FACULTY OF SCIENCE
UNIVERSITY OF COPENHAGEN

MR Image texture analysis applied to the diagnosis of Alzheimer's Disease

Mathias Bjørn Jørgensen & Mirza Hasanbasic

5th June 2016

Contents

List of Figures

List of Tables

List of Corrections

Abstract

Chapter 1

Introduction

In this report we will examine MRI data of the hippocampus using image texture analysis and apply machine learning. We have XX normal controls and XX Alzheimer's Disease (AD) patients. They are split into a training set (XX control and XX AD) and a test set (XX control and XX AD).

We will be using two different texture analysis on the data, XX and the gray level co-occurrence matrix (GLCM). We will calculate the GLCM using two different methods, one in 2D that that runs along the z-axis with angel 90 and distance 1 (Change depending on the results, and more research might include multiple angles). Castellano [2] The other method is calles voxel-based GLCM in 3D**FiXme Note: citation** space (VGLCM-TOP-3D), and is from the paper Voxel-Based Texture Analysis of the Brain (indsæt biblografi). We want to see if there is a difference between diagnostising a AD succesfully, by calculating the co-occurence matrix in 3D compared to 2D, and how well the GLCM methods work compared to XX. To do that we will use two different machine learning methods, k-NN and Gaussian mixture, based on each of the image texture models.

FiXme Note: citation

We will also try to replicate the analysis from MRfreeborough [1] and meanwhile tell if we can get better accuracy

1.1 Problem Definition

Is it possible to classify MRI data of the hippocampus into groups of healthy controls vs Alzheimer's patient, using a predefined set of image texture metrices, with an accuracy greater than 80%?**FiXme Note: Mere i intro. Måske 3D vs 2D**

FiXme Note: Mere i intro. Måske 3D vs 2D

1.2 Alzheimer's Disease

About 70% of the risk is believed to be genetic, where other factors include head injuries, depression or hypertension. Magnetic [3] **FiXme Note: Mere til AD**

FiXme Note: Mere til AD

Chapter 2

Data

Chapter 3

Method

3.1 Image texture analysis methods

Et image texture er bare et sæt af metricer? som udregnes for at opfatte texturen på et billede. Image textures giver os denne information.

Der findes flere approaches til hvordan man udregner image textures, vi bruger den statistiske vej.

3.1.1 Co occurrence matrix

The co-occurence matrix (COM) is a method to measure the greyscale intensity (GI) of an image, it is defined as the distribution values at a given offset. The element (i,j) in the COM, **C**, over an image $n \times m$, with $\Delta x, \Delta y$ being the parameterized offset, is calculated by

$$C_{\Delta x,\Delta y}(i,j) = \Sigma_{p=1}^{n}\Sigma_{q=1}^{m}\begin{cases}1, & \text{if } I(p,q)=i \text{ and } I(p+\Delta x, q+\Delta y)=j\\ 0, & \text{otherwise}\end{cases}$$

FiXme Note: find reference The element (5,4) in the COM can be translated to meaning how many times there exist an element in the image with GI **FiXme Note: level or intensity?** 5 and another element offset $\Delta x, \Delta y$ from the originial with GI 4, i.e. if the offset is (1,0) and the first element is (x,y)(4,3) with GI 5 it would mean that element (x,y)(5,3) would have GI 4. If COM(5,4) is ten, it translates into there being ten instances with element (x,y) = 5 and (x+Δx,y+Δy) = 4.

FiXme Note: find reference
FiXme Note: level or intensity?

A single image have multiple COMs as different offsets creates different relations. Consider a 3 by 3 matrix looking at element (2,2) we can then create eight different offsets, (1,0),(1,1), (0,1)),(-1,1),(-1,0),(-1,-1),(0,-1),(1,-1), however they are not unique. **FiXme Note: Lave 3x3 matrice med offsets.** Focusing on the two offsets (1,0), (-1,0) in element (2,2) and (1,2) with GI 1 and 2 respectfully increases the entry COMs$_{1,0}$(1,2) and COMs$_{-1,0}$(2,1) with one, showing that COMs$_{1,0}{}^{\mathbf{T}}$ = COMs$_{-1,0}$. This leaves four different offsets for analysis (1,0),(1,1), (0,1)),(-1,1), which are denoted angle 0°

FiXme Note: Lave 3x3 matrice med offsets.

