**КЛАСИФИКАЦИЈА НА ТУМОР НА МОЗОК СО ПОМОШ НА ДЛАБОКО УЧЕЊЕ**

Петар Трајковски 172/2021, Александар Гулабоски 6/2021, Стефанија Трајковска171/2021

Факултет за електротехника и информациски технологии,

Универзитет „Св. Кирил и Методиј“, Скопје, Македонија

petartrajkovski002@gmail.com, gulaboskialeksandar5@gmail.com, stefanija18trajkovska@gmail.com

***Апстракт –*** Туморите на мозок претставуваат сериозен здравствен предизвик, со значително влијание врз квалитетот на животот и преживувањето на пациентите. Нивната патогенеза е комплексна и вклучува различни генетски, молекуларни и еколошки фактори, што го прави нивното проучување и рана дијагноза клучен предизвик за медицинските истражувачи. Раната и точна дијагноза на туморите на мозокот е од клучно значење за успешен третман на пациентите.

Во ова истражување, имплементирани се модели на длабоко учење за автоматска класификација на мозочни тумори користејќи снимки од магнетна резонанца (MRI). Моделите се тренираат врз база на податочен сет кој содржи четири категории: **глиом**, **менингиом**, **хипофиза** и **нормален наод** (без тумор).

# **Вовед**

Тумор на мозокот претставува абнормален раст на клетки во мозочното ткиво, кој може да биде **бениген (неканцероген)** или **малиген (канцероген)**. Овие тумори можат да потекнуваат директно од мозочното ткиво (**примарни тумори**) или да се прошират од други делови на телото (**метастатски тумори**).

Точната причина за појавата на тумор на мозокот сè уште не е целосно разјаснета, но неколку фактори може да придонесат за негов развој. Тоа се: генетските мутации, јонизирачко зрачење (високи дози на радијација), семејна историја, хормонални промени, изложеност на токсини и хемикалии.

Туморите на мозокот се релативно ретки во споредба со други видови рак. Се јавуваат кај 5-10 лица на 100.000 жители годишно.

Дијагнозата на мозочен тумор вклучува неколку методи, вклучувајќи магнетна резонанца, компјутеризирана томографија, биопсија и невролошки испитувања.

Датасетот кој се обработува вклучува три патолошки наоди:

* **Глиом,** слика 1.1**,** е тумор што се развива во поддржувачките клетки на мозокот и може да биде побавен или многу агресивен, влијае на функциите на мозокот и често предизвикува главоболки, напади и промени во однесувањето.
* **Менингиом** , слика 1.2, е тумор што расте во обвивките на мозокот и најчесто е бениген, но ако стане голем, може да притиска врз мозокот и да предизвика главоболки, проблеми со видот или движењето.
* **Хипофизен тумор** , слика 1.4, расте во хипофизата, жлезда што контролира многу хормони во телото, и може да предизвика хормонални нарушувања, проблеми со видот или главоболки.

Напредокот во медицината и технологијата овозможува подобра дијагностика и поефикасни третмани, што значително ги зголемува шансите за преживување и подобрување на квалитетот на животот на пациентите.

Датасетот што ќе се анализира се состои од 7 022 снимки од магнетна резонанца, поделени во два фолдера за тренирање и тестирање на моделите. Секој од двата фолдери е поделен на 4 фолдери (глиом, менингиом, нормален наод (без тумор) и хипофизен тумор). Тренинг множеството содржи 5712 снимки од кои глиом содржи 1321, меннгиом содржи 1339 , нормален наод(без тумор) 1595, слика 1.3, и хипофизен 1457. Додека тренинг множеството содржи 1310 од кои глиом содржи 300, меннгиом содржи 306 , нормален наод(без тумор) 405 и хипофизен 300.

|  |  |
| --- | --- |
|  |  |
| Слика 1.1  Глиом | Слика 1.2  Менингиом |

|  |  |
| --- | --- |
|  |  |
| Слика 1.3  Нормален наод | Слика 1.4  Хипофизен тумор |

# **Претпроцесирање**

Претпроцесирањето на податоци е клучен чекор во процесот на длабоко учење кој вклучува подготовка на податоците за анализа. Овој процес вклучува отстранување на дупликати, лабелирање и ковертирање во повеќе формати за да се обезбеди олеснето тренирање, прецизност и точност на моделите.

