Brain Tumor Classification

Project Report

Team Members: Pramit Vyas, Ajay Tumu, Moulica Goli

Abstract

In the realm of machine learning, our focus revolves around the intricate world of healthcare machine learning tasks, specifically brain tumor recognization. This project delves into the fascinating domain where we can identify if a brain has a tumor without performing an invasive procedure, we can tell this with just a scan. We have 4 types of brain tumor whether they be a pituitary brain tumor, a meningioma brain tumor, a glioma brain tumor or if the scan has no tumor. The significance of accurate brain tumor detection extends beyond just machine learning, as it can save lives by identifying malignant activity on one's brain much quicker than past methods, helping start treatment much quicker. To accomplish this task we used a variety of techniques including Convolutional Neural Networks, and transfer learning with ResNet and VGG. Our transfer learning didn't do as well due to long training time which caused us to only use 10 epochs. We were able to accomplish this with an accuracy of 90 percent with a simple convolutional neural network model with 3 convolution and 3 max pooling layers with a dropout rate of .2. Then we were able to reach a 96 percent with a model which had 2 convolution and max pooling layers in a block that got repeated 3 times followed by a fully connected layer flattened. This report delves into the exploration, interpretation, and classification of brain tumor images using deep learning techniques such as a Convolutional Neural Network. Our journey aims to contribute to the ever-evolving landscape of health AI, which can be used to revolutionize how we have been treating brain tumors. In addition to the technical aspects, this project underscores the profound impact of machine learning in healthcare, particularly in the realm of brain tumor recognition. By harnessing the power of convolutional neural networks and transfer learning, we aim to not only improve diagnostic accuracy but also expedite the initiation of life-saving treatments. Through our endeavors, we aspire to contribute to the ongoing evolution of health AI, ultimately reshaping the landscape of brain tumor management and enhancing patient outcomes on a global scale.

1 Introduction

Brain tumors (BT) pose a significant threat to individuals worldwide, impacting countless lives across the globe. Their critical nature stems from their potential to disrupt normal brain function, making them one of the most dangerous forms of brain cancer. While statistically less common compared to other cancers, brain tumors still affect a significant portion of the population.

This introductory section outlines the different types of brain tumors – Meningioma, Glioma, Pituitary – and highlights their varying prevalence within the broader spectrum of brain tumors. We further emphasize the long-lasting and complex consequences brain tumors can have on patients, both physically and psychologically.

To address the crucial need for early diagnosis and effective treatment, researchers and medical experts within the field of Internet of Things (IoT) healthcare have actively explored various non-invasive techniques for brain tumor classification and cancer identification. Our focus in subsequent sections will delve into the application of Machine Learning (ML) and Deep Learning (DL) models in designing computer-aided diagnostic systems for brain cancer detection.

Using Convolutional Neural Network(CNN) to build a model that classifies types of brain tumours. Starting with the baseline model and improving the model using different parameters and increasing the layers in the model. The data-set we are using is Kaggle Brain Tumor MRI Dataset .The task is to categorize each type of tumor based on the given MRI of the brain and the dataset has 4 categories No-tumor,Meningioma, Glioma, Pituitary . The expected outcome of the project is to build an accurate model for classifying the given MRI into one of the four categories. The applications are broad ranging Healthcare and diagnostic devices.

2 Related Work

The Scientific Report published a paper which tried to classify brain tumors using a convolutional neural network function. They first classified a brain image as a tumor or not a tumor and if it is a tumor, it classified what type of tumor it was, either a Meningioma, Pituitary or Glioma. To do this, they wanted to use non-euclidian distances, and thus used a Graph based Neural Network structure to solve the problem. A graph based Neural Network uses a Graph Neural Network structure and combines it with a 26 layer Convolutional Neural Network structure. They achieved an accuracy of 99 percent. Another related work was called Computer-aided diagnosis system for tissue characterization of brain tumor on magnetic resonance images. In this work, In8 Arakeri and Reddy present a computer-aided diagnosis (CAD) system employing an ensemble classifier to distinguish between benign and malignant brain tumors using MRI scans. Their approach involves utilizing a segmentation technique to automatically isolate tumor tissue from MR images. Extracted features such as texture, shape, and border characteristics are then employed for representation. Significant features are determined through a feature ranking process based on information acquisition and independent component analysis. These features are subsequently used to train an ensemble classifier, incorporating SVM, artificial neural network, and KNN classifiers, to characterize the tumor. In contrast, Pereira et al.9 employed a fully convolutional 3D method. However, the transition from 2D-CNN to 3D presents challenges such as increased parameter complexity and greater memory and computational requirements. Addressing these concerns involves considerations such as network depth, training sampling techniques, and the utilization of the fully convolutional method to achieve dense inference.

