this point, I want to shortly describe the iteration of the algorithm in detail so that potential advancements and modifications can be considered easily. After the algorithm has computed the template image, it traverses the directory of images for each patients image sequence. For each image in the sequence the image is cropped and matched against the template. Here, two variables can be used to modify the execution:

- *methods* is a list of all available methods within the pattern matching function of OpenCV, which can be added or removed from the list. However, for each method the execution is realized ones and the best match added to the output set.
- *templateSizingSteps* is also a list that allows the programmer to specify the if the template should be resized to other proportions before matching. For each item in the list, the algorithm performs another execution and returns the best outcome.

The information that are gathered through the execution are written to the *features.csv* file, where for each patient and each applied method the following information are persisted:

- *relative position z* indicates in which image of the sequence with regard to the total number the bet match has been found.
- *position y* determines the y coordinate of the position of the match
- *position x* determines the x coordinate of the position of the match
- *best probability* is the sum of all pixel values within the best matching template frame
- *best probability ratio* is the value of relative probability score (see: section 3.3.3)

Each row of this intermediate file is enriched with the most probable real tumor location. Since three radiologists were consulted for the evaluation of the tumor, the *readTruthTable.py* file reads the provided information, evaluates the average of the provided expert opinions and returns them so that they can be written to the intermediate file.

3.4 Classification

Within the previous sections, the comlex process of the information gathering was described as the extraction of features became the main part of this project. Now, this input can be used for the final classification task. It should be mentioned that the REMBRANDT data set just provides reliable labels for 33 patients (*900-00-1961* to *900-00-5477*) so that an excessive analysis can't be realized based on the given data set.
To clarify the actual outcome of the following prediction, the provided labels for each instance should be reviewed: The consulted radiologists used the following codes to identify the tumor location within the brain (see: section 2.3):

- 0: no tumor
- 1: frontal lobe

- 2: temporal lobe
- 3: insular lobe
- 4: parietal lobe
- 5: occipital lobe
- 6: brainstem
- 7: cerebellum

Due to the fact that insular lobe is very small and a good localization is very hard since it is sorrounded by the parietal lobe when slicing top-down, these two instance values were combined for the prediction purpose. An *R* script named *classifyImages.Rmd* is performing this second step of the ensemble classification. First, the extracted data is split into a training and test set containing the application of two ML algorithms that are trained on the test set and then perform an analysis on the test set. These algorithms are:

- Naive Bayes
- Decision Tree

4. Discussion

Within this final chapter, I want to conclude the learning outcomes of this study. Primarily, it should be stated that the chosen approach was realized with limited resources regarding computing power, the training data as well as time. Most successful modern image classifiers rely on so-called *deep learning* algorithms that use a set of many layers of *software neurons* to imitate the functionality of the brain. Large implementations include hundrets of millions of entity types and billions of relationships and are trained on very large data sets (Hof). In comparison to these applications, this paper addressed a different topic: One the one hand it could be shown that brain tumors share some visual characteristics

References

American-Cancer-Society. Age-adjusted seer incidence and u.s. death rates and 5-year relative survival (percent) by primary cancer site, sex and time period. URL `"https://seer.cancer.gov/archive/csr/1975_2012/results_merged/topic_survival.pdf"`.

H.M. Duvernoy, J.L. Vannson, P. Bourgouin, E.A. Cabanis, F. Cattin, J. Guyot, M.T. Iba-Zizen, P. Maeder, B. Parratte, L. Tatu, et al. *The Human Brain: Surface, Three-Dimensional Sectional Anatomy with MRI, and Blood Supply.* Springer Vienna, 2012. ISBN 9783709167922.

Robert R Edelman and Steven Warach. Magnetic resonance imaging. *New England Journal of Medicine*, 328(10):708–716, 1993.

Robert D. Hof. Mit technology review: Deep learning. URL `"https://www.technologyreview.com/s/513696/deep-learning/"`.

Smith John Freymann Justin KirbyPaul Koppel Stephen Moore Stanley Phillips David Maffitt Michael Pringle Lawrence Tarbox Fred Kenneth Clark, Bruce VendtKirk. The cancer imaging archive (tcia): Maintaining and operating a public information repository".

Opencv-Dev-Team. Opencv 2.4.13.2 documentation, a. URL "http://docs.opencv.org/2.4.13.2/".

Opencv-Dev-Team. Template matching, b. URL "http://docs.opencv.org/2.4/doc/tutorials/imgproc/histograms/template_matching/template_matching.html".

World-Health-Organization. Early detection of cancer. URL "http://www.who.int/cancer/detection/en/".