CS229 2015 | Group 277

Learning with Difference of Gaussian Features in the 3D Segmentation of Glioblastoma Brain Tumors

Zhao Chen (zchen89[at]Stanford.edu), **Tianmin Liu** (tianminl[at]Stanford.edu),
Darvin Yi (darvinyi[at]Stanford.edu), Project Mentor: Irene Kaplow

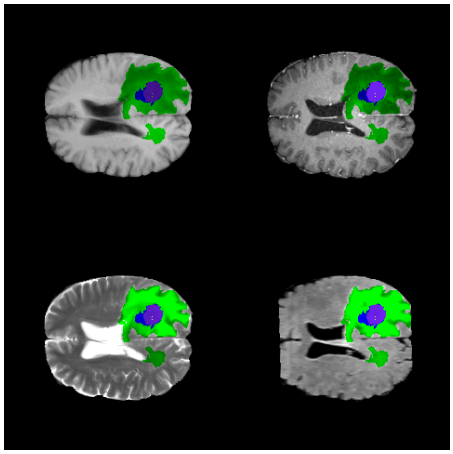
Introduction

Glioblastoma (GBM) is an especially aggressive brain tumor that accounts for over 50% of brain tissue tumor cases. [2] GBMs have a high mortality rate with a one year survival of 50% and a three year survival rate of 90%. [2] Much is still being done to understand GBM and its two subtypes, high grade glioma (HGG) and low grade glioma (LGG), but there is still a gold mine of untapped data. Chief amongst these is imaging data in the form of magnetic resonance (MR) scans. Most methods of analyzing and extracting quantitative information from these imaging data requires some form of segmentation of the GBM, and better yet, classification of the tumor into four sub-categories: necrosis, edema, non-enhancing tumor, and enhancing tumor. To date, the gold standard of segmentation is still human radiologist segmentations. However, with the pure size of imaging data being accrued, manual segmentation of all images is no longer a sustainable system. We propose a statistical learning pipeline that takes difference of gaussian features into a hierarchal neural net to segment and classify tumors into their sub-categories. We thus take as input a 3D MR image and output one of five labels (normal or one of 4 tumor subtypes) for each voxel (the 3D analog of pixel). Our combined median HGG and LGG results in a Dice accuracy score of 0.90 for the whole tumor detection, 0.72 for the tumor core detection, and 0.74 for the active tumor detection. We will see that this is quite competitive with the leading programs at the moment.

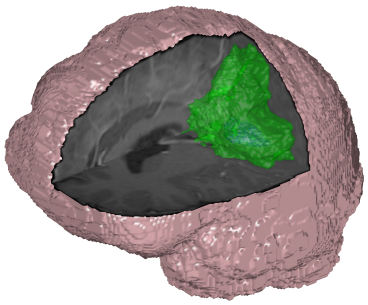
Data

We used the pre-processed training data provided by the MICCAI **Bra**in **T**umor Image **S**egmentation (BRATS) challenge. [9] For each patient, four main MR modalities were given: (1) T-1 pre-contrast, (2) T-2 post-contrast, (3) T-2 Weighted, and (4) FLAIR. The training data comes already pre-processed, which involves skull stripping [1] and co-registration. Co-registration is crucial, as it aligns images for all four modalities such that the voxel found in the image coordinates $[x, y, z]$ in all four modalities will point to the same coordinates in real space.

Figure 1a shows an example of a single cross section of the four modalities, juxtaposed with the expert segmentation of the tumor which we will use as ground truth. Note that it is co-registration which allows for reliable overlay of the four modalities with the expert segmentation. Note also that the FLAIR image is somewhat cut off due to the nature of the MR scan. However, big data algorithms are generally robust against small artifacts like this.



(a) Modalities
(ul: T1-Pre, ur: T1-post, dl: T2W, dr: FLAIR)



(b) 3D Rendering of Tumor

Figure 1: Tumor Visualization

Throughout this project, all of the analysis and methodology built around the data interprets the data as a 3-dimensional object. Each slice gives information on MR intensity in `[x,y]`, but the slices themselves represent the MR information in `z`. Thus, the data can be imagined as a 3-dimensional cube as seen in figure 2, and we can extend classic 2D imaging techniques to an additional dimension to work on our data.

