Application of Deep Learning (CNN) for Classification of Brain Tumors

By

Chenchal Subraveti

Capstone Project

Submitted to the Panel of Reviewers

Applied Data Science Bootcamp

MIT IDSS - ADSB

Great Learning

January 2021
Nashville, Tennessee

Summary

- Neuroradiologic analysis of MRI (Magnetic Resonance Imaging) is the primary method of diagnosing human brain tumors.
- In this project a Convolutional Neural Network (CNN) was built which classifies MRI images into glioma, meningioma, pituitary tumors or no-tumors.
- The sensitivities were 97% for no-tumor, 87% for meningioma, 73% for pituitary tumor and 31% for glioma with a prediction accuracy of 73%.
- The response time for each test image prediction as about 15 msec compared to a best case of 20 min/MRI by skilled neuroradiologist.

Motivation and Importance (*Health*)

- Cancer is the leading cause of death in the world
- People diagnosed with brain tumor have relative 5 year survival of 32.6%
- Rapid accurate diagnosis is crucial for better prognosis and treatment
- Radiologic assessment is the main diagnostic tool to even locate a tumor for eventual biopsy and diagnosis confirmation
- Availability of skilled neuroradiologist is a limitation in diagnosis

Motivation and Importance (*ROI*)

- A neuroradiologist can examine about 3 scans/hour at approximately $168/hr based on median compensation of $350K/year [https://pubmed.ncbi.nlm.nih.gov/29929936/]
- AI systems can scan more than 700 scans/hr (5 sec/scan) to classify brain MRIs
- While a neuroradiologists' services are very much needed, having a tool that flags scans for brain tumors will increase the *efficiency* as well as control for *mis-diagnosis* or even a *missed diagnosis*
- This increased efficiency would translate to higher *return-of-investment* for investing in development of such an artificial intelligence tool

Project Goals

- Develop an artificial intelligence system based on Convolutional Neural Networks (CNN) to rapidly predict the type of brain tumor in an MRI scan of head
- The system will flag brain MRIs for *glioma, meningioma, pituitary or no-tumor*
- The system could be used for *clinical decision support* as well as to control for *mis-diagnosis* or even a *missed diagnosis*
- The system will use *precision, recall,* and *accuracy* metrics for reporting

Exploratory Data Analysis

1A) Categories

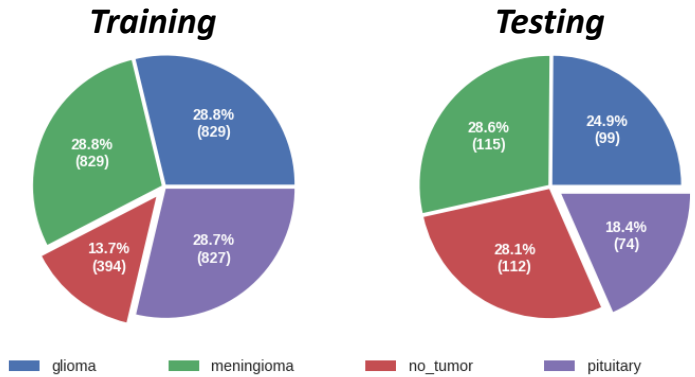


Figure 1A.

- Training and testing datasets have 2881 and 402 MRI images respectively
- Imbalance in the distribution of tumor and non-tumor categories
  - ***fewer no_tumor*** images in training (13.7%) compared to testing (28.1%)
  - ***fewer pituitary_tumor*** images in testing (18.4%) compared to training (28.7%)

1B) Aspect ratio

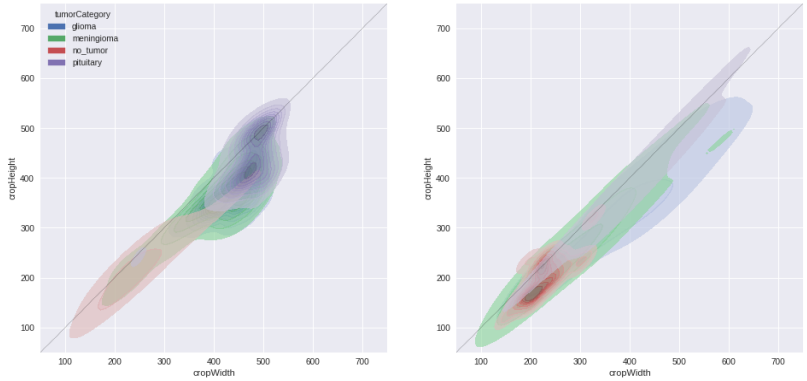


Figure 1B.

- Wide variation in aspect ratio of both training and testing datasets
- Variation among different tumor categories
- Example *glioma* class is square aspect, while others vary 100 pixels to more than 750 pixels in both width and height

1C) Pixel values

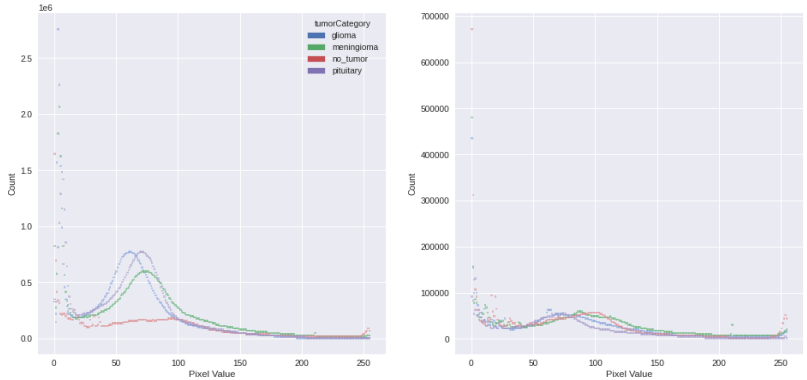
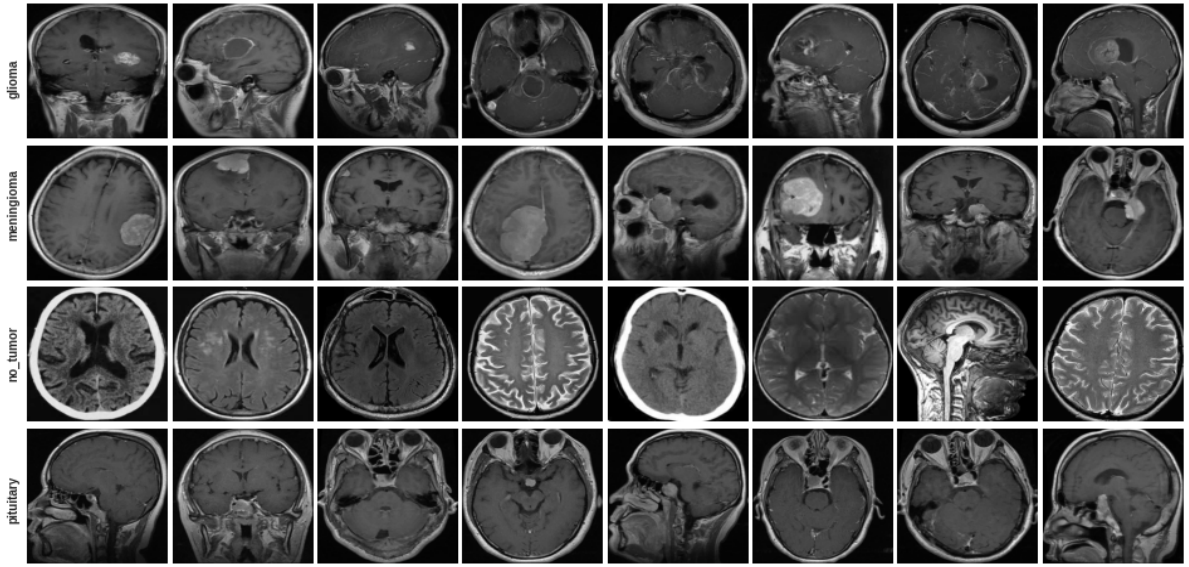


Figure 1C.