**II.I) Отстранување на дупликати**

Поради присуството на снимки кои изгледаат слични искористена е функција за отстранување на дупликати. Со функцијата “Compute hash” се добива хеш вредноста на сликата. При итерирање низ датасетот се добива хеш вредноста на секоја слика и се додава во посебна кофа(dictionary), каде клучот е хеш вредноста додека вредноста е апсолутната патека на снимката. Снимките со исти хеш вредности се додаваат во една кофа. Со фукнцијата “remove duplicates” се отстрануваат сите дупликати од кофите со цел во секоја кофа да остане само една снимка.

Во датасетот пронајдени се 297 дупликати кои исто така и се остранети, Слика 2.1.

|  |
| --- |
|  |
| Слика 2.1 Отстранети дупликати |

|  |
| --- |
|  |
|  |
| **II.II) Лабелирање на податоците**  Бидејќи моделите предвидуваат четири состојби потребно е снимкитее да бидаат лабелирани соодветно со состојбата која ја прикажуваат. За таа цел снимките се лабелирани со нумерички вредности од 0 до 3 и тоа:  Глиом – 0  Менингиом – 1  Нормален наод (без тумор) – 2  Хипофизен тумор – 3  Со цел лабелите да соодвествуваат со снимките, Слика 2.2, лабелата и соодветната патека на снимката се сместуваат во две листи истовремено (“train\_image\_paths” соодветно со “train\_labels” и “test\_image\_paths” соодветно со “test\_labels” ).   |  | | --- | |  | | Слика 2.2 Лабели со соодветни патеки | |
| **II.III) Мешање на датасетот**  За да се избегне пристрасност при тренирање на моделите потребно е да се измешаат снимките. Со цел да не се изгубат соодветните лабели за секоја снимка искористена е функцијата “shuffle\_data”. Функцијата shuffle\_data ги меша листите image\_paths и labels така што формира парови, ги меша случајно со ”random.shuffle”, а потоа ги разделува назад во две листи. Листите кои се добиваат се: “shuffled\_train\_image\_paths”, “shuffled\_train\_labels”, “shuffled\_test\_image\_paths” и “shuffled\_test\_labels”. Со ова се гарантира дека снимките и нивните соодветни ознаки остануваат поврзани по мешањето. Сликата 2.3 ги прикажува измешаните снимки и нивните соодветни лабели.   |  | | --- | |  | | Слика 2.3 Снимки и соодветни лабели по мешањето | |
|  |

**II.IV) Поделба на датасетот**

Од тренинг листите и лабелите (“shuffled\_train\_image\_paths” и “shuffled\_train\_labels”) се зема 20% од снимките за валидација за однапред да се процени перформансот на моделите и да се избегне презаситување (overfiting). Оваа функција враќа четири нови листи “shuffled\_val\_image\_paths” и “shuffled\_val\_labels” и намалените листи “shuffled\_train\_image\_paths” и “shuffled\_train\_labels”. Бројот на снимки во двете множества е прикажан на Слика 2.4.



|  |
| --- |
|  |
| Слика 2.4 Број на снимки во валидациско и тренинг множество |

**II.V) Конвертирање во grayscale и промена на димензиите на снимките**

Со повикување на функцијата “load\_and\_preprocess”, каде како аргумент се испраќа името на сите три листи (“shuffled\_train\_image\_paths”, “shuffled\_val\_image\_paths” и “shuffled\_test\_image\_paths”), се формираат три фолдери (“Training\_Site”, “Validation\_Site” и “Testing\_Site”) каде соодветно се сместуваат снимките. Функцијата вклучува и конвертирање на секоја снимка во формат grayscale, Слика 2.5.