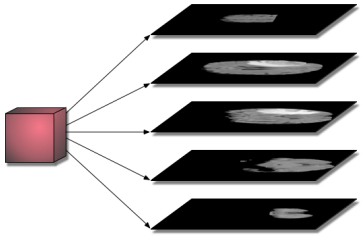


Figure 2: Data Visualization

Feature Extraction The Difference of Gaussian (DoG) convolution filter has been used as a blob detector for some time in 2D, and it features prominently in the Scale Invariant Feature Transform (SIFT) algorithm.[7] We can see SIFT's feature detection applied to the iconic sunflower image and one of our Brain MR slices in figure 3. On the sunflower image, we can see that the features (based on DoG filters) find almost all circular objects in the image. Similarly, on the brain MR slice, the most predominant SIFT feature is the tumor, which is quite blob like.

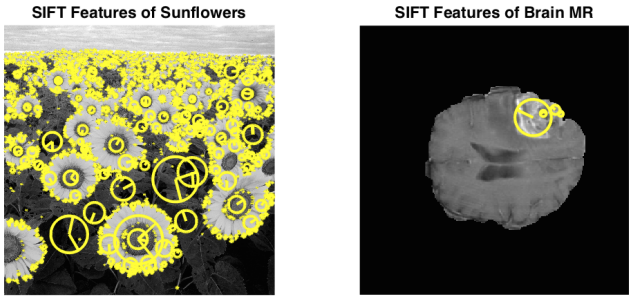


Figure 3: SIFT as Evidence of Blob Detection

Our project is in 3D, and so we propose building a bank of 3-Dimensional DoG filters (figure 5), all of different scales. By convolving our 3D data with these 3D filters, we will be able to build a blob profile feature vector for each pixel of our data. In addition to the robustness of DoG filters as blob detectors, we can also notice two more important features: (1) DoG filters are rotationally symmetric and (2) DoG filters are efficient for 3D convolution. The rotational symmetry reduces system bias by not placing special importance on a set of discretized directions. This makes DoG filters more robust than directional filters, such as the 3D Gabor filters which were used in first iterations of our work. The DoG filter convolution is easy to calculate by separating the Gaussian into each linear dimension and performing all calculations in Fourier space. Thus, for our data volume V and our DoG filter $f \equiv G_1 - G_2$ (where $\mathrm{Var}(G_1) = \sigma_1^2$ and $\mathrm{Var}(G_2) = \sigma_2^2$).

$$\begin{aligned}(V * f)[x,y,z] \equiv &\sum_n \sum_m \sum_l V[n,m,l] f[x-n, y-m, z-l] \\ =&\mathrm{fft}^{-1}\left\{\mathrm{fft}(V)\mathrm{fft}(f)\right\}[x,y,z] \\ =&\mathrm{fft}^{-1}\left\{\mathrm{fft}(V)\mathrm{fft}(G_1 - G_2)\right\}[x,y,z] \qquad (1)\end{aligned}$$

where we know $\mathrm{fft}(G_i) = e^{-\frac{1}{2}\sigma_i^2(m^2+n^2+l^2)}$. As mentioned, we can further simplify the calculation by taking each linear dimension separately.

As a final point, we note similarities between the DoG filter and the edge-sensitive Laplacian of Gaussian (LoG) filter,[7] as visualized in figure 4. Thus, our DoG is not only a blob detector, but also a makeshift edge detector. We will get a high peak at a pixel if it is at the center of a blob. However, we will be getting a very low absolute value if we're at an edge. Because our segmentation program is very interested in finding accurate edges, this edge detection aspect of the DoG filters is very useful.

We thus associate with each voxel $[x, y, z]$ the following set of features:

- ONE Voxel intensity $V[x, y, z]$.
- ONE Voxel intensity gradient $\nabla V[x, y, z]$.
- EIGHT DoG convolutions. $(V * \mathrm{DoG})[x, y, z]$.
- EIGHT DoG convolutions in gradient space. $(\nabla V * \mathrm{DoG})[x, y, z]$.

This gives 18 features per modality, and thus we have 72 features overall. In addition, because DoG convolutions act as edge (i.e. inflection point) detectors, we can argue that they provide information on a function's second derivative. Thus, the above set of features gives us information on all derivatives from 0th to 3rd order.

Algorithm We begin with an overview of our algorithm. First, we divide our patients into a training and test set. For our program, we feed in all four modalities (T1 pre-contrast, T1 post-contrast, T2 weighted, and FLAIR) of our pre-processed data into feature extraction. Once we extract our features, we will have a feature vector associated with each voxel in our brain. We treat each voxel's feature vectors completely independently. Thus, in the hierarchal neural net training phase of our program, we will not take into account voxel position or neighborhood other than what is already encoded in the feature extraction.

