

Clasificarea Tumorilor Cerebrale din Imagini MRI folosind Rețele Neuronale Convoluționale

Flanja Tudor-Calin

Cuprins

1		4
1.1	Cerința 1: Implementarea clasei personalizate pentru setul de date	4
1.1.1	Descriere	4
1.1.2	Rezultatele obținute	4
1.1.3	Rezultatul vizualizării primei imagini	5
2		6
2.0.1	Cerința 2: Împărțirea dataset-ului în subseturi pentru antrenare și validare	6
2.0.2	Rezultatele împărțirii	6
2.0.3	Analiza unei mostre din subsetul de validare	6
3		7
3.1	Cerința 3	7
3.1.1	Distribuția Claselor în Seturile de Date	7
3.1.2	Observații	9
3.1.3	Soluții pentru Dezechilibru	9
4		10
4.1	Cerința 4	10
4.1.1	Analiza Vizuală a Imaginilor	10
4.1.2	Observații	11
4.1.3	Recomandări	11
5		12
5.1	Cerința 5	12
5.1.1	Asigurarea Consistenței și Integrității Setului de Date	12
5.1.2	Metodologie	12
5.1.3	Rezultate ale Procesării	12
5.1.4	Observații și Concluzii	13
6		14
6.1	Cerința 6	14
6.1.1	Preprocesarea și Normalizarea Imaginilor	14
6.1.2	Compararea Vizuală a Imaginilor Preprocesate	14
6.1.3	Concluzii	16
7		17
7.1	Cerința 7: Implementarea pipeline-ului de antrenare și evaluare	17
7.1.1	Descriere	17
7.1.2	Rezultate obținute	17
7.1.3	Evoluția pierderii și a acurateței în timpul antrenării	18

7.1.4	Rezultatele evaluării pe subsetul de testare	19
7.1.5	Concluzii	20

Capitolul 1

1.1 Cerința 1: Implementarea clasei personalizate pentru setul de date

1.1.1 Descriere

A fost implementată o clasă personalizată derivată din `torch.utils.data.Dataset` pentru gestionarea setului de date *Brain Tumor Classification (MRI)*. Aceasta permite încărcarea imaginilor și etichetelor asociate în mod eficient, utilizând un mod de încărcare *lazy*, economisind astfel resursele de memorie.

Clasa implementată are următoarele funcționalități:

- Primește ca argument calea către datele de antrenare sau testare (`train/test`).
- Încarcă imaginile și etichetele din categoriile disponibile (`glioma_tumor`, `meningioma_tumor`, `no_tumor`, `pituitary_tumor`).
- Permite accesarea individuală a imaginilor și aplicarea transformărilor specificate.

După implementarea acestei clase, s-au aplicat transformări de redimensionare și conversie a imaginilor la tensori, asigurând compatibilitatea acestora cu rețelele neuronale utilizate ulterior.

1.1.2 Rezultatele obținute

- **Numărul total de imagini în dataset:**
 - Număr de mostre pentru antrenare: 2870
 - Număr de mostre pentru testare: 394
- **Analiza primei mostre din dataset:**
 - Dimensiunea imaginii: `torch.Size([3, 128, 128])`
 - Eticheta asociată: 0 (*Glioma Tumor*)

1.1.3 Rezultatul vizualizării primei imagini

Imaginea de mai jos reprezintă un exemplu din setul de date de antrenare. Eticheta asociată este 0, corespunzând categoriei *Glioma Tumor*.

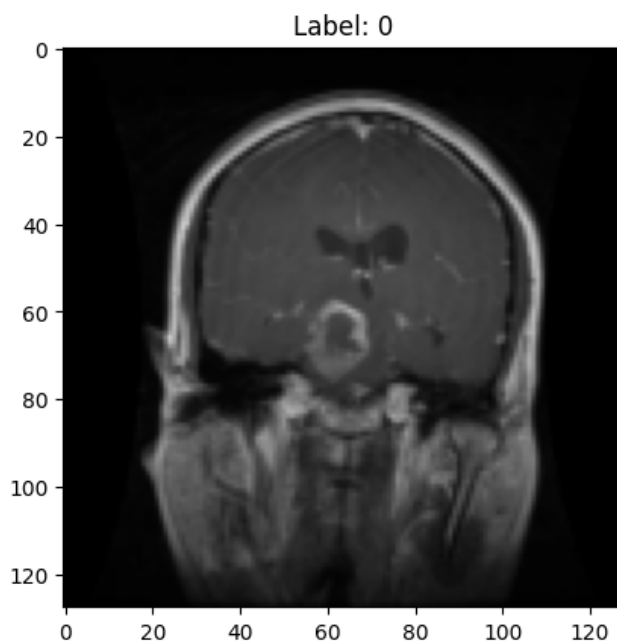


Figura 1.1: Prima imagine afișată din dataset. Etichetă: 0 (*Glioma Tumor*).

Capitolul 2

2.0.1 Cerința 2: Împărțirea dataset-ului în subseturi pentru antrenare și validare

Descriere:

Pentru a asigura o evaluare robustă și generalizabilă a modelului, dataset-ul de antrenare a fost împărțit în două subseturi:

- 80% din imagini au fost alocate pentru antrenare.
- 20% din imagini au fost rezervate pentru validare.