CNNs are adept at identifying features through various layers such as convolution layers, pooling layers, and fully connected layers via backpropagation. Brain Tumor Classification Using Convolutional Neural Network by Das focuses on developing a CNN model specifically for classifying brain tumors in T1-weighted contrast-enhanced MRI images.

The methodology involves two primary steps: image preprocessing using diverse techniques and subsequent classification using the CNN model. The experimentation was conducted on a dataset comprising 3064 images encompassing glioma, meningioma, and pituitary tumor types.

The results indicate a high testing accuracy of 94.39 percent, with an average precision of 93.33 percent and an average recall of 93 percent. Notably, this CNN model outperformed several established methods, highlighting its efficacy in medical image classification.

Since we plan to use Resnet, we looked at a paper that used resnet-50 for brain tumor classification by Sahaii. They used 256 x 256 images as their dataset, and initially set up their neural network with just an input layer, a hidden layer and finally an output layer. They then applied transfer learning with a ReLu activation function to model the data.

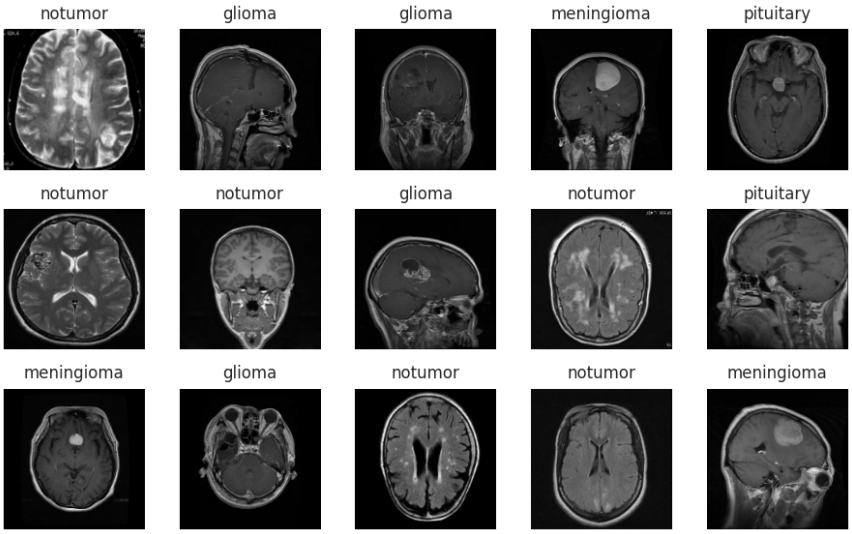
ResNet was employed to address the challenges of training deep neural networks effectively. By leveraging skip connections, ResNet allowed gradients to flow more smoothly during training, mitigating the vanishing gradient problem. This enabled the network to learn residual mappings, which are easier to optimize, thus improving classification accuracy. Additionally, by skipping problematic layers, ResNet prevented performance degradation, making it suitable for training deep networks for tasks such as brain tumor classification.

They were able to get an accuracy of 95.3 percent with a precision of 93.7 percent precision with a Resnet-50, and up to a 97 percent accuracy in classifying no tumors. They used trasnfer learning and data augmentation to ensure that the lack of data didn't affect their bias.

3 Methodology

Dataset and Preprocessing Training utilizes the Kaggle Brain Tumor, consisting of 5720 samples of 48x48 pixel gray-scale brain images once rescaled. The brain scans are uniformly centered and occupy a standardized space in each image.The objective is to classify each face based on the expressed emotion

in the facial expression, assigning it to one of four categories, a pituitary brain tumor, a meningioma brain tumor, a giloma brain tumor or if the scan has no tumor .This data-set has 2 columns, image and classification.