Hierarchal Neural Nets Once we have extracted our features, our learning environment will be a hierarchal neural net. We first train a standard feed-forward neural net (2 hidden layers, 10 neurons per layer) for each of the four tumor subtypes. Neural nets are standard in machine learning, and make predictions based on

learned features (neurons) which are linear combinations $\sum w_i Z_i$ of the inputs Z_i to that neuron. The algorithm iteratively finds the weights w_i, and these neurons are then activated (fired) when they observe inputs parallel to their learned feature directions. NNets are relatively low bias and can model complex nonlinear relationships (the neuron "firing" potential, which is usually a sigmoid, is highly nonlinear) between input features and output. This is appropriate here as we do not expect our tumor classifications to be simple linear combinations of our DoG convolutions. After all four neural nets are trained, we then classify in a cascading fashion as shown in algorithm 1 on the next page.

Essentially, our classification priority in descending order goes: enhancing, necrosis, non-enhancing, and edema. If our neural nets returns positive classifications for multiple tumor subtypes, we classify to the positive subtype with the highest priority. This hierarchal design is based off of the hierarchical majority vote used to combine several different algorithmic results. [9]

This seemingly arbitrary methodology makes perfect sense in the context of our classification problem. Tumor segmentations are judged generally in terms of three accuracies: whole tumor accuracy, tumor core accuracy, and enhancing tumor accuracy. Thus, because they have their own accuracy scores, we must prioritize classification of the core over the non-core (edema), and then also the enhancing core over the other core. The enhancing core generally covers a smaller area of the brain, which lends even more reason to be more sensitive to its detection.

Results are reported as the standard Dice score calculated via 10-fold cross validation (see: beginning of next section). We *do not* use cross validation to select parameters, deciding to keep our neural net parameters set to default values. This is both because the additional computation time would be prohibitive, and also because our Dice scores (which are also calculated from the cross validation) would become biased upwards.

Results and Discussion

For the rest of the paper, we report accuracies as *Dice Coefficients* (also known as the Sørensen-Dice Index). We can describe this index as [12]

$$\text{Dice Score} = \frac{2\,|\text{Pred} \cap \text{Ref}|}{|\text{Pred}| + |\text{Ref}|}, \tag{2}$$

where "Pred" is the voxels that return a positive prediction and "Ref" is the set of voxels which are positive in the ground truth (in our case, the expert segmentation). We can see that in our case, this is also equal to the harmonic mean of the precision (as denoted by $\frac{|\text{Pred} \cap \text{Ref}|}{|\text{Pred}|}$) and the recall (as denoted by $\frac{|\text{Pred} \cap \text{Ref}|}{|\text{Ref}|}$).