- Wide variation in grayscale values across both training and testing
- Variation among different tumor categories within each set
- Training set has a vey large proportion of *dark* or low-value pixels

Exploratory Data Analysis

2A) Training Images



2B) Testing Images

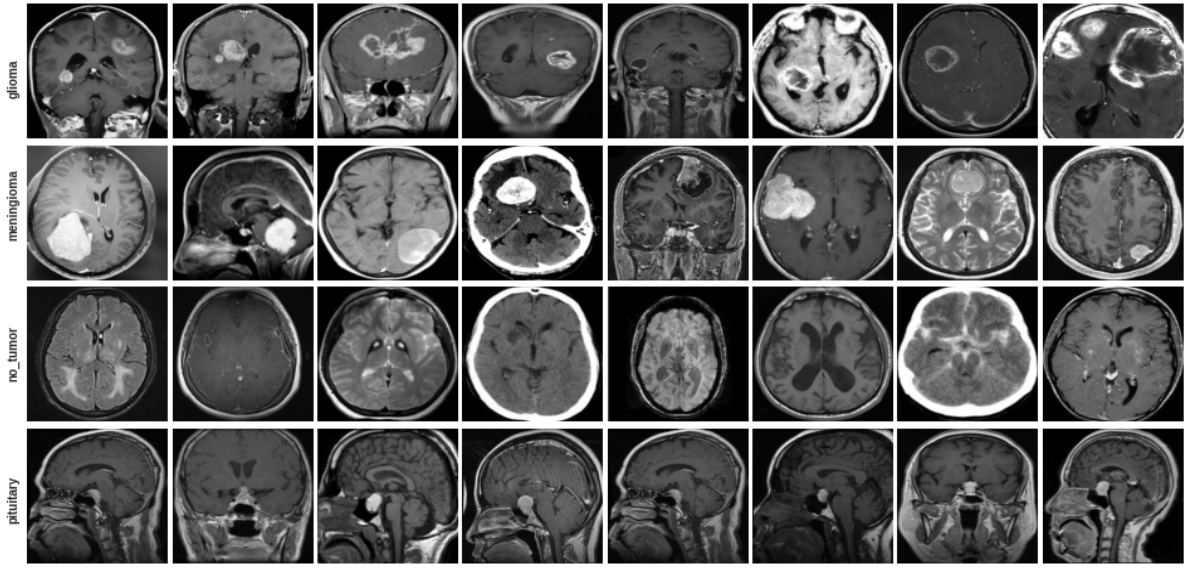
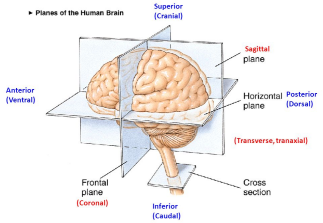
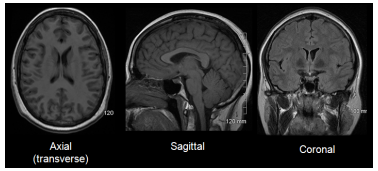


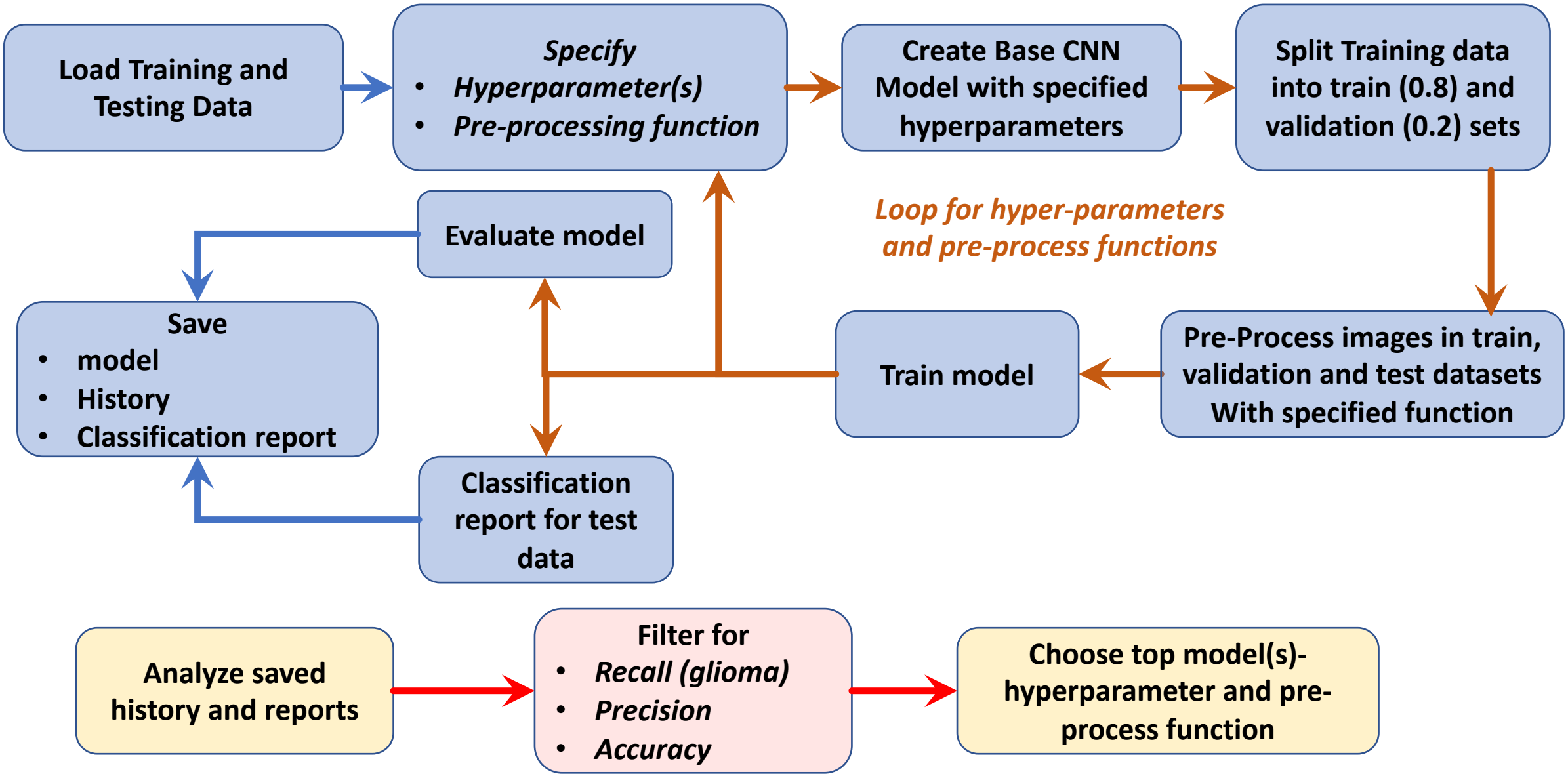
Figure 2A & 2B

- Variation in following can be seen in both training and testing datasets:
  - Contrast
  - Brightness
  - Rotations
  - Slice orientation – axial, sagittal, coronal
- Tumors in *testing* images appear more prominent than in training images
- No bounding boxes for the tumors