Această metodă de împărțire permite utilizarea eficientă a datelor și poate fi extinsă ulterior în strategii de tip *cross-validation*. Pentru realizarea acestei împărțiri, am utilizat funcția `random_split` din biblioteca PyTorch, care asigură selecția aleatorie a mostrelor.

După împărțirea dataset-ului, au fost create *DataLoaders* pentru subseturile de antrenare și validare, cu un *batch size* de 32, asigurând eficiența procesului de antrenare.

2.0.2 Rezultatele împărțirii

- Număr de mostre pentru antrenare: 2296
- Număr de mostre pentru validare: 574

2.0.3 Analiza unei mostre din subsetul de validare

- Dimensiunea imaginii: `torch.Size([3, 128, 128])`
- Eticheta asociată: 0 (*Glioma Tumor*)

Capitolul 3

3.1 Cerința 3

3.1.1 Distribuția Claselor în Seturile de Date

Pentru a determina echilibrul dintre categoriile de tumori în seturile de date, s-a realizat o vizualizare a distribuției claselor pentru fiecare dintre cele trei seturi: antrenare, validare și testare. Acest proces ajută la identificarea eventualelor clase sub-reprezentate care pot necesita tehnici suplimentare de preprocesare.

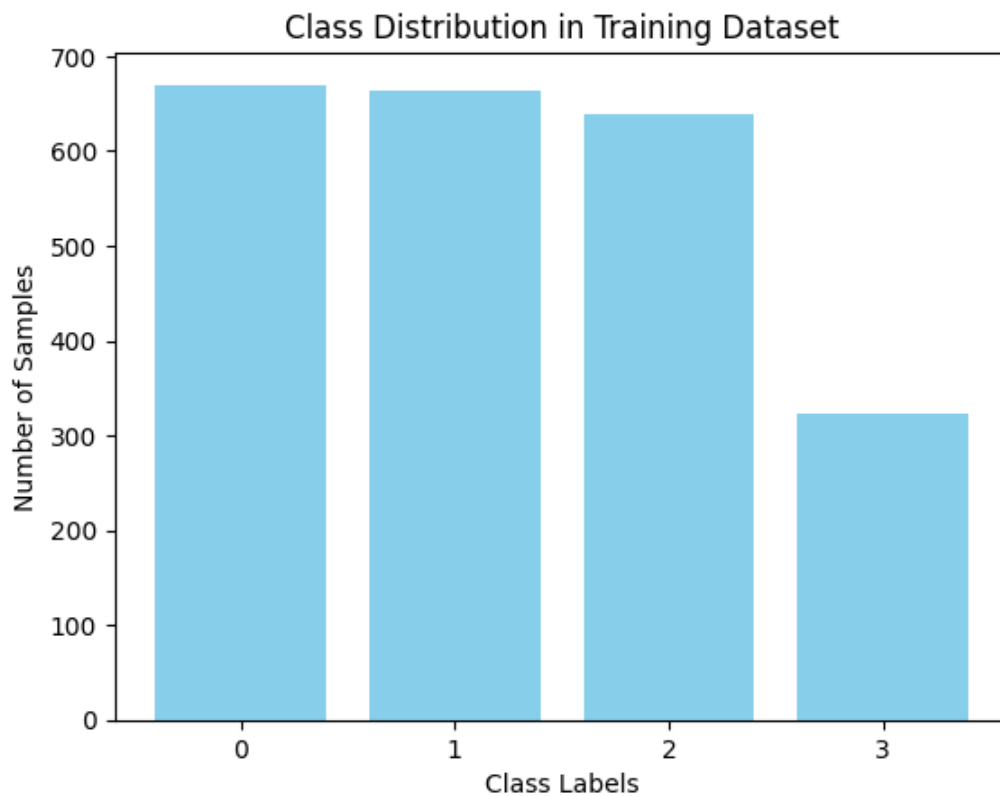


Figura 3.1: Distribuția claselor în setul de date de antrenare.