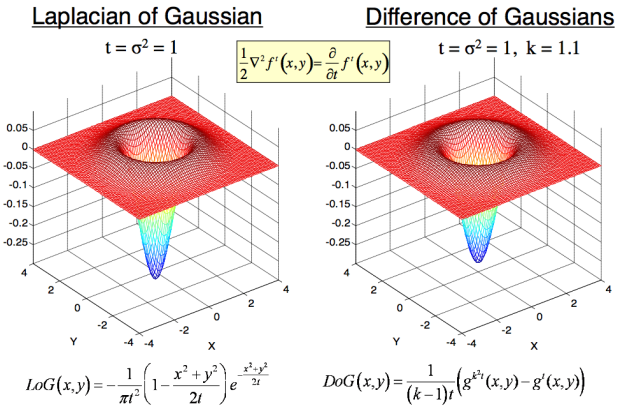


Figure 4: Use of DoG as LoG Proxy

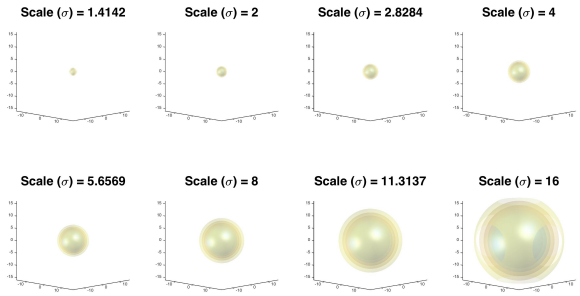


Figure 5: DoG Kernels

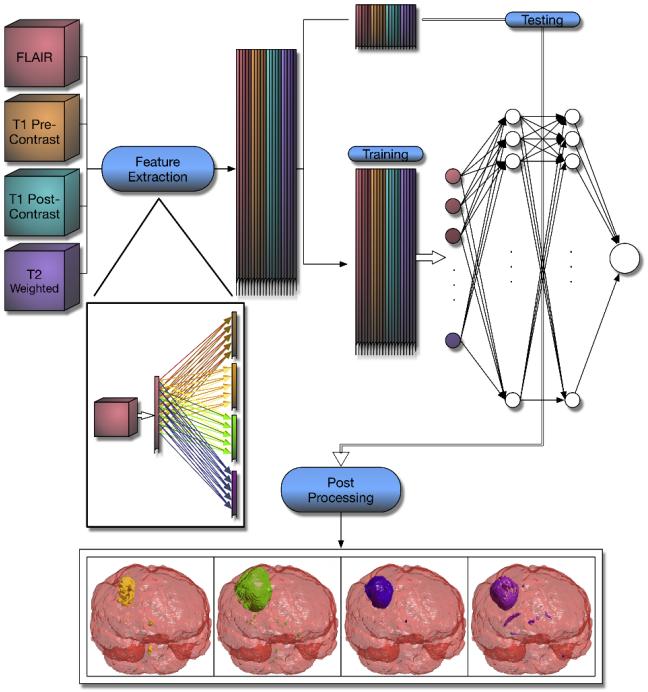


Figure 6: Main Methodology of Learning Pipeline

Algorithm 1 The hierarchical majority vote. The neural net output (between 0 and 1) per voxel for each of the four tumor structures (edema, non-enhancing core, necrotic core, enhancin core) is indicated by p_{edm}, p_{nen}, p_{nec}, p_{enh}, respectively.

```
label ← "nrm"                                  #Initialize Normal Tissue.
if p_edm ≥ 0.5 then (label ← "edm")            #Edema
if p_nen ≥ 0.5 then (label ← "nen")            #Non-Enhancing Core
if p_nec ≥ 0.5 then (label ← "nec")            #Necrotic Core
if p_enh ≥ 0.5 then (label ← "enh")            #Enhancing Core
end
```

We first tried our pipeline (figure 6) with different algorithms (LDA/GDA, Decision Tree, Naive Bayes), and we see that neural nets perform the best. We further note that we only trained on a subset of the data for the results in figure 7, and because neural nets are low bias they also have the most potential to improve with more data. In contrast, the other higher biased methods have already most likely asymptoted to their final performance. Thus, neural nets is the clear choice for training algorithm.

We can see the CV accuracies of segmentation on all 274 patients in figure 8. Our median Dice performance on whole tumor detection is above 90%! To wit, the inter-radiologist repeatability is only 85%, so our accuracy has saturated with respect to the ground truth. One particularly successful segmentation can be seen in 9. A full 3D visualization of our tumor segmentation can be found at `https://youtu.be/kWdE94RvDpQ`. The main draw-back of our program are outliers. More than half of our segmentations are wildly successful, but some segmentations return sub-50% scores, which you would not typically see with a radiologist.

The confusion matrix is:

$$C = \begin{bmatrix} healthy & edm & nec & neh & enh \\ 0.9921 & 0.0002 & 0.0071 & 0.0005 & 0.0001 \\ 0.0585 & 0.6989 & 0.0622 & 0.1637 & 0.0168 \\ 0.1606 & 0.0072 & 0.7772 & 0.0497 & 0.0053 \\ 0.0600 & 0.1600 & 0.3216 & 0.3807 & 0.0777 \\ 0.0210 & 0.0572 & 0.0257 & 0.0396 & 0.8565 \end{bmatrix} \quad (3)$$

The matrix was normalized to rowsums (so the diagonal represents percent true positives for each class). Thus, C_{ij} is the percentage of pixels in class i that were classified as class j. The most confused classes are $edm \mapsto neh$, $neh \mapsto nec$, and $nec \mapsto healthy$. The first two are natural results of our hierarchy (neh overwrites edm, neh overwrites nec), but the last one is more interesting, and can be explained by the fact that nec generally shows up as darker on MR scans, but so does healthy tissue.