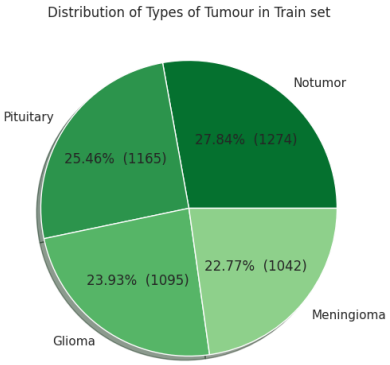


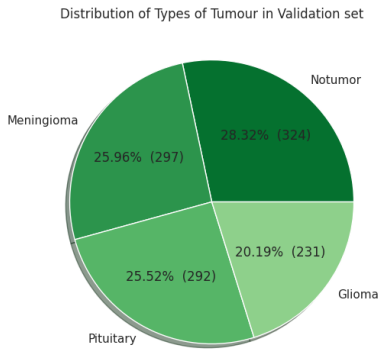
The brain id column includes what type of tumor it is, if it is a tumor, and the image column contains the images.

For pre-processing the data we converted the images to gray scale and standardized the pixel labels.

The target labels are encoded as 1D integers,to optimize training efficiency,converted them into categorical data. Reshaping the data into 48,48 pixels and 1 channel which will be a gray scale image for training. When there is a significant disparity in the number of images between classes, it leads to imbalance and biases the model towards the over represented class. To overcome this issue, data augmentation is employed, utilizing ImageDataGenerator in Keras.We have used rescaling the pixel values by dividing it with 255 and rotating the images and flipping the images for data augmentation.

Then, we split the data into training and testing, and of those 5720, we used 4576 for the training step and the rest for testing. Of that training set, we set aside 1144 for validation. To ensure that the training and validation sets have a representative sample (so we don't have to do any sort of cross validation which would be more computationally expensive), we made a pie chart of the training data to show the distribution of different classes.





Model 1 Used Convolutional Neural Network(CNN) to build the model for brain tumor classification. We have Initialized a sequential model with a 3 blocks(each block is stacked with convolutional layers(Conv2D) with relu activation function and maxpooling layer) and followed by dropout layer(0.2 rate) and flatten and then 2 fully connected layers to the output of 4 classifications and we have 319140 trainable parameters. you can check the model flow in the below figure

```
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 sequential (Sequential)     (None, 48, 48, 3)         0

 rescaling (Rescaling)       (None, 48, 48, 3)         0

 conv2d (Conv2D)             (None, 48, 48, 16)        448

 max_pooling2d (MaxPooling2  (None, 24, 24, 16)        0
 D)

 conv2d_1 (Conv2D)           (None, 24, 24, 32)        4640

 max_pooling2d_1 (MaxPoolin  (None, 12, 12, 32)        0
 g2D)

 conv2d_2 (Conv2D)           (None, 12, 12, 64)        18496

 max_pooling2d_2 (MaxPoolin  (None, 6, 6, 64)          0
 g2D)

 dropout (Dropout)           (None, 6, 6, 64)          0

 flatten (Flatten)           (None, 2304)              0

 dense (Dense)               (None, 128)               295040

 outputs (Dense)             (None, 4)                 516

=================================================================
Total params: 319140 (1.22 MB)
Trainable params: 319140 (1.22 MB)
Non-trainable params: 0 (0.00 Byte)
```