Model training flowchart



Data Preparation – Pre-Processing

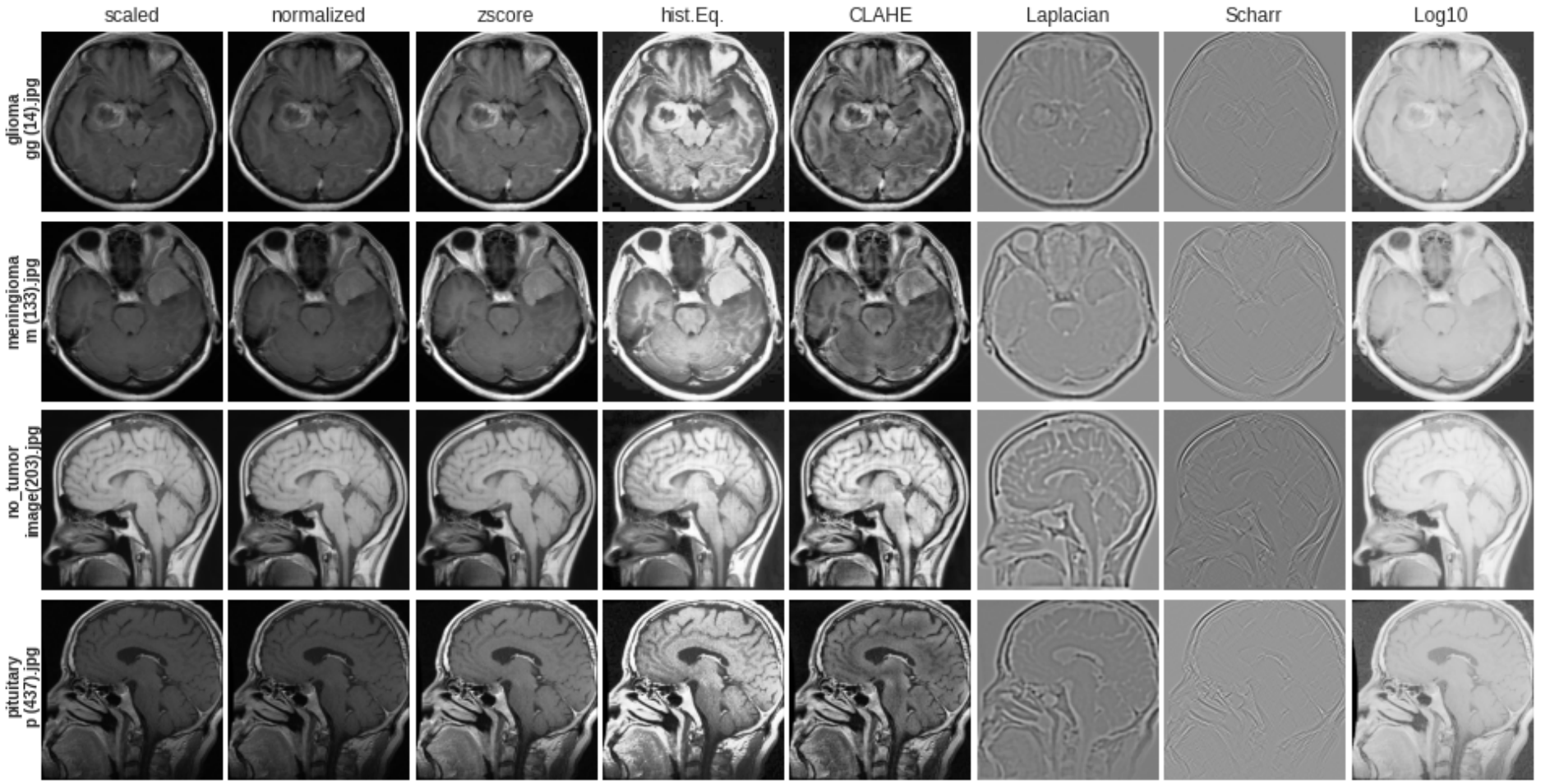


Fig. 3) Pre-processed training images

- **Raw**: No processing.
- **Scaled**: new_img = image/255.0
- **Normalize**: new_img = (image-min(image))/range(image)
- **Z-Score**: new_img = (image-mean(image))/std(image)
- **Histogram Equalize**: new_img = cv2.equalizeHist(im)
- **CLAHE**: new_img = cv2.createCLAHE().apply(im)
- **Laplacian**: new_img = cv2.Laplacian().apply(im
- **Scharr:** new_img = cv2.Scharr(cv2.Scharr(im))
- **Log10:** new_img = np.log10(im+1)

Model – tuning and training

- Several model hyper-parameters along with different image processing functions were evaluated
- Analysis of results (see *supplementary materials)* from the following are used in deciding the final model:
  - Pre-processing functions
  - Hyper parameters – Batch normalization, kernel regularization, dropout
  - Category imbalance and class weights
  - Loss functions (categorical-crossentropy, kl-divergence)
  - Metrics (accuracy, recall, and precision)
  - Transfer learning for Vgg16, Xception, InceptionResNetV2, and NasNetMobile
  - Data augmentation
- Best models were selected based on both accuracy and specifically for *glioma recall as well as response time*

	epoch	loss	accuracy	precision	recall	val_loss	val_accuracy	val_precision	val_recall	fileId	modelName	imgPreProc
0	0	1.015708	0.556858	0.710744	0.335938	0.873579	0.589255	0.640860	0.516464	final_06_BaseModel-ZScore_history.csv	BaseModel	ZScore
1	1	0.606517	0.756076	0.791479	0.685330	0.543726	0.795494	0.824092	0.746967	final_06_BaseModel-ZScore_history.csv	BaseModel	ZScore
2	2	0.379654	0.858073	0.876538	0.835069	0.440654	0.821490	0.835443	0.800693	final_06_BaseModel-ZScore_history.csv	BaseModel	ZScore
3	3	0.263037	0.907552	0.915892	0.898003	0.414595	0.845754	0.856128	0.835355	final_06_BaseModel-ZScore_history.csv	BaseModel	ZScore
4	4	0.143409	0.955729	0.960509	0.950087	0.398939	0.854419	0.863475	0.844021	final_06_BaseModel-ZScore_history.csv	BaseModel	ZScore

Final Model Evaluation

Created final models for recommendation

- Train top 3 models with cross validation for recommendation.
- The choice of model and pre-processing are based on *giloma recall* and overall accuracy highlighted in table below.
- All models used ***categorical_crossentropy*** minimization although *KL-divergence* was very slightly beneficial (1 percentage point) for Laplacian filter.
- All models were evaluated on *accuracy* during training but also assessed for *precision* and *recall*. We filtered models for recall.
- Since the goal of the project is to improve the ability of the model to correctly identify patients with *glioma,* we wanted to minimize the *false-negatives*, we focused the efforts on maximizing the ***recall*** or ***sensitivity*** of the model.
- Early stopping is based on *accuracy* metric gain of 0.05 percentage points for 8 epoch run. Compared to recall or precision accuracy simultaneously improved both recall and precision of the model.