|  |
| --- |
|  |
| Слика 2.5 Снимка во grayscale формат |

Дополнително во функцијата е искористена и промена на големината на снимките во димензии 384х384 бидејќи не сите слики во датасетот имаат иста големина. Само искористување на функцијата resize() доведува до деформација на снимките, Слика 2.6.

|  |
| --- |
|  |
| Слика 2.6 Споредба на оригинална со деформирана снимка |

За да се избегне овој проблем искористен е aspect ratio. Прво, сликата се ресајзира според нејзиниот оригинален сооднос, потоа се додава рамка (padding) за да одговара точно на target\_size, по што се зачувува или нормализира за понатамошна обработка, Слика 2.7.

|  |
| --- |
|  |
| Слика 2.7 Споредба на оригинална со снимка форматирана со aspect ratio |

**II.VI) Конвертирање во RGB**

За голем дел од моделите снимките треба да бидат во формат RGB, три канали. Заради тоа потребно е да постои и датасетот во тој формат. За таа цел направена е копија на веќе постоечките фолдери (“Training\_Site”, “Validation\_Site” и “Testing\_Site”) и тие се преименувани соодветно (“Training\_Site\_RGB”, “Validation\_Site\_RGB” и “Testing\_Site\_RGB”). За секој од фолдерите се повикува функцијата load\_and\_preprocess\_RGB со што се итерира низ секоја од снимките и секоја се се конвертира во формат RGB. Ова означува дека секој пиксел во сликата ќе добие канал со исти вредности, секој пиксел што има вредност **X** ќе се претвори во **(X, X, X)** во RGB формат Пример, доколку пикселот има вредност 128, за конвертирање во формат RGB ќе добие канал со вредности (128, 128, 128), со што се задржува оригиналната нијанса на сивата боја. Ова осигурува дека визуелниот изглед на сликата останува непроменет, само што таа сега има три канали наместо еден. Слика конвертирана во RGB е прикажана на Слика 2.9.

|  |
| --- |
|  |
| Слика 2.9 Снимка во RGB формат |

**II.VII) Конвертирање во RGB со додавање на сина нијанса**

Поради претпоставка дека додавањето на некоја боја во снимката ќе ги истакне неправилностите во снимките и со тоа ќе помогне на моделите полесно да ги класифицираат снимките додадена е сина боја на истите. За таа цел направена е копија на веќе постоечките фолдери (“Training\_Site\_RGB”, “Validation\_Site\_RGB” и “Testing\_Site\_RGB”) и тие се преименувани соодветно (“Training\_Site\_RGB\_Blue”, “Validation\_Site\_RGB\_Blue” и “Testing\_Site\_RGB\_Blue”). Со функцијата add\_blue\_tint\_to\_rgb() се изминува низ сите снимки и се додава син тинт. Функцијата add\_blue\_tint\_to\_rgb ги разделува RGB каналите на снимката, зголемува вредностите на синиот канал за 50 и потоа ги комбинира назад каналите за да создаде снимка со додаден плави тинт. Вредностите на синиот канал се ограничуваат во опсегот 0-255 за да се избегнат преголеми вредности, Слика 2.10.

|  |
| --- |
|  |
| Слика 2.10 Снимка во RGB Blue формат |

# **Модели за класификација на состојби и нивно тренирање**

Моделите за класификација на рентген снимките се клучен дел од оваа семинарска задача. Тие се алгоритми (математички функции) кои го овозможуваат препознавањето на мозочните патологии (доколку ги има) во снимката и истите ја изнесуваат претпоставката. Во оваа семинарска работа беа искористени 4 различни невронски мрежи и тоа:

* Обична длабока невронска мрежа;
* Oбична конволуциска невронска мрежа;
* Visual Geometry Group 16 (VGG16) конволуциска невронска мрежа;
* Трансформер.