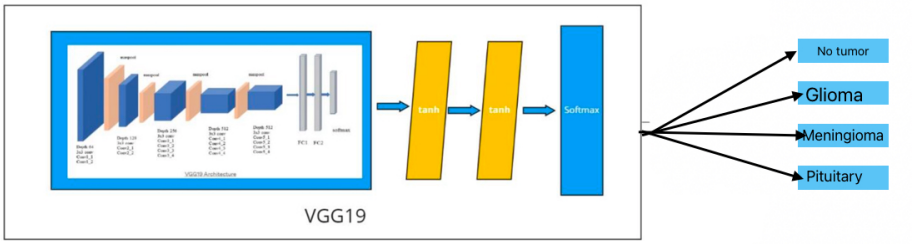
Model 2

Used CNN here similar to model 1 but we have updated the block here we used a block which has 2 conv2d layers followed by maxpooling layer and followed by dropout layer(0.2 rate) and flatten and then 2 fully connected layers to the output of 4 classifications and we have 367636 trainable parameters. you can check the model flow in the below figure

```
Model: "sequential_2"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 sequential (Sequential)     (None, 48, 48, 3)         0

 rescaling (Rescaling)       (None, 48, 48, 3)         0

 conv2d_3 (Conv2D)           (None, 48, 48, 16)        448

 conv2d_4 (Conv2D)           (None, 48, 48, 16)        2320

 max_pooling2d_3 (MaxPoolin  (None, 24, 24, 16)        0
 g2D)

 conv2d_5 (Conv2D)           (None, 24, 24, 32)        4640

 conv2d_6 (Conv2D)           (None, 24, 24, 32)        9248

 max_pooling2d_4 (MaxPoolin  (None, 12, 12, 32)        0
 g2D)

 conv2d_7 (Conv2D)           (None, 12, 12, 64)        18496

 conv2d_8 (Conv2D)           (None, 12, 12, 64)        36928

 max_pooling2d_5 (MaxPoolin  (None, 6, 6, 64)          0
 g2D)

 dropout_1 (Dropout)         (None, 6, 6, 64)          0

 flatten_1 (Flatten)         (None, 2304)              0

 dense_1 (Dense)             (None, 128)               295040

 outputs (Dense)             (None, 4)                 516

=================================================================
Total params: 367636 (1.40 MB)
Trainable params: 367636 (1.40 MB)
Non-trainable params: 0 (0.00 Byte)
```

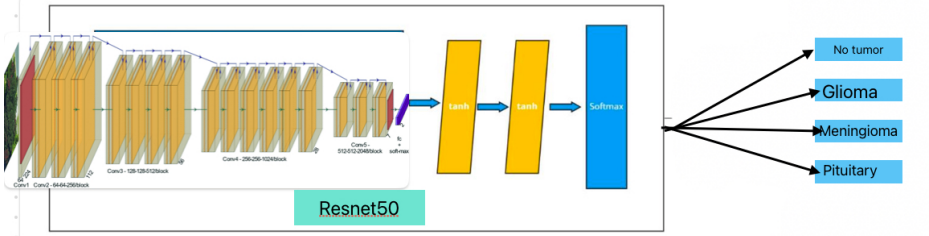
Model 3

Two custom dense layer with Tanh activation and output dense layer with softmax activation is added to base VGG19 to adapt the model for facial emotion recognition. which has total trainable parameters of 20024384 .



Model 4

Two custom dense layer with Tanh activation and output dense layer with softmax activation is added to base ResNet50 to adapt the model for facial emotion recognition. which has total trainable parameters of 23587712.

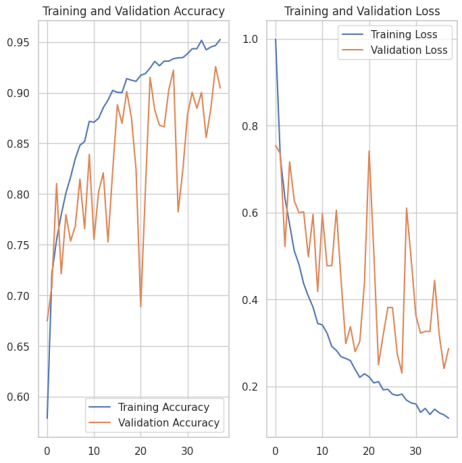


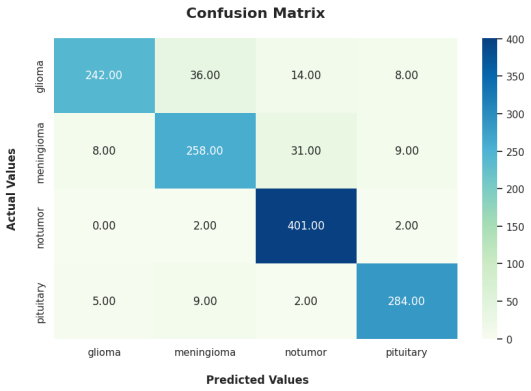
All the above models were Compiled with categorical_crossentropy loss function and used Adam as optimizer and accuracy as metrics and used callback for early stopping with patience of 10 and restore_best_weights is kept true.