Final model (batch normalization and regularization as needed

Layer	Name	Properties
1	Input Layer	(256, 256, 1)
2	Conv 2D	filters=32, kernel size = [3,3]
3	LeakyReLU	alpha=0.1
4	Max Pooling	pool_size=2
5	Conv 2D	filters=64, kernel size = [3,3]
6	LeakyReLU	alpha=0.1
7	Max Pooling	pool_size=2
8	Conv 2D	filters=64, kernel size = [3,3]
9	LeakyReLU	alpha=0.1
10	Max Pooling	pool_size=2
11	Conv 2D	filters=32, kernel size = [5,5]
12	LeakyReLU	alpha=0.1
13	Max Pooling	pool_size=2
14	Conv 2D	filters=16, kernel size = [5,5]
15	LeakyReLU	alpha=0.1
16	Max Pooling	pool_size=2
18	Flatten	
19	Dense	filters=512
20	LeakyReLU	alpha=0.1
21	Dense	filters=256
22	LeakyReLU	alpha=0.1
23	Dense	filters=4
22	Softmax	
23	Output	4 classes (0=glioma, 1=meningioma, 2=no_tumor, 3=pituitary)

Final Models Recommendation

- The top-3 models performance varied between 0.25 to 0.32 for glioma recall
- Model that used both batch normalization and regularization had highest accuracy when used with images that were filtered with a Laplacian
- This study recommends the ***CNN model with batch normalization and regularization on raw images*** since it had ***higher sensitivity (recall) for glioma***.
- Metrics used for recommendations
  - Accuracy = (TP + TN)/(TP + TN + FP + FN)
  - ****Recall (sensitivity) = TP/(TP + FN)***
  - Precision = TP/(TP + FP)
- Time taken to predict for 402 test images = 6.23s, Response time = 0.015s/image

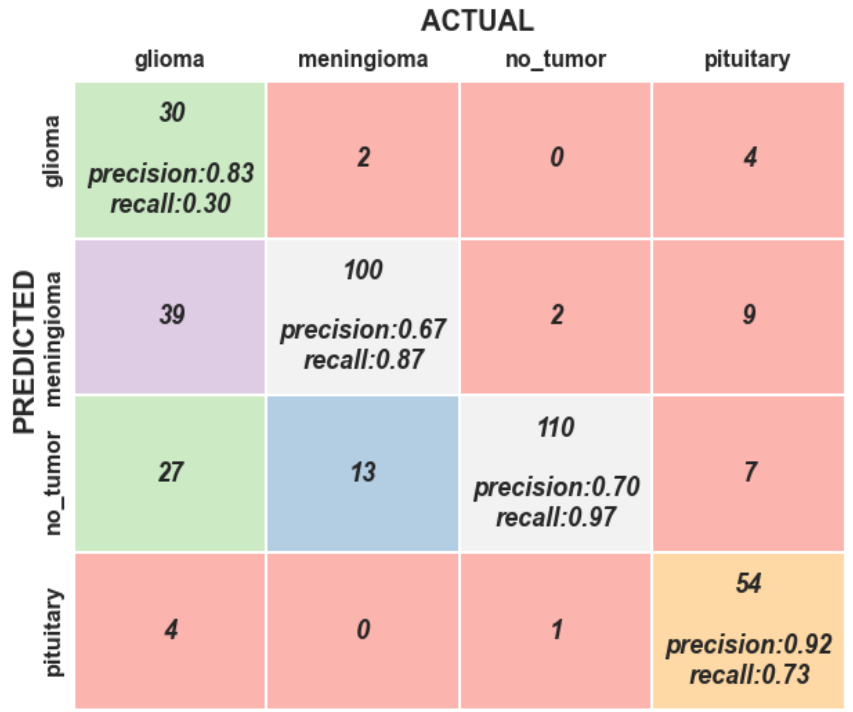
4A) Performance of top 3 models

rank	modelName	imgPreProc	glioma	meningioma	no_tumor	pituitary	accuracy
1	BaseModel-BatchNorm-L2Regularize	Laplacian	0.37	0.87	0.98	0.78	0.76
2	BaseModel-BatchNorm-L2Regularize	Raw	0.37	0.87	0.97	0.80	0.76
3	BaseModel-BatchNorm	ZScore	0.35	0.88	1.00	0.68	0.74

4B) Final recommended model

rank	modelName	imgPreProc	glioma	meningioma	no_tumor	pituitary	accuracy
1	BaseModel-BatchNorm-L2Regularize	Raw	0.31	0.87	0.97	0.73	0.73

4C) Performance of the recommended model



Future directions

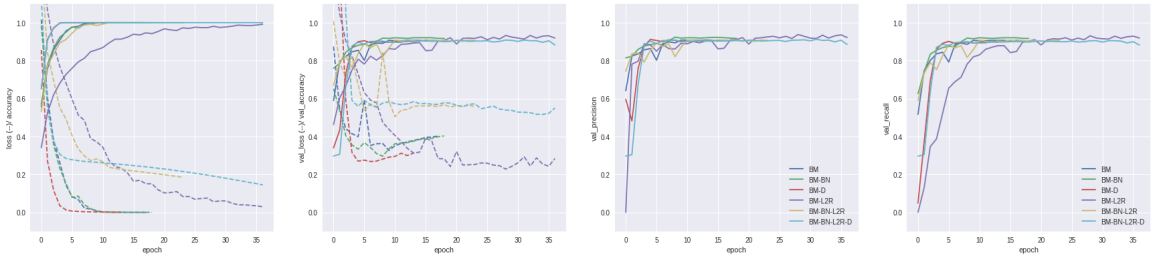
- Analyze mis-classified images to gain insights into the nature of the problem
- Get a better understanding of the variations in images due to ***source*** by checking image associated meta-data including slice orientation (axial, sagittal, and coronal).
- The model will learn to differentiate local features better if we could get ground-truth bounding boxes for the location of tumor.
- Model stacking to explore building multiple models for different categories could also help in improving recall and high precision for all classes
- Finally their availability of good segmentation data would go a long way in training and building a more robust model.