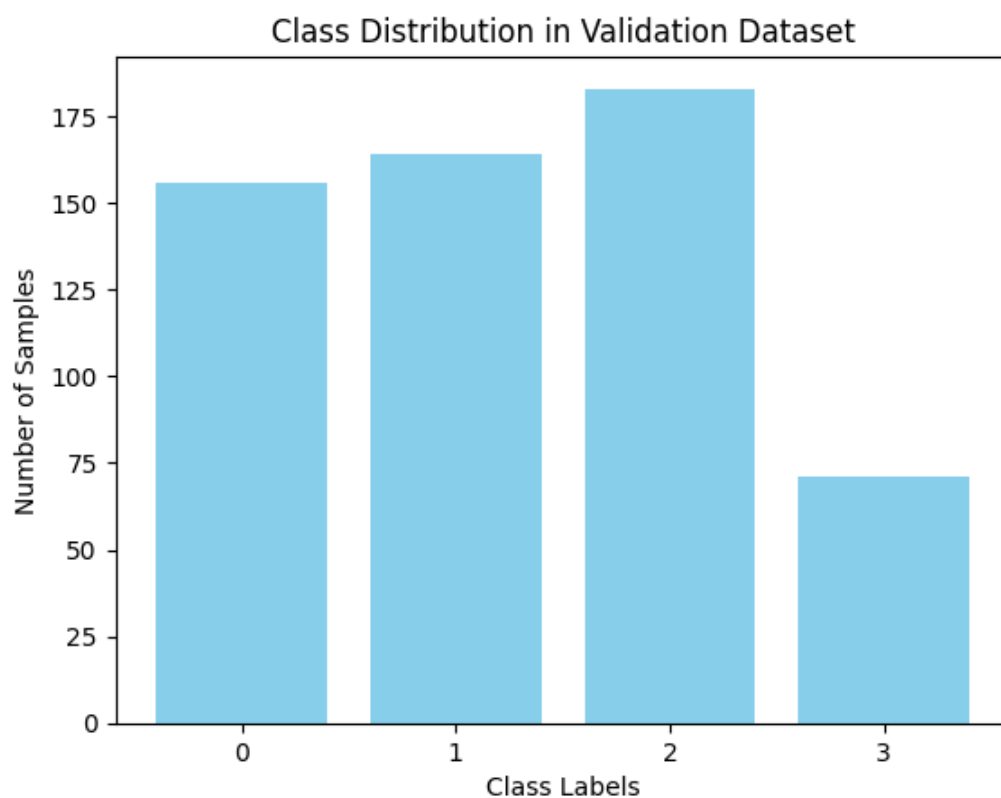


Figura 3.2: Distribuția claselor în setul de date de validare.



Figura 3.3: Distribuția claselor în setul de date de testare.

3.1.2 Observații

În urma analizei distribuției claselor:

- Setul de antrenare este relativ echilibrat între categoriile de tumori, cu toate cele patru clase reprezentate.
- În setul de validare și setul de testare, există uşoare variații în numărul de mostre pentru fiecare clasă, dar nu există clase complet sub-reprezentate.

3.1.3 Soluții pentru Dezechilibru

Pentru a trata eventualele dezechilibre, se pot aplica următoarele tehnici:

- **Augmentarea datelor:** Generarea de noi mostre pentru clasele sub-reprezentate prin aplicarea de transformări (rotații, flip-uri, modificări de contrast etc.).
- **Reponderarea claselor:** Utilizarea unei funcții de pierdere ponderată pentru a ține cont de frecvența fiecărei clase.
- **Sub-eșantionarea:** Reducerea dimensiunii claselor supra-reprezentate pentru a echilibra proporțiile.

Capitolul 4

4.1 Cerința 4

4.1.1 Analiza Vizuală a Imaginilor

Pentru a înțelege mai bine variabilitatea și tiparele vizuale ale fiecărei categorii de tumori cerebrale, au fost extrase și afișate un set de imagini pentru fiecare dintre cele patru categorii prezente în dataset (*glioma_tumor*, *meningioma_tumor*, *no_tumor*, *pituitary_tumor*).

Aceste imagini permit observarea directă a caracteristicilor fiecărei clase, astfel încât să putem identifica diferențe semnificative între categorii și să înțelegem posibilele provocări de clasificare.

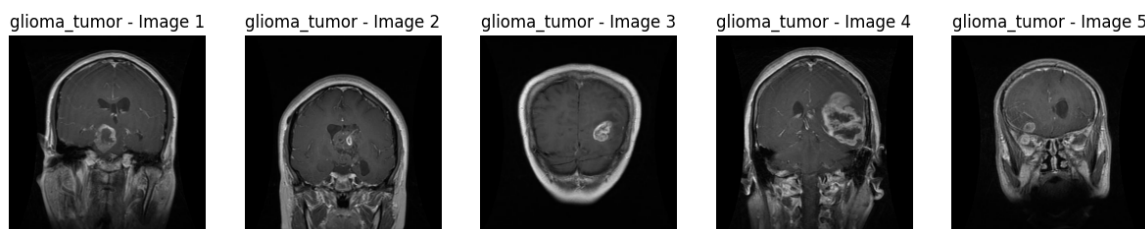


Figura 4.1: Imagini din categoria *no_tumor*.

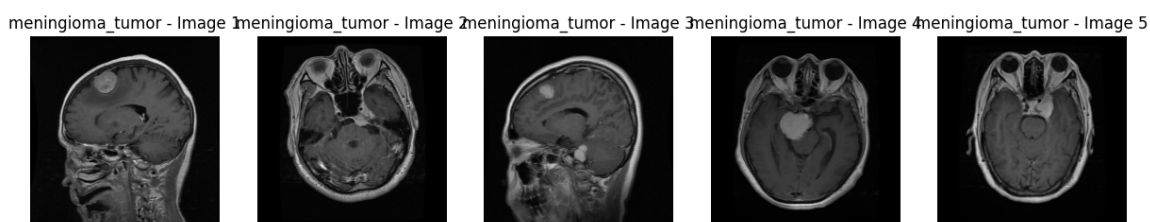


Figura 4.2: Imagini din categoria *pituitary_tumor*.



Figura 4.3: Imagini din categoria *meningioma_tumor*.