4 Experimental Results

Model 1

We are able to achieved testing accuracy of 90% over 38 epochs with using early stopping(overall epochs of 50), using the accuracy metrics and history of the model we have got the classification report and confusion matrix for the predicted values and true value which are shown below. We were also able to get a precison of 91% and a recall of 90% over those 38 epochs. We also plotted the accuracy and loss curves between validation and train set which are shown below.





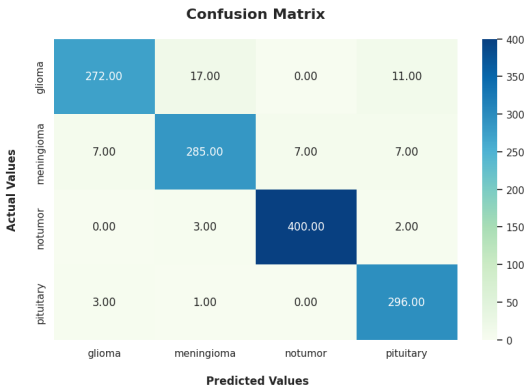
(a) confusion matix

	precision	recall	f1-score	support
glioma	0.95	0.81	0.87	300
meningioma	0.85	0.84	0.84	306
notumor	0.90	0.99	0.94	405
pituitary	0.94	0.95	0.94	300
accuracy			0.90	1311
macro avg	0.91	0.90	0.90	1311
weighted avg	0.91	0.90	0.90	1311

(b) classification report

Model 2

We are able to achieved testing accuracy of 96% over 40 epochs with using early stopping(overall epochs of 50), using the accuracy metrics and history of the model we have got the classification report and confusion matrix for the predicted values and true value which are shown below. We were also able to get a precison of 95% and a recall of 95% over those 43 epochs. We also plotted the accuracy and loss curves between validation and train set which are shown below.



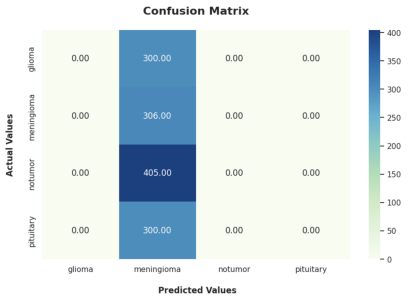
(a) confusion matix

	precision	recall	f1-score	support
glioma	0.96	0.91	0.93	300
meningioma	0.93	0.93	0.93	306
notumor	0.98	0.99	0.99	405
pituitary	0.94	0.99	0.96	300
accuracy			0.96	1311
macro avg	0.95	0.95	0.95	1311
weighted avg	0.96	0.96	0.96	1311

(b) classification report

Model 3

We are able to achieved testing accuracy of 31% over 12 epochs with using early stopping(overall epochs of 50), using the accuracy metrics and history of the model we have got the classification report and confusion matrix for the predicted values and true value which are shown below. We were also able to get a precison of 8% and a recall of 25% over those 38 epochs.



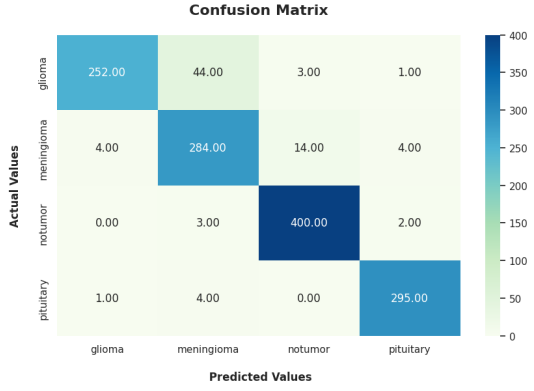
(a) confusion matix

	precision	recall	f1-score	support
glioma	0.00	0.00	0.00	300
meningioma	0.23	1.00	0.38	306
notumor	0.00	0.00	0.00	405
pituitary	0.00	0.00	0.00	300
accuracy			0.23	1311
macro avg	0.06	0.25	0.09	1311
weighted avg	0.05	0.23	0.09	1311

(b) classification report

Model 4

We are able to achieved testing accuracy of 94% over 34 epochs with using early stopping(overall epochs of 50), using the accuracy metrics and history of the model we have got the classification report and confusion matrix for the predicted values and true value which are shown below. We were also able to get a precison of 94% and a recall of 94% over those 34 epochs. We also plotted the accuracy and loss curves between validation and train set which are shown below.