The co-occurrence matrix is quadratic with the number of rows and columns equal to the amount of GI, for example if we have 256 GI we get a 256 × 256 COM. nd [0 1]]FiXme Note: Show a example on a 5x5 matrix with GI 8, offset [1 0] and [0 1]**FiXme Note: Show example on a 5x5 matrix with GI 8, offset [1 0] and [0 1]**

Helt specifikt udregner vi en co occurence matricer, hvor man kan få en del numeriske features fra gray tones. Disse kan ses i appendix A på s. 14. Hvordan vi har 256x256 matrix, med en distance på δ og angles θ, hvor $\delta = \{1, 2, ..., 10\}$ og $\theta = \{0°, 45°, 90°, 135°\}$. Ud fra denne GLCM matrix så kan vi udregne textural features fra de 100 GLCM af MRI grayscale image data set. Snakke om hvordan den fungere, med distance og grader

3.2 Machine learning methods

3.2.1 Crossval

3.2.2 Feature selection

Forward feature selection

Naive

3.2.3 K-nearest neighbors algorithm

k-NN for short is a method that is used for classification and regression. Where the output is a class and member of this class, and this object is classified by its neighbors. For instance, if we chose k to 1, then the object will be assigned to the class of the single nearest neighbor.

The algorithm consist of training examples, that are vectors in multidimensional space, with each its label. The most used distance metric is Euclidean distance.

The drawback of k-NN is that classification can be skewed in that way, that the more frequent class tend to dominate the prediction of new examples, because they tend to be common among the k-NN due to their large number.

The way we wish to implement the k-NN in matlab is, first we handle the data, then we will calculate the distance between two data instances and after that, we can locate k most similar data instances and generate a response from a set. After all this is done, we have to summarize the accuracy of predictions.

Dette vil være en lille introduktion til de tools vi bruger til at analysere vores data med. Da vores data er MRI skanninger af hjernen, som er nogen voxels[1] som bliver repræsenteret i 3D.

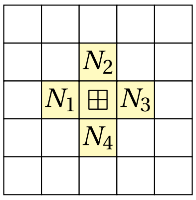
[1] noget her

3.3 Erode

Normalt brugte man dette til binære billeder, men senere hen er det udvidet til også at omfatte grayscale billeder. Grunden til at dette bruges er for at fjerne støj på billedet.

Vi bruger Erode på vores MRI scanning, da der kan være gray-bit mix og derfor fjerner overflydig og blandet data. Hvis vi starter på hvordan 2D virker, så

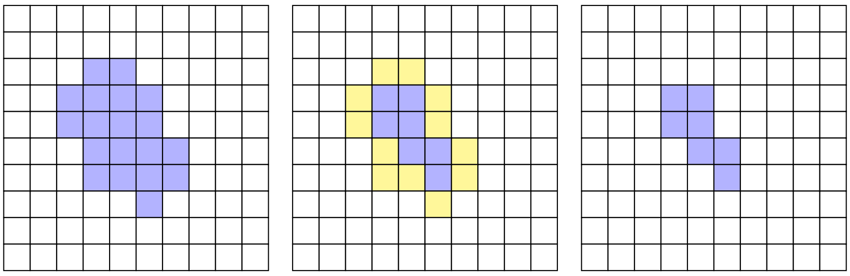
For at illustrere hvordan erosion virker, gør vi det på et 2D plan, betragt figur 3.1, hvor vi bruger et plus til at fjerne støj.



Erosion2D

Figure 3.1: Text

Så med figur 3.1, bruger vi denne på figur 3.2.

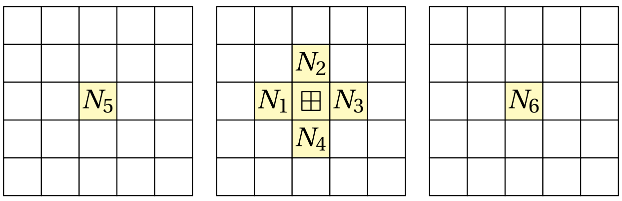


rosionExample

Figure 3.2: Left: Middle: Right:

Således er støjen nu fjernet. Vi vil udvide dette til 3D, da vores MRI er i 3D. Som det ses på figur 3.1 har denne 4 naboer den tjekker, når man udvider til 3D, får vi 2 nye naboer, dvs 6 naboer i alt. Hvis en af dem er udenfor den ønskede matrix, ekskluderes pixlen.