Our CV *mean* scores for whole, core, and active tumor detection are 87/76/80, respectively. This is very competitive with previous methods. Some notable ones in 2013 include ones by Zhao and Subbanna which incorporated Markov Random Fields (MRF), achieving Dice accuracies

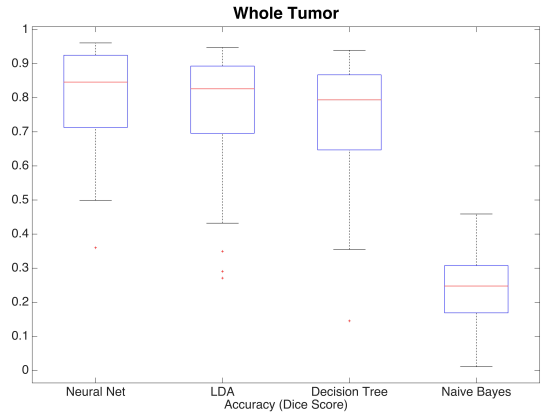
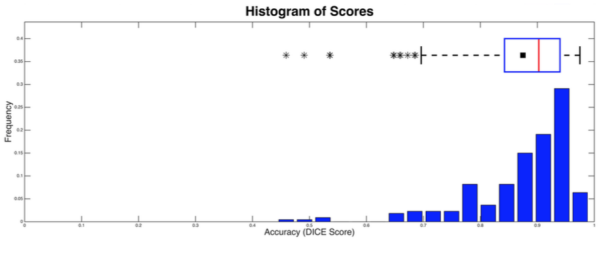
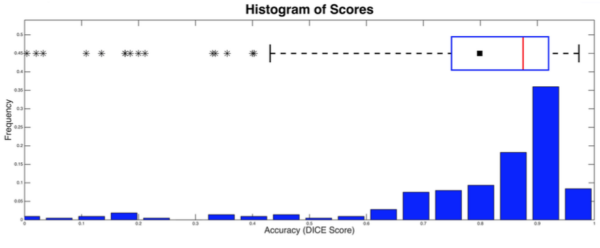


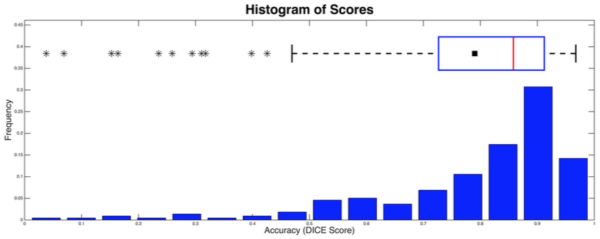
Figure 7: Comparison of Algorithms



(a) Histogram of Total Tumor Scores



(b) Histogram of Active Tumor Scores



(c) Histogram of Core Tumor Scores

Figure 8: Histogram of Dice Score Accuracies

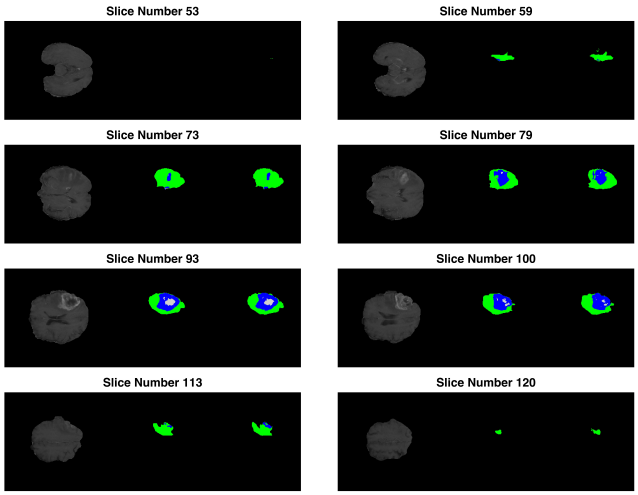


Figure 9: Data Visualization. From Left to Right: T1 post-contrast, our prediction, expert segmentation

of 82/66/49 and 75/70/59, respectively. [9] Festa from 2013 used random forests to achieve a Dice of 62/50/61. [9] In 2014, groups used deep learning and convolution neural nets (CNNs) to achieve accuracies of 85/74/68 (Davy) and 88/83/72 (Urban). [3] * It may come as rather surprising that our methods are competitive with highly complex learning methods such as CNNs. It is worth asking why this is the case.

We begin with a discussion of the bias-variance tradeoff. One important assumption in our data is that voxels are independent, only coupled by information ingrained in their feature vectors. While this is a high bias decision, it allows us to use $n = 4.5$ billion training samples rather than only $n = 274$ patient samples. Contrast this to deep learning algorithms like CNNs, which "learn" the important features from the data by choosing convolution kernels on the inputs that capture the maximal amount of variance of the outputs. [4] However, CNNs use each patient as a single training sample, which allows access to the entire atlas of a 3D MR scan at once and hence the ability for the computer to automatically find complex features which relate wholly different parts of the brain.