Supplementary material

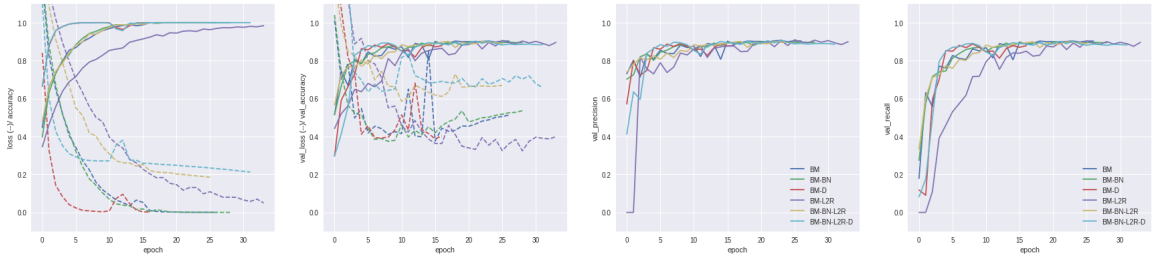
Top 10 model and image pre-processing combination

	modelName	imgPreProc	glioma	meningioma	no_tumor	pituitary	accuracy
89	BaseModel-BatchNorm-L2Regularize	Laplacian	0.37	0.869565	0.982301	0.783784	0.761194
161	BaseModel-BatchNorm-L2Regularize	Raw	0.37	0.869565	0.973451	0.797297	0.761194
5	BaseModel-BatchNorm	ZScore	0.35	0.878261	1.000000	0.675676	0.743781
145	BaseModel	Raw	0.35	0.826087	0.973451	0.770270	0.738806
149	BaseModel-BatchNorm	Raw	0.34	0.869565	0.973451	0.743243	0.743781
125	BaseModel-BatchNorm	Normalize	0.33	0.852174	0.973451	0.635135	0.716418
29	BaseModel-BatchNorm	HistEqual	0.32	0.886957	0.982301	0.864865	0.768657
17	BaseModel-BatchNorm-L2Regularize	ZScore	0.32	0.878261	1.000000	0.797297	0.758706
1	BaseModel	ZScore	0.32	0.904348	0.982301	0.662162	0.736318
121	BaseModel	Normalize	0.32	0.852174	0.955752	0.662162	0.713930

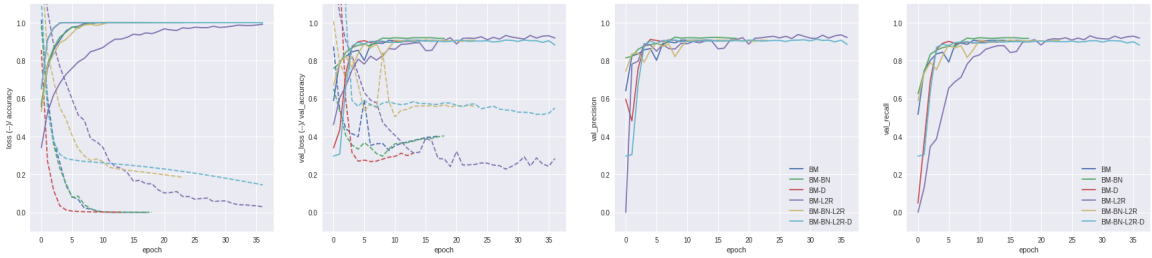
Plot of evaluation metrics for image processing function ZScore



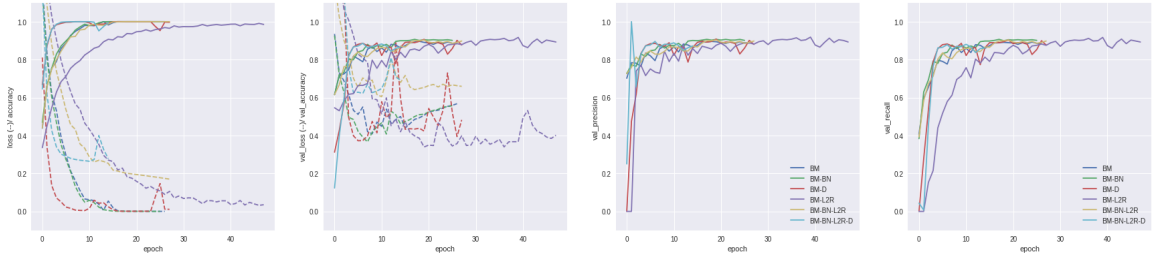
Plot of evaluation metrics for image processing function Scaled



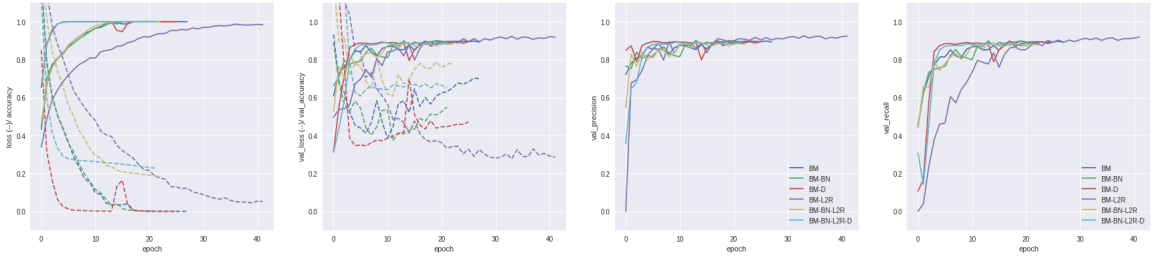
Plot of evaluation metrics for image processing function ZScore



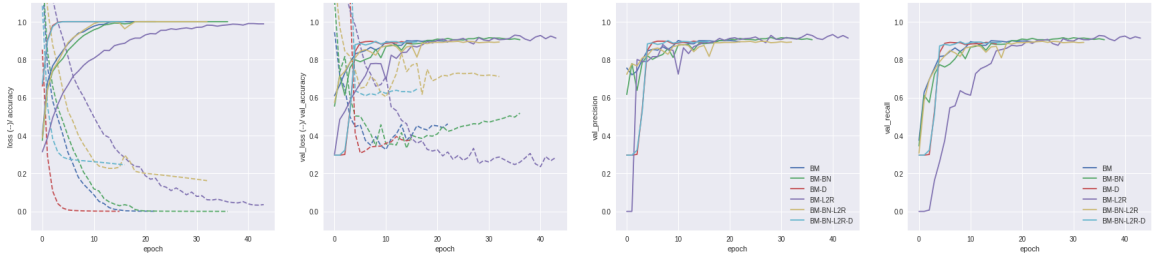
Plot of evaluation metrics for image processing function Normalize



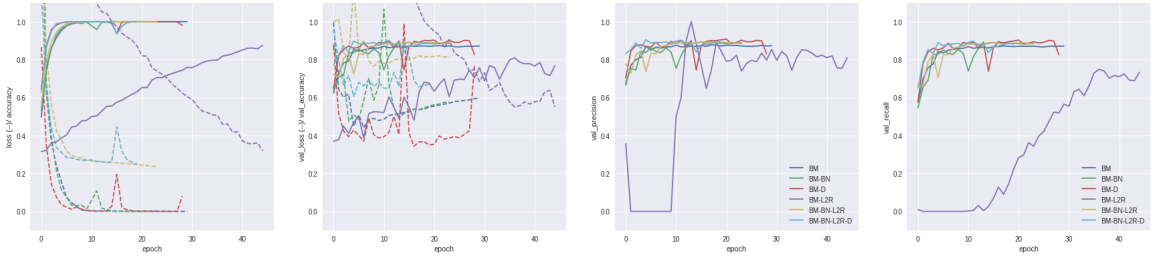
Plot of evaluation metrics for image processing function CLAHE



Plot of evaluation metrics for image processing function HistEqual



Plot of evaluation metrics for image processing function Raw



Supplementary material

Insight for hyperparameter tuning and Image-Processing combinations

- The above plots show that the Base model with batch normalization (BM-BN) achieved most *stable* reaults and saturates without large oscillations for Z-scored, normalized and raw images. Histogram equalized images with batch normalization achieved a higher accuracy for a slightly loweer glioma recall.
- **Pre-processing with Laplacian filter, z-scored and raw images combined with batch normalization and regularization gave the best performance for comparable accuracy**.
- Most other image pre-orocessing were not as useful in improving recall for glioma tumor.