(a) confusion matix

	precision	recall	f1-score	support
glioma	0.98	0.84	0.90	300
meningioma	0.85	0.93	0.89	306
notumor	0.96	0.99	0.97	405
pituitary	0.98	0.98	0.98	300
accuracy			0.94	1311
macro avg	0.94	0.93	0.94	1311
weighted avg	0.94	0.94	0.94	1311

(b) classification report

	Epochs	Accuracy	Precision	Recall
Model1	38	.90	.91	.90
Model2	43	.96	.95	.95
Model3/VGG	12	.31	.08	.25
Model4/Resnet	34	.94	.94	.94

Figure 6: Metrics for each model

This shows the metrics for all four of the models, and highlights the well-doing of model 2 with the blocks of convolution, and it also highlights the value of using more epochs.

5 Conclusion

In conclusion, we were able to achieve up to a 96 percent accuracy in a model we had that included 3 blocks which included 2 convolutional layers and a pooling layer. We were able to do this after one set of a convolutional neural network. Then we tried a VGG and a ResNet model, and we were not able to get the same accuracy as we were with the previous models. This is due to the limitations of those models and transfer learning in general on a small number of epochs that we can afford with Google Collab.

Another side note is that we were not able to achieve the accuracy mentioned on the related works with the Artificial Neural Network with a 99 percent test accuracy that they have. This highlights the usefulness of using more complex structures of a neural network that enable further hyper-parameter tuning.

It is worth mentioning that there can be a lot of variance in brain tumour classifications as an image of a brain can seem very similar between types of tumours.

We can explore other parameters in the future to increase our accuracy to be on par with the industry standard. For example, one thing we can do is increase the number of parameters, and with that increase the number of layers or increase the number of epochs.

To do this, we can add an attention layer, and for this, we would need to further clean the data to be compatible with an extra attention layer. We will also experiment with other activation function such as tanh to see if it can improve the accuracy scores.

After this, we will experiment with different models such as an AlexNet model or a Graph Neural Network or other ways to perform transfer learning.

In conclusion, the development of a robust model capable of accurately classifying brain tumors into categories such as Meningioma, Glioma, Pituitary, and none holds immense promise for the field of neurology and oncology. Such a tool not only streamlines diagnosis and treatment planning but also paves the way for personalized medicine, enabling tailored interventions for patients based on the specific characteristics of their tumors. With further refinement and validation, this model could become an indispensable asset in clinical practice, empowering healthcare professionals with the precision and efficiency needed to combat brain tumors effectively.

6 References

References

[1] Arakeri, M., Reddy, G. Computer-aided diagnosis system for tissue characterization of brain tumor on magnetic resonance images. SIViP 9(2), 409–425 (2015).

[2] Das, S., Aranya, O. R. R., Labiba, N. N. (2019, May). Brain tumor classification using convolutional neural network. In 2019 1st international conference on advances in science, engineering and robotics technology (ICASERT) (pp. 1-5). IEEE.

[3] El-Feshawy, S. A., Saad, W., Shokair, M., Dessouky, M. (2023). IoT framework for brain tumor detection based on optimized modified ResNet 18 (OMRES). The Journal of Supercomputing, 79(1), 1081-1110.

[4] Majib, M. S., Rahman, M. M., Sazzad, T. S., Khan, N. I., Dey, S. K. (2021). Vgg-scnet: A vgg net-based deep learning framework for brain tumor detection on mri images. IEEE Access, 9, 116942-116952.

[5] Ravinder, M., et al. "Enhanced Brain Tumor Classification Using Graph Convolutional Neural Network Architecture." Nature News, Nature Publishing Group, 11 Sept. 2023, `www.nature.com/articles/s41598-023-41407-8#Sec3.`

[6] Sahaai, M. B., Jothilakshmi, G. R., Ravikumar, D., Prasath, R., Singh, S. (2022, May). ResNet-50 based deep neural network using transfer learning for brain tumor classification. In AIP Conference Proceedings (Vol. 2463, No. 1). AIP Publishing.

[7] Xiao, Y., Yin, H., Wang, S. H., Zhang, Y. D. (2021). TReC: transferred ResNet and CBAM for detecting brain diseases. Frontiers in Neuroinformatics, 15, 781551.