We, on the other hand, a priori select DoG convolution filters as our way to relate voxels to their neighborhoods; we are not "learning" the optimal neighborhood information to train on, but instead choose a contrived set of information based on prior knowledge. This injects much bias into our model, but allows us to increase our number of training samples by a factor of more than 15 million. Not only are neural nets intrinsically low bias, the hope is that the improved variance caused by the enlarged sample space will more than compensate for our high-bias assumptions. From the success of our algorithm, it is very plausible that our hopes were not in vain.

However, it is worth mentioning that there may also be biological evidence backing our higher-bias model. It has been known for a long time that neurons in the optical system form receptive fields resembling two concentric circles, with positively firing neurons surrounded by feedback neurons or vice versa. [8] [10] See figure 10 for reference. However, these receptive fields look very much like DoG profiles! Thus, although we are highly biased compared with a deep learning framework, our features may be more successful models of how the human eye perceives information at a low level. The subsequent feed-forward neural net can then learn higher level features from each pixel's lower level biological features. In such a way, we may be more successfully mimicing human biology in this context than higher-level models like deep learning can claim. By no means are we suggesting that our methodology outperforms deep learning in all contexts, but we may in some ways be on the right side of the bias-variance tradeoff, possibly due to the biological underpinnings of our feature space.

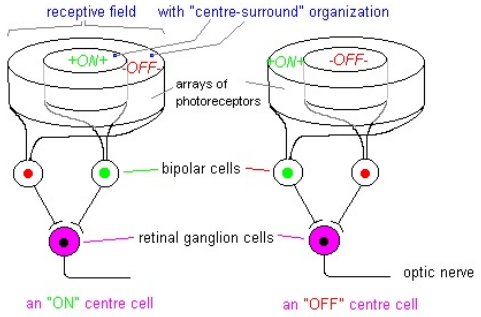


Figure 10: Biological Receptive Fields

Figure taken from Tim Jacob. [5]

Conclusions We have thus shown that a hierarchical neural net model performs remarkably well on the glioblastoma segmentation problem. Our segmentation results are competitive with those using much more complex methods, and we argue our success is due to our smart choice of features along with a greatly enlarged sample space and flexible training method (neural nets). Our algorithm is powerful despite its relatively high bias, and we hope that it may serve the medical community in their work.

The natural next step of this project is a thorough analysis of our model's asymptotics. We have claimed that our large data set has significantly reduced model variance, but it is unknown whether we can further reduce variance with more data. Given that our segmentation algorithm is already on par with our reference expert segmentations, we suspect but would like to confirm that our model has already reached its large-data asymptotic performance.

*Many of these other algorithms were trained on only a subset of 30 patients in line with BraTS Challenge rules. However, our algorithm returns Dice Scores of 89/78/71 on the same subset, not appreciably different from our results on the full 274 patients.

References

[1] Bauer, S., *et. al.* (2012). "A skull-stripping filter for ITK." *Insight Journal.*

[2] Bleeker, F. E., *et. al.* (2012). "Recent advances in the molecular understanding of glioblastoma." *Journal of Neuro-Oncology* **108** (1): 11-27.

[3] "BraTS Challenge Manuscripts." (2014) *MICCAI 2014.* Harvard Medical School, Boston, Massachusetts.

[4] Hinton, Geoffrey. (2014). "Neural Networks for Machine Learning." *Coursera*

[5] Jacob, Tim. (2003). "Vision" *Cardiff University*

[6] Lawrence, Steve. (1997). "Face Recognition: A Convolutional Neural-Network Approach." *IEEE Transactions on Neural Networks* **8** (1).

[7] Lowe, D. G. (1999). "Object recognition from local scale-invariant features." *Proceedings of the International Conference on Computer Vision* **2**. pp. 1150-1157.

[8] Martin, John H. (1989). "Neuroanatomy: Text and Atlas." 4ed.

[9] Menze, B. H. (2013). "The Multimodal Brain Tumor Image Segmentation Benchmark (BRATS)." *IEEE Transactions on Medical Imaging 2014.*

[10] Ng, E.Y.K., et. al. (2012). "Human Eye Imaging and Modeling."

[11] Olga Russakovsky*, Jia Deng*, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg and Li Fei-Fei. (* = equal contribution) ImageNet Large Scale Visual Recognition Challenge. IJCV, 2015.

[12] Sørensen, T. (1948). "A method of establishing groups of equal amplitude in plant sociology based on similarity of species and its application to analyses of the vegetation on Danish commons." *Kongelige Danske Videnskabernes Selskab* **5** (4): 1-34.