Insight for choice of loss functions, metrics

- Of the different loss functions
  - for ***Z-Scored images*** `kl-divergence`, which is a measure of similarity or relative-entropy between the distributions of true-labels and predicted-labels, gives best ***recall*** for glioma, as well as other tumor categories
  - for ***Laplacian images*** `categorical-crossentropy`, which is a measure of total entropy between distributions of true-labels and predicted-labels, gives best ***recall*** for glioma, as well as other tumor categories

image-preProcess	loss-function	glioma	meningioma	no_tumor	pituitary	accuracy
Laplacian	**categorical-crossentropy**	**0.3600**	**0.4261**	**0.8142**	**0.5946**	**0.5498**
Z-Score	**kl-divergence**	**0.2700**	**0.7739**	**0.5841**	**0.5405**	**0.5522**
Z-Score	categorical-crossentropy	0.1200	0.2870	0.8230	0.4054	0.4179
Z-Score	mean-squared-error	0.06000	0.6261	0.5310	0.5676	0.4478

Insight using Program suggested weights for training

- After training with suggested weights `class_weight= {0: 20, 1: 4, 2: 1, 3: 1.5}`, the recall did not show any improvements from base model with no weights for clioma recall.

Smote operation	glioma	meningioma	no_tumor	pituitary	accuracy
weights NOSMOTE	0.3300	0.9130	0.9469	0.5811	0.7164
no weights no SMOTE	**0.3800**	**0.8609**	**0.9558**	**0.6081**	**0.7213**

Supplementary material

Insight for fixing dataset imbalance for tumor categories and performance

- We have an uneven distribution of tumor categories in the training dataset. In order to generate a balanced distribution of tumor categories, `imblearn.over_sampling.SMOTE()` was used. After `SMOTE` call the distribution of classes were as follows, although, the number of samples for categories are duplicated to get balanced distribution:

Tumor Category	train	trainFraction	trainSMOTE	trainSMOTEFraction
glioma	829	0.288	830	0.25
meningioma	830	0.288	830	0.25
no_tumor	**395**	**0.137**	**830**	**0.25**
pituitary	827	0.287	830	0.25
Total	2881	1.000	3320	1.00

- Using ***Z-Scored images*** with `categorical-crossentropy` and `L2-Regularization`, on `BaseModel` we observed a small ***fall** in the ***recall*** for glioma, suggesting that balancing the tumor categories resulted in a *lowered* or no gain in improving ***recall*** for glioma.
- with SMOTE, while glioma recall was lower, recall for memingioma, no_tumor, and pituitary were are reatively higher. Accuray was also higher when using SMOTE.

Smote operation	glioma	meningioma	no_tumor	pituitary	accuracy
SMOTE	0.3500	0.9304	0.9823	0.6757	0.7537
no SMOTE	**0.3800**	**0.8609**	**0.9558**	**0.6081**	**0.7213**

Insights for experiments with Transfer learning using models pre-trained on *imagenet*

- Transfer learning for **NasNetMobile** gave the over best result for *glioma recall* followed by **InceptionResNetV2**, eventhough these architectures were primarily trained on ImageNet data. Only the top layers of these nets were trained during transfer learning.

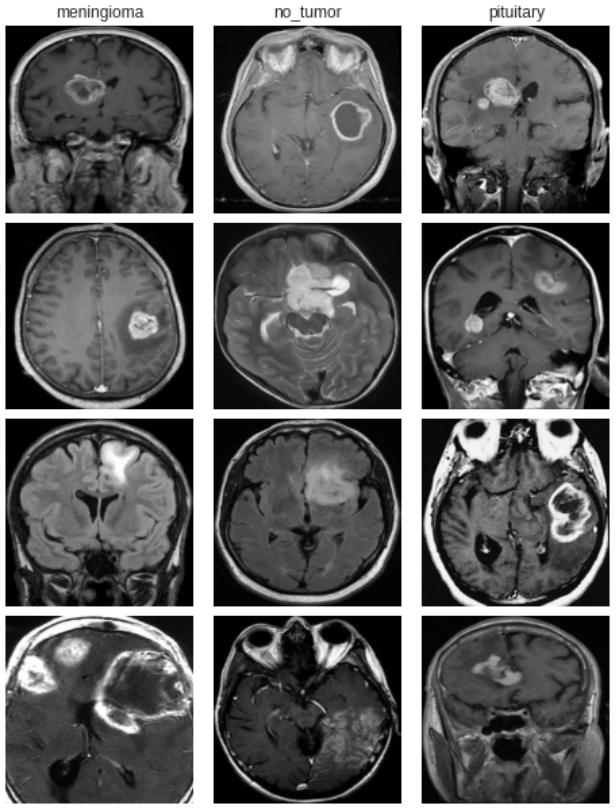
CNN Model	glioma	meningioma	no_tumor	pituitary	accuracy	time(s)
VGG16	0.3400	0.8348	0.8761	0.8378	0.7239	148.79
Xception	0.3200	0.9043	0.9823	0.5946	0.7239	185.61
InceptionResNetV2	0.3800	0.8696	0.9204	0.7432	0.7388	647.40
NasNetMobile	**0.4100**	**0.8609**	**0.8584**	**0.6892**	**0.7164**	141.02

Supplementary material

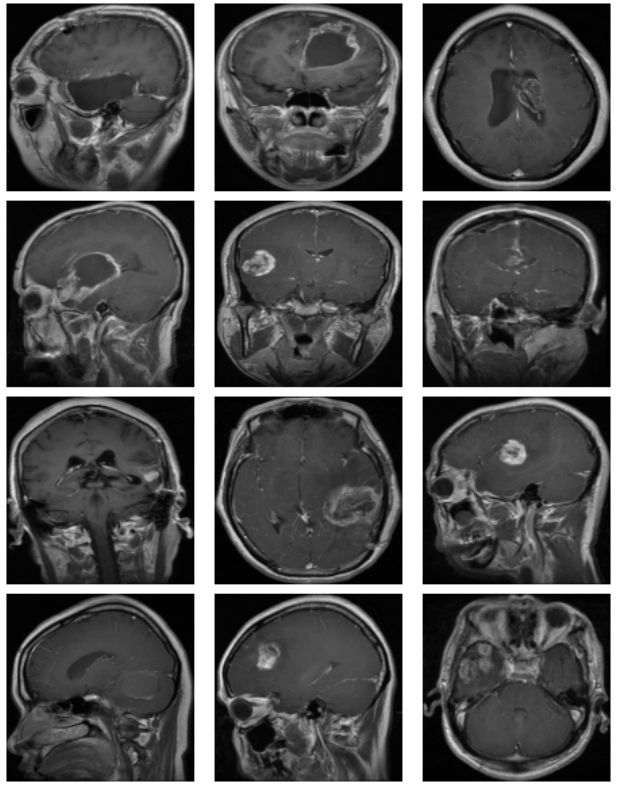
Insight for if *data augmentation* improve performance on tumor classification