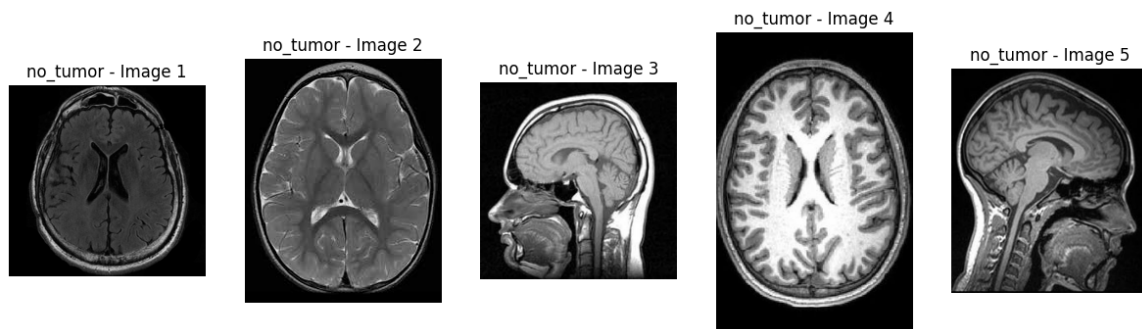


Figura 4.4: Imagini din categoria *glioma_tumor*.

4.1.2 Observații

În urma analizei vizuale:

- Fiecare categorie de tumori prezintă caracteristici vizuale distincte, cum ar fi forme, dimensiuni și texturi diferite.
- În unele cazuri, imaginile sunt similare între clase, ceea ce poate reprezenta o dificultate suplimentară pentru clasificare.

4.1.3 Recomandări

Pentru a îmbunătăți modelul de clasificare:

- Creșterea setului de date prin augmentarea imaginilor pentru clasele care prezintă variabilitate ridicată.
- Aplicarea unor tehnici avansate de preprocesare, cum ar fi normalizarea intensității sau eliminarea zgomotului, pentru a îmbunătăți calitatea imaginilor.

Capitolul 5

5.1 Cerința 5

5.1.1 Asigurarea Consistenței și Integrității Setului de Date

Pentru a asigura consistența și integritatea setului de date, au fost implementate mai multe verificări și ajustări asupra imaginilor din setul de date. Aceste operațiuni au inclus standardizarea numărului de canale, dimensiunile imaginilor și valorile pixelilor.

5.1.2 Metodologie

Procesul de preprocesare a inclus următorii pași:

- **Verificarea și uniformizarea numărului de canale:** Imaginile au fost verificate pentru a avea același număr de canale (RGB cu 3 canale). Imaginile în alb-negru au fost convertite în format RGB.
- **Standardizarea dimensiunilor imaginilor:** Toate imaginile au fost redimensionate la 256x256 pixeli pentru a asigura uniformitatea intrării în model.
- **Normalizarea valorilor pixelilor:** Valorile pixelilor au fost scalate în intervalul $[0, 1]$ pentru a facilita procesul de învățare automată.

5.1.3 Rezultate ale Procesării

Pe baza codului executat, procesarea setului de date a fost finalizată cu succes. Mai jos sunt evidențiate câteva exemple de mesaje de procesare afișate în timpul rulării scriptului:

```
Processing category: glioma_tumor
Processed image: gg (801).jpg
...
Processing category: pituitary_tumor
...
Processed image: p (348).jpg
Processing category: meningioma_tumor
Processed image: m3 (110).jpg
...
Processed image: m (61).jpg
Processing category: no_tumor
Processed image: image(172).jpg
...
Processed image: image(119).jpg
```

```
Processed image: image(238).jpg  
Processed image: image(120).jpg  
Dataset processing complete.
```

5.1.4 Observații și Concluzii

În urma acestei etape:

- Toate imaginile din setul de date au fost convertite în format RGB, asigurând consistența canalelor.
- Dimensiunile tuturor imaginilor au fost ajustate la 256x256 pixeli, eliminând posibilele probleme cauzate de variațiile de dimensiune.
- Valorile pixelilor au fost normalizate, ceea ce îmbunătățește compatibilitatea cu modelele de învățare automată.

Această etapă de preprocesare a fost crucială pentru asigurarea unui set de date curat și uniform, gata de utilizare în următoarele faze ale proiectului.

Capitolul 6

6.1 Cerința 6

6.1.1 Preprocesarea și Normalizarea Imaginilor

Pentru a crește calitatea și consistența datelor de intrare, au fost aplicate o serie de tehnici de preprocesare și normalizare asupra imaginilor din setul de date. Aceste operații permit reducerea zgomotului, îmbunătățirea contrastului și asigurarea unei scale uniforme pentru valorile pixelilor.

- **Gaussian Blur:** Aplicat pentru a reduce zgomotul din imagini.
- **Normalizarea Intensității:** Valorile pixelilor sunt scalate pentru a se încadra într-un interval uniform.
- **CLAHE (Contrast Limited Adaptive Histogram Equalization):** Utilizat pentru a îmbunătăți contrastul imaginilor.
- **Sobel Edge Detection:** Aplicat pentru a detecta marginile din imagini.
- **Adjustarea Luminozității:** Folosit pentru a spori vizibilitatea detaliilor din imagini.