Моделите се преземени од библиотеките TensorFlow и Keras, библиотеки за креирање и тренирање невронски мрежи.

Додека се тренира моделот, треба да се обрне внимание на точностите на тренинг и валидациските снимки. Моделот може да претпри заситување доколку точноста на тренинг податоците е висока а валидациската точност е доста пониска. Како последица, моделот може да има лоши резултати.

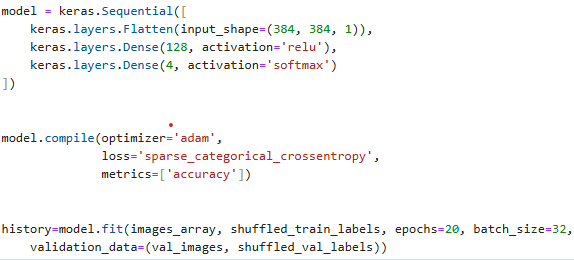
Пред тренирање на моделот, потребно е да се испитаат перформансите на истиот. За таа цел, од податочното множество се испраќаат издвоените податоци за тестирање и со нив се проверува точноста на моделот.

**III.I) Oбична длабока невронска мрежа**

Првиот искористен модел е обична длабока конволуциска невронска мрежа. Овој модел содржи 3 слоеви и тоа: влезен слој во кој влезната матрица од вредностите на пикселите на снимката ја претвара во еднодимензионална низа од истите со 384x384 елементи, еден скриен слој со 128 неврони кои се активираат со ReLU активациска функција и еден излезен слој со четири неврони со SoftMax активациска функција (Слика 3.1.1).

Моделот ги оптимизира своите параметри според „Adam“ алгоритмот, а како мерило за неговите перформанси се користи точноста на истиот.

Тренирањето се одвива во дваесет епохи, а во секоја епоха се зема по 32 примероци („batch size“). На моделот се испраќаат снимките сместени во низи(NumPy array), и тоа: “images\_array”, “val\_images”, “testing\_images” со соодветните лабели “shuffled\_train\_labels”, “shuffled\_val\_labels”, “shuffled\_test\_labels”.



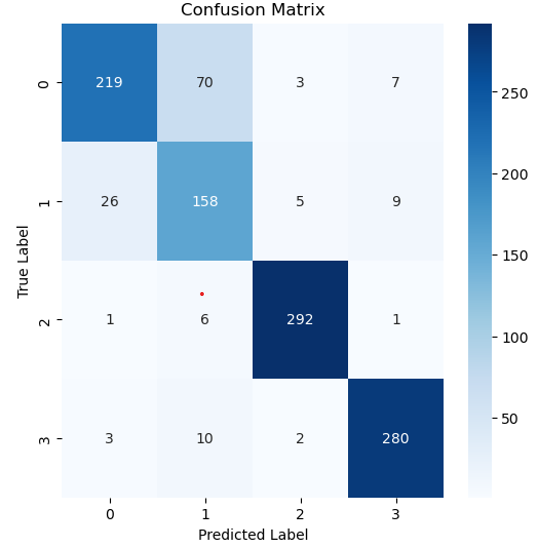
Слика 3.1.1: Обична длабока невронска мрежа

На Слика 3.1.2 се прикажани графиците кои го отсликуваат менувањето на тренинг и валидациските точности. Од нив се забележува континуиран раст на точностите при изминување на епохите.



Слика 3.1.2 Графици на точностите

За да се добие реалната слика за точноста на моделот, потребно е да се разгледа конфузионата матрица на истиот. Конфузионата матрица е матрица која посочува кои класи се точно погодени и ги истакнува оние кои се погрешно предвидени. Слика 3.1.3.