- Data augmentation did not run on TPU, which maybe due to the the inaccessibilty of the flow iterator to the running TPU processes.
- Although, the recall for glioma is higher, the traces for training data for validation set show a very unstable behavior of the system when using data augmentation. The overall accuracy of the system is is low around 54%
- Data augmentation also has the disadvantage of not knowing the *on the fly* images producuced by the iterator during training. One way to overcome this is to first create the iterator and save the resulting augmented images, either in memory or to file(s) which comes with its own overhead for the *ROI*, hence data augmentation was not used in other model trainng and evaluations.

Mis-classified Glioma tumors

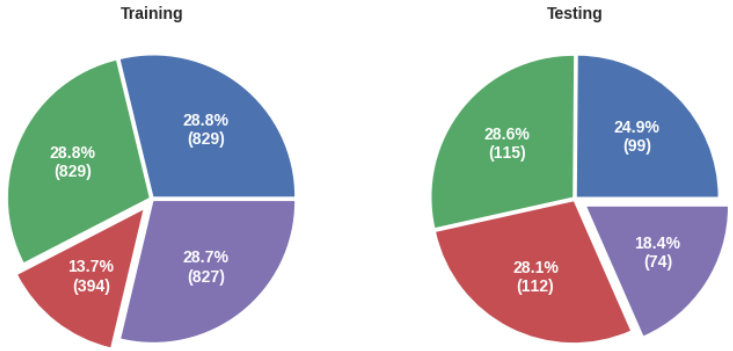


Training Glioma tumors

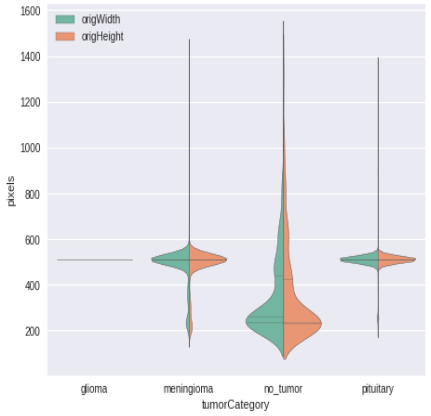


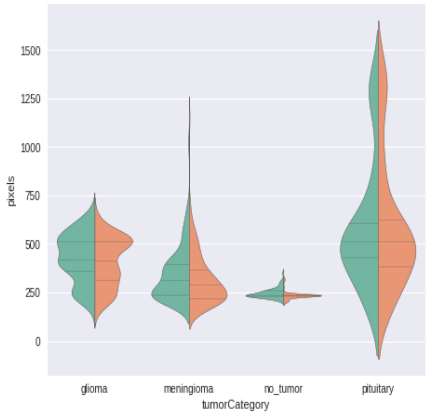
Supplementary material

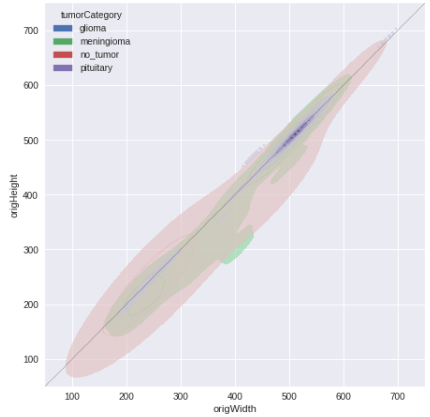
Original Dataset: EDA

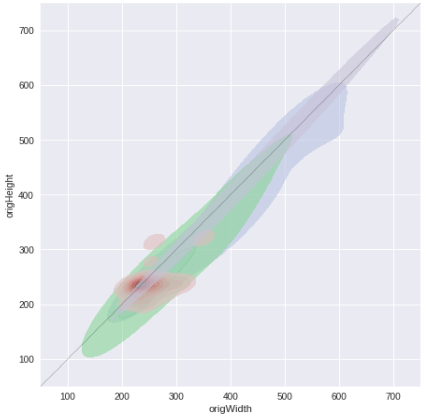


Original size

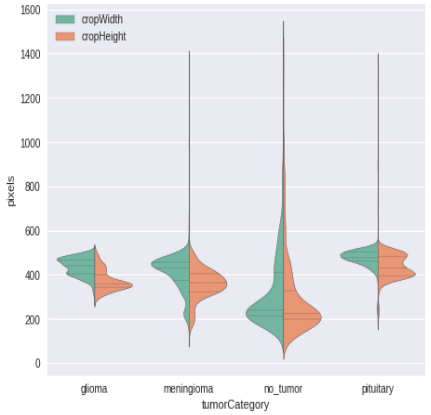


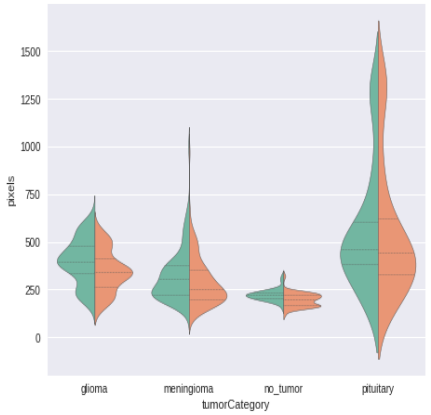


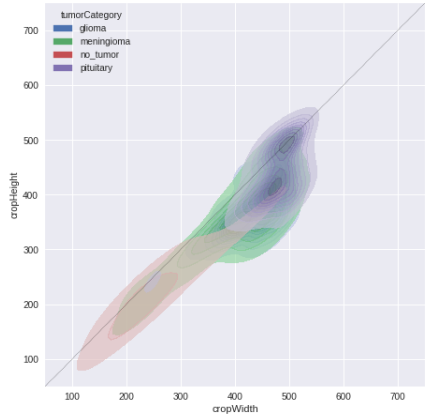


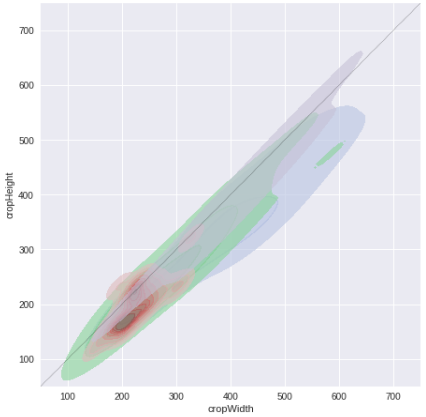