6.1.2 Compararea Vizuală a Imaginilor Preprocesate

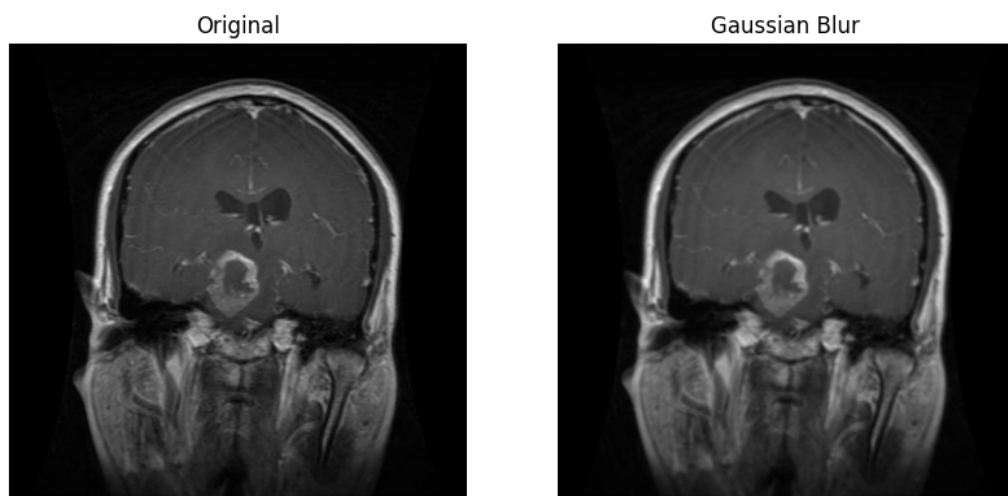


Figura 6.1: Comparare între imaginea originală și imaginea cu Gaussian Blur aplicat.

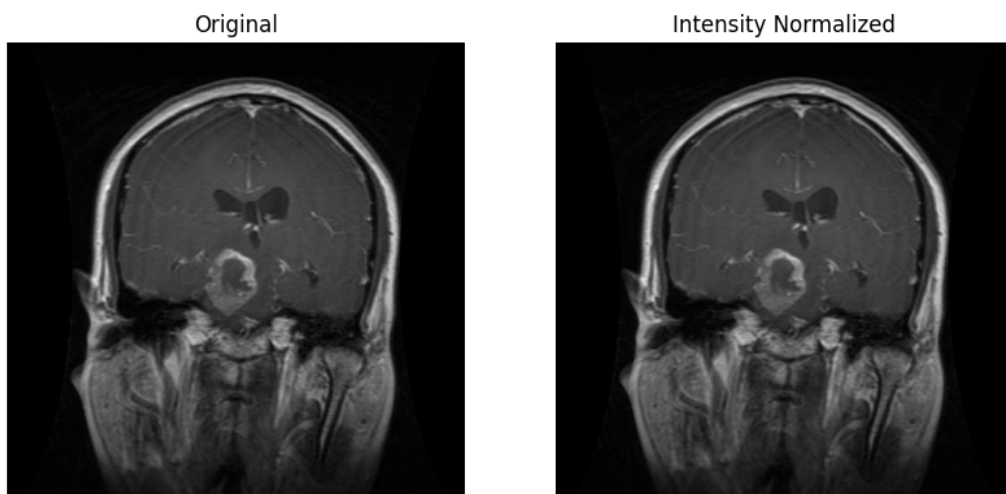


Figura 6.2: Comparare între imaginea originală și imaginea cu intensitatea normalizată.

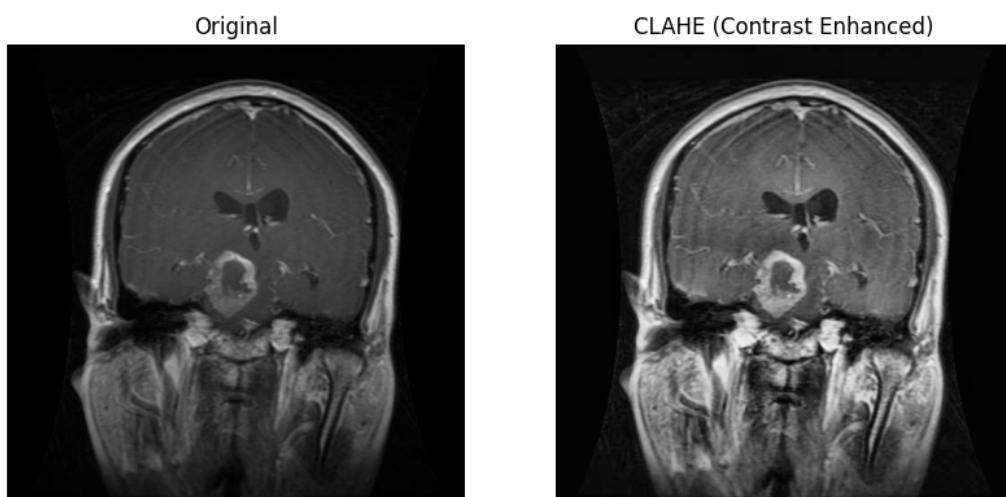


Figura 6.3: Comparare între imaginea originală și imaginea cu CLAHE aplicat.