Слика 3.1.2 Конфузиона матрица на DNN

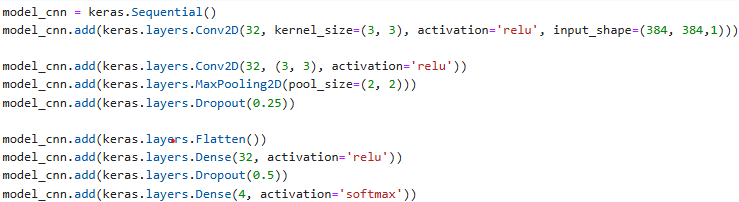
Во конкретниот пример се забележува дека моделот најмногу ги греши снимките во кои е присутен глиом и менингиом меѓусебно. Причината за тоа е сличноста меѓу самите фотографии. Голем дел од снимките кои припаѓаат на глиом и оние на менингиом изгледаат доста слично. Другите состојби се соодветно предвидени. Точноста на овој модел изнесува 0.87.

**III.II) Конволуциска невронска мрежа**

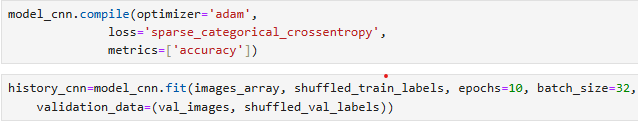
Конвулуцисканевронскамрежа **(**Convolutional Neural Network**)** е специјален вид на невронска мрежа која е најчестокористеназаобработканасликиивидео записи. Тие автоматски учат карактеристики (features) од податоците, што ги прави многу ефикасни за компјутерски вид (computer vision) задачи.

Основните компоненти на CNN се конволуцискислоеви, кои извлекуваат карактеристики од сликите со помош на филтри (kernels) кои конволуираат низ сликата и ги истакнуваат истите, пулингслоеви, кои ја редуцираат димензионалноста и истовремено ги зачувуваат главните карактеристики на сликата , и целосноповрзани(Dense) слоеви, кои вршат финална класификација. ReLUактивацијата се користи за нелинеарност, а SoftmaxилиSigmoid за одлука во класификацијата. Овие компоненти заедно овозможуваат CNN да препознава сложени објекти во сликите.

На Слика 3.2.1 е поставена конволуциска невронска мрежа која ќе ги класифицира снимките во соодветните категории. Оваа мрежа е формирана од еден влезен слој со 32 филтри (kernels) со димензии 3x3, влезен податок со димензии 384x384 и ReLU активациска функција. Следниот слој повторно има 32 филтри со истите димензии и истата активациска функција. Потоа следи пулинг слој кој ги редуцира димензиите сликата со 2x2 филтер (ја зема макимална вредност од елементите). На крај се додава целосно поврзана невронска мрежа со 3 слоја која ги предвидува една од четирите категории во која може да припаѓа снимката. Во структурата на мрежата се додаваат и „испаѓачки” функции кои ги поставуваат вредностите на дел од невроните на нула за да се избегне презаситување на моделот. Начинот на оптимизација и метриката за мерење перформанси е иста како на обичната невронска мрежа. Моделот се тренираше во 10 епохи со 32 примероци по епоха (Слика 3.2.2). На моделот се испраќаат снимките сместени во низи(NumPy array), и тоа: “images\_array”, “val\_images”, “testing\_images” со соодветните лабели “shuffled\_train\_labels”, “shuffled\_val\_labels”, “shuffled\_test\_labels”.