Cropped & resized



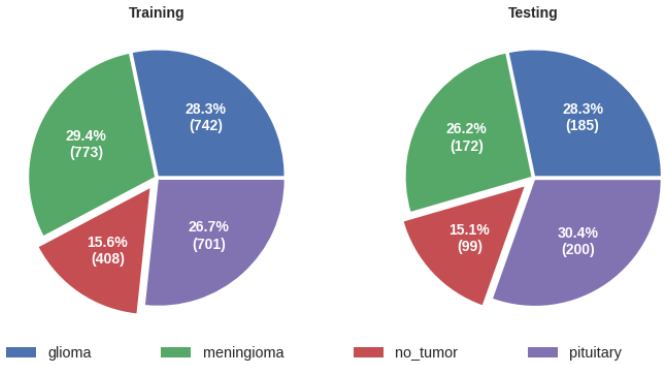




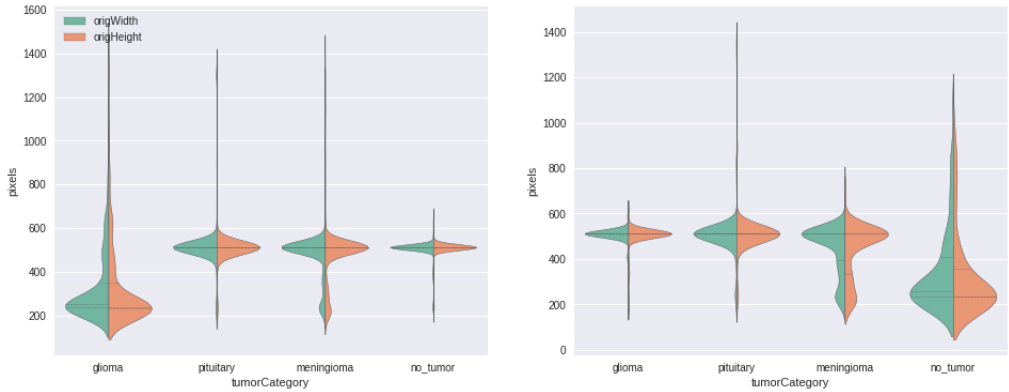


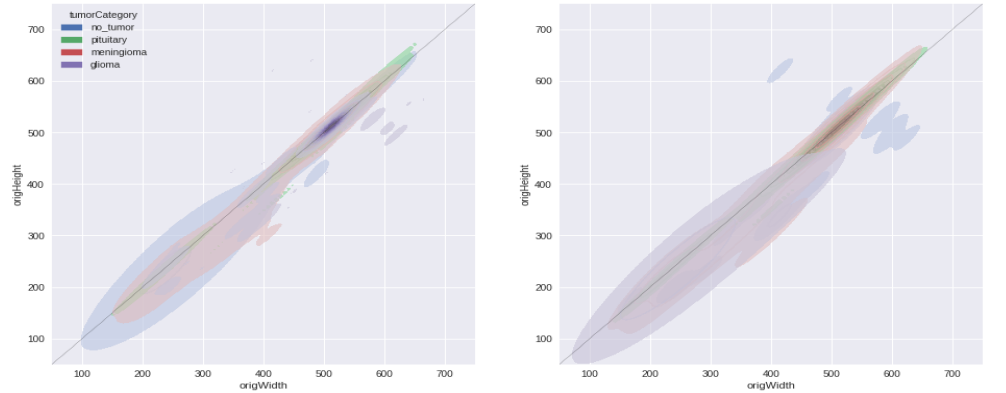
Supplementary material

Merged and repartitioned Dataset: EDA

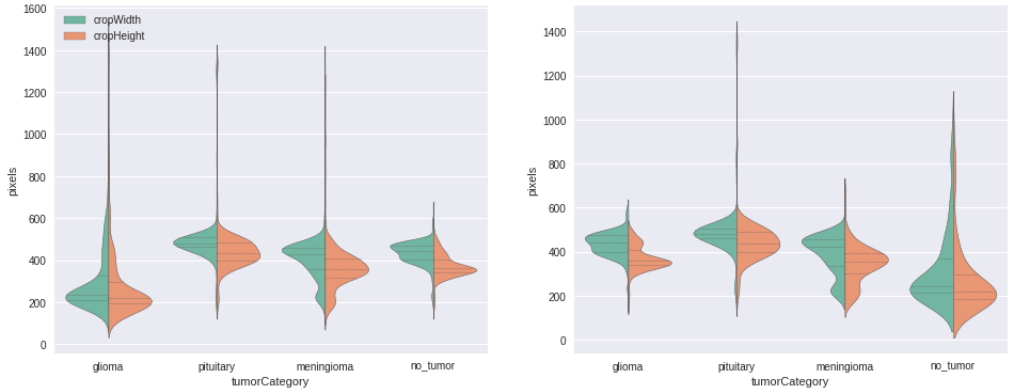


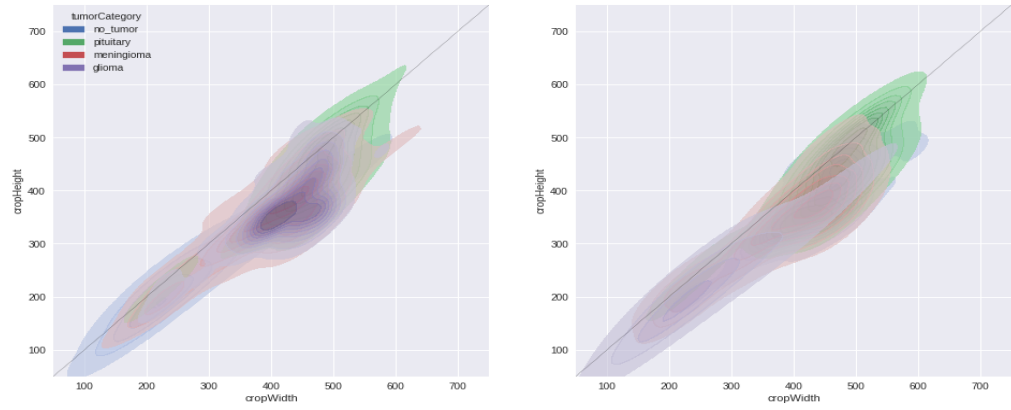
Original size





Cropped & resized





Supplementary material

Original Dataset

Top 3 model and image pre-processing combination **ORIGINAL DATASETS**

	modelName	imgPreProc	glioma	meningioma	no_tumor	pituitary	accuracy
89	BaseModel-BatchNorm-L2Regularize	Laplacian	0.37	0.869565	0.982301	0.783784	0.761194
161	BaseModel-BatchNorm-L2Regularize	Raw	0.37	0.869565	0.973451	0.797297	0.761194
5	BaseModel-BatchNorm	ZScore	0.35	0.878261	1.000000	0.675676	0.743781

Merged and repartitioned Dataset

Top 3 model and image pre-processing combination **AFTER MERGING and SPLITTING DATASETS**

	modelName	imgPreProc	glioma	meningioma	no_tumor	pituitary	accuracy
41	BaseModel-BatchNorm-L2Regularize	ZScore	0.41	0.947826	0.973451	0.513514	0.741294
17	BaseModel-BatchNorm-L2Regularize	Raw	0.38	0.930435	0.964602	0.729730	0.766169
161	BaseModel-BatchNorm-L2Regularize	Laplacian	0.38	0.930435	0.973451	0.662162	0.756219