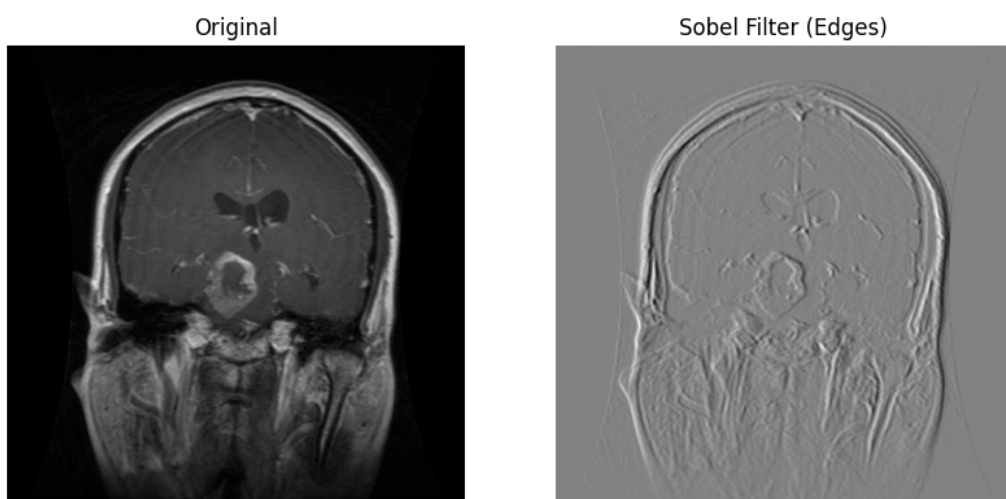


Figura 6.4: Comparare între imaginea originală și imaginea cu Sobel Edge Detection aplicat.

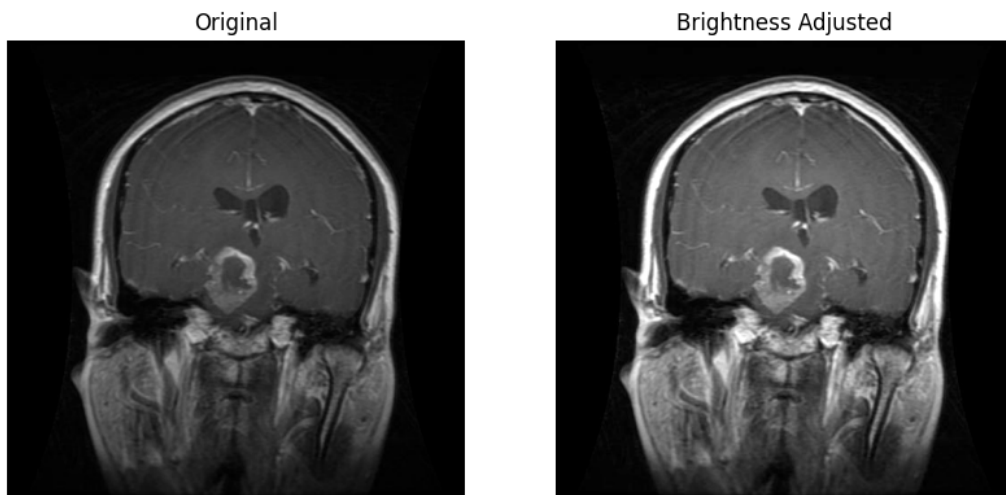


Figura 6.5: Comparare între imaginea originală și imaginea cu luminozitatea ajustată.

6.1.3 Concluzii

Preprocesarea imaginilor prin tehnicile aplicate a dus la îmbunătățirea clarității și consistenței acestora. Fiecare tehnică a fost selectată în funcție de necesitatea specifică a tipului de imagistică utilizat, iar rezultatele obținute evidențiază impactul vizual al fiecărei transformări asupra datelor.

Capitolul 7

7.1 Cerința 7: Implementarea pipeline-ului de antrenare și evaluare

7.1.1 Descriere

Un pipeline complet pentru antrenarea și evaluarea unei rețele neuronale convoluționale (CNN) a fost implementat pentru clasificarea tumorilor cerebrale utilizând setul de date *Brain Tumor Classification (MRI)*. Procesul include următoarele etape:

1. **Încărcarea datelor:** Utilizarea clasei personalizate derivată din `torch.utils.data.Dataset` pentru citirea și prelucrarea imaginilor, incluzând augmentări de date și normalizări.
2. **Definirea modelului:** O arhitectură CNN personalizată a fost implementată pentru clasificarea în cele patru categorii de tumori (*Glioma*, *Meningioma*, *No Tumor*, *Pituitary*).
3. **Definirea funcției de pierdere și optimizatorului:** Funcția `CrossEntropyLoss` și optimizatorul `Adam` au fost utilizate pentru antrenarea modelului.
4. **Antrenarea modelului:** Modelul a fost antrenat timp de 10 epoci, iar performanța a fost evaluată pe subsetul de validare după fiecare epocă.
5. **Evaluarea modelului:** După antrenare, modelul a fost evaluat pe subsetul de testare pentru a calcula metrici precum acuratețea, precizia, *recall* și *F1-score*, și pentru a genera matricea de confuzie.