Da vores data er i 3D, så i udvider vi erosion, hvor det stadig er et plus, men med 2 ekstra naboer Now we expand this cross for the 3D and its the same concept for 3D. Now it 6 neighbours instead of 4, where we expand it for the 3D



Erosion3D

Figure 3.3: Text

3.4 Principal Component analysis

3.4.1 Application to images

3.5 K-nearest neighbors

3.5.1 Cross validation

Image texture PCA Principal Component Analysis Application to images Machine learning (Knn, Ann, Gaussian)

noget tekst

noget mere tekst **FiXme Note: en note**

endnu mere tekst

og til slut mere tekst **FiXme Error: en fejl**

FiXme Fatal: her er noget galt
FiXme Note: en note
FiXme Warning: en advarsel – noget er helt forkert
FiXme Error: en fejl

Chapter 4

Implementation

Chapter 5

Result

Chapter 6

Discussion

Nogen angle og planes er ens

Når vi loader ind i en train og test, så

Chapter 7

Conclusion

Appendices

Appendix A

Co occurrence matrix derivation features

ationfeatures

$$C_x(i) = \sum_{j=1}^{N} C(i,j)$$

$$C_y(i) = \sum_{i=1}^{N} C(i,j)$$

$$C_{x+y}(k) = \sum_{\substack{i=1 \\ i+j=k}}^{N} \sum_{j=1}^{N}, \quad \text{k = 2, 3, ... , 2N}$$

$$C_{x+y}(k) = \sum_{\substack{i=1 \\ |i-j|=k}}^{N} \sum_{j=1}^{N}, \quad \text{k=0,1,...,N-1}$$

$$f_1 = \sum_{i=1}^{N} \sum_{j=1}^{N} \{C(i,j)\}^2 \tag{A.0.1}$$

AngularSecond

$$f_2 = \sum_{n=0}^{N-1} n^2 \left\{C_{x+y}(k)\right\} \tag{A.0.2}$$

Contrast

$$f_3 = \frac{\sum_{i=1}^{N} \sum_{j=1}^{n} ijC(i,j) - \mu_x \mu_y}{\sigma_x \sigma_y} \tag{A.0.3}$$

Correlation

$$f_4 = \sum_{i=1}^{N} \sum_{j=1}^{N} (i-\mu)^2 C(i,j) \tag{A.0.4}$$

Variance

$$f_5 = \sum_{i=1}^{N} \sum_{j=1}^{n} \frac{1}{1+(i-j)^2} C(i,j) \tag{A.0.5}$$

InverseDiffer

$$f_6 = \sum_{i=2}^{2N} iC_{x+y}(i) \tag{A.0.6}$$

SumAverage

$$f_7 = \sum_{i=2}^{2N} (i-f_6)^2 C_{x+y}(i) \tag{A.0.7}$$

SumVariance

$$f_8 = \sum_{i=2}^{2N} C_{x+y}(i) \log(C_{x+y}(i)) \tag{A.0.8}$$

SumEntropy

$$f_9 = -\sum_{i=1}^{N} \sum_{j=1}^{N} C(i,j) \log(C(i,j)) \tag{A.0.9}$$

Entropy

$$f_{10} = \text{variance of } C_{x-y} \tag{A.0.10}$$

DifferenceVar

$$f_{11} = -\sum_{i=0}^{N-1} C_{x-y}(i) \log(C_{x-y}(i)) \tag{A.0.11}$$

DifferenceEnt

Bibliography

MRfreeborough [1] Peter A. Freeborough & Nick C. Fox. Mr image texture analysis to the diagnosis and tracking of alzheimer's disease. *IEEE*, 17(3):5, June 1998.

Castellano [2] L.M. Li F. Cendes G. Castellano, L. Bonilha. Texture analysis of medical images. *Neuroimage Laboratory*, page 9, April 2004.

Magnetic [3] Stephane Lehericy & Malgorzata Marjanska & Lilia Mesrob & Marie Sarazin & Serge Kinkingnehun. Magnetic resonance imaging of alzheimer's disease. *Springer Verlag*, 17(3):5, June 2005.