7.1.2 Rezultate obținute

- Număr de epoci: **10**
- Cea mai bună acuratețe pe subsetul de validare: **85.04%**
- Acuratețea pe subsetul de testare: **84%**

7.1.3 Evoluția pierderii și a acurateței în timpul antrenării

```
Epoch 1/10: 100%|██████████| 51/51 [01:06<00:00, 1.30s/it]
Epoch 1/10, Train Loss: 1.1557, Val Loss: 0.8411, Val Accuracy: 0.6959
Epoch 2/10: 100%|██████████| 51/51 [01:04<00:00, 1.27s/it]
Epoch 2/10, Train Loss: 0.5785, Val Loss: 0.6261, Val Accuracy: 0.7780
Epoch 3/10: 100%|██████████| 51/51 [01:04<00:00, 1.27s/it]
Epoch 3/10, Train Loss: 0.3642, Val Loss: 0.6393, Val Accuracy: 0.7712
Epoch 4/10: 100%|██████████| 51/51 [01:04<00:00, 1.27s/it]
Epoch 4/10, Train Loss: 0.2053, Val Loss: 0.6719, Val Accuracy: 0.8214
Epoch 5/10: 100%|██████████| 51/51 [01:05<00:00, 1.28s/it]
Epoch 5/10, Train Loss: 0.1413, Val Loss: 0.6253, Val Accuracy: 0.8282
Epoch 6/10: 100%|██████████| 51/51 [01:03<00:00, 1.25s/it]
Epoch 6/10, Train Loss: 0.0655, Val Loss: 0.6979, Val Accuracy: 0.8504
Epoch 7/10: 100%|██████████| 51/51 [01:04<00:00, 1.26s/it]
Epoch 7/10, Train Loss: 0.0363, Val Loss: 0.7489, Val Accuracy: 0.8504
Epoch 8/10: 100%|██████████| 51/51 [01:05<00:00, 1.28s/it]
Epoch 8/10, Train Loss: 0.0173, Val Loss: 0.7964, Val Accuracy: 0.8620
Epoch 9/10: 100%|██████████| 51/51 [01:04<00:00, 1.27s/it]
Epoch 9/10, Train Loss: 0.0103, Val Loss: 0.9013, Val Accuracy: 0.8504
Epoch 10/10: 100%|██████████| 51/51 [01:05<00:00, 1.28s/it]
Epoch 10/10, Train Loss: 0.0077, Val Loss: 0.9296, Val Accuracy: 0.8494
```

Figura 7.1: Graficul pierderii pentru seturile de antrenare și validare.

```
Classification Report:
              precision    recall  f1-score   support

   Glioma           0.82       0.89       0.86         66
 Meningioma         0.97       0.91       0.94         66
   No Tumor         0.76       0.73       0.74         66
   Pituitary        0.79       0.81       0.80         32

 accuracy                   0.84         230
 macro avg           0.83       0.84       0.83         230
 weighted avg        0.84       0.84       0.84         230
```

Figura 7.2: Graficul acurateței pe subsetul de validare.

7.1.4 Rezultatele evaluării pe subsetul de testare

Matricea de confuzie:

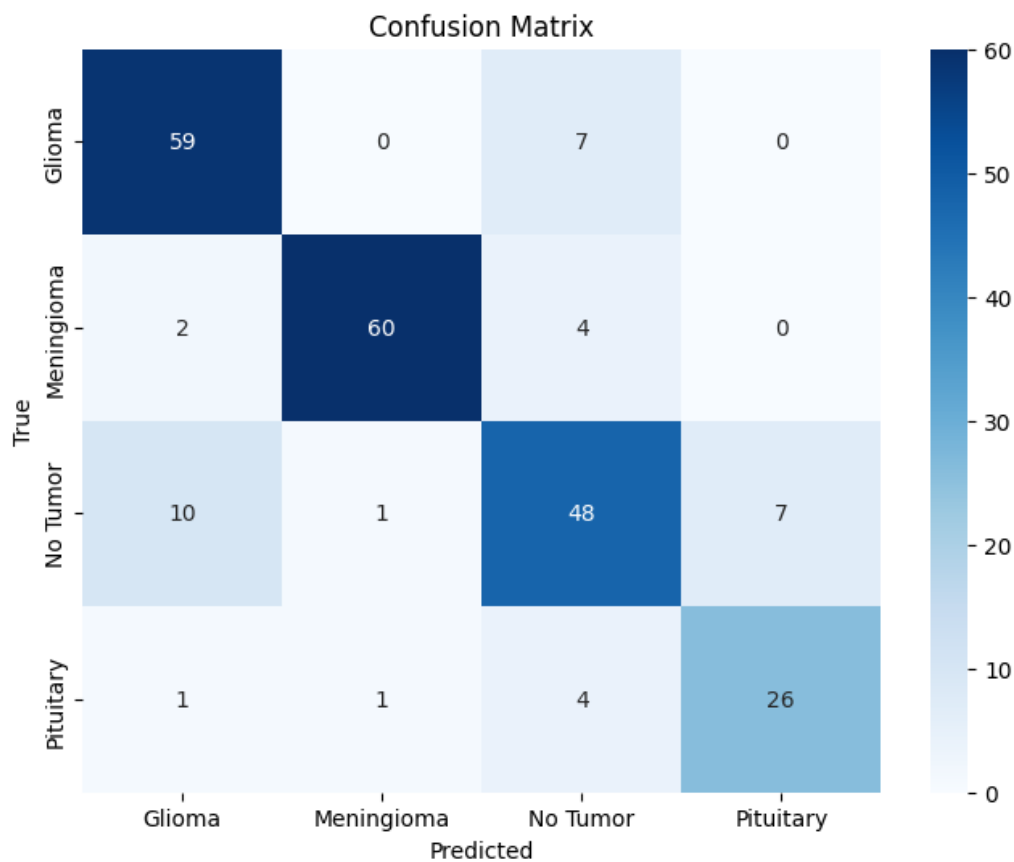


Figura 7.3: Matricea de confuzie pentru subsetul de testare.

Raportul de clasificare:

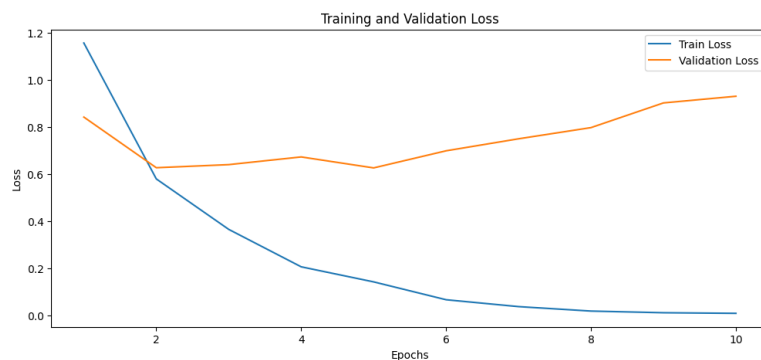


Figura 7.4: Raportul de clasificare pentru subsetul de testare.

Jurnalul antrenării:

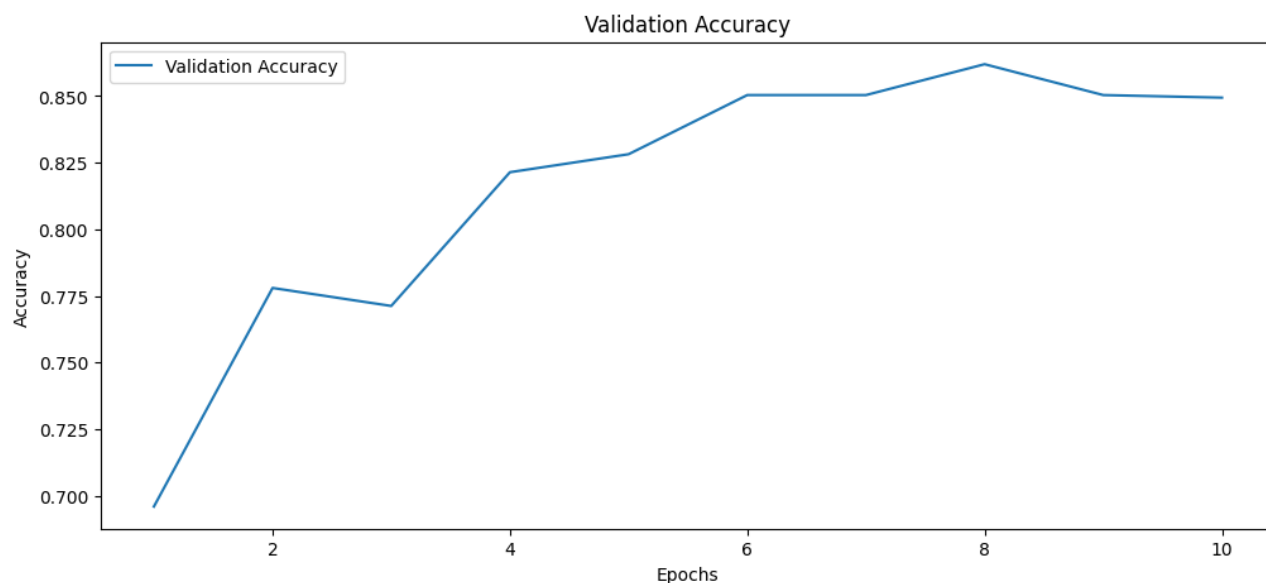


Figura 7.5: Jurnalul procesului de antrenare.

7.1.5 Concluzii

Implementarea pipeline-ului a demonstrat capacitatea modelului de a clasifica imaginile în mod eficient, cu o acuratețe de **84%** pe subsetul de testare. Matricea de confuzie și raportul de clasificare evidențiază distribuția performanței pe fiecare clasă, indicând că cele mai mari îmbunătățiri ar putea fi obținute prin aplicarea de tehnici avansate de augmentare a datelor sau optimizare.