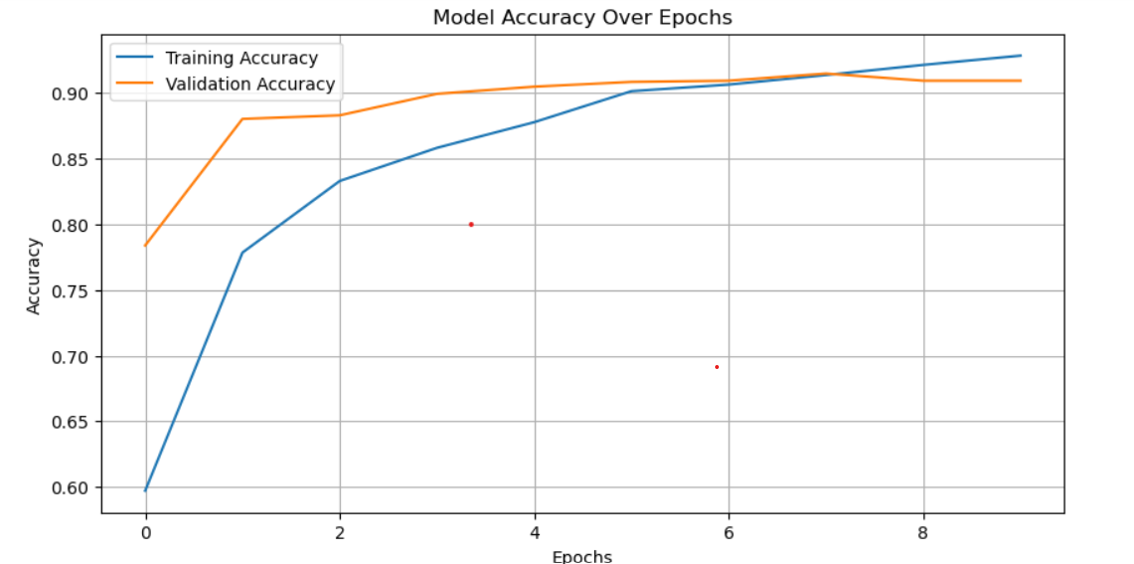


Слика 3.2.1: Конволуциска невронска мрежа



Слика 3.2.2 Тренирање на моделот

Од графикот на Слика 3.2.3 се забележува брз пораст и конвергенција на точностите при изминување на една епоха. Точноста на моделот изнесува 0.92.



Слика 3.2.3: Графици на точностите на CNN

Од конфузионата матрица (Слика 3.2.4) се забележува подобрување при препознавање на глиом во снимките. За разлика од претходниот модел, овој модел има потешкотии при препознавање на менингиом бидејќи истиот погрешно ги класифицира снимките како глиом. Во остатокот од предвидените категории нема значителни грешки.



Слика 3.2.4: Конфузиона матрица на CNN

**III.III) Transfer Learning, VGG16 and Fine Tuning**

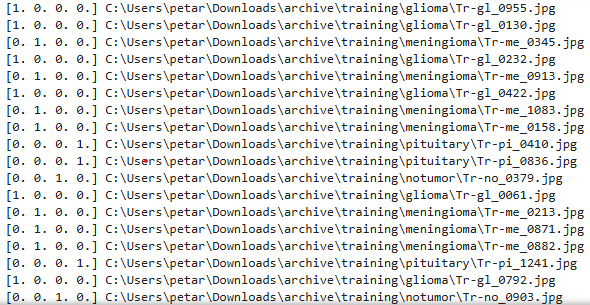
Преносно учење (Transfer learning) е техника во машинското учење каде што моделот што е развиен за еден тип задачи се користи како почетна точка за нов модел на друг тип задачи. Наместо да се тренира модел од почеток, стекнатото знаење при решавањето на еден проблем се пренесува на нов, сроден проблем. Ова е особено корисно кога има пристапот до податоци за новиот проблем е ограничен, бидејќи претходно обучениот модел може да ги искористи научените карактеристики за подобрување на перформансите на новиот проблем. Оваа техника е широко користена во длабокото учење, особено во области како што се обработката на слики и процесирање на природен јазик (Natural Language Processing).

VGG16 е типичен пример на Transfer Learning модел. Истрениран е за датасет за класификација на слики во 1000 категории, меѓутоа служи како добра основа за тренирање на модели кои ќе предвидуваат состојби за проблеми од различна природа. Крајните слоеви на овој модел можат да се отстранат и на нивно место да се прилепат нови слоеви кои ќе овозможат решавање на новиот проблем. Во овој случај, последниот слој е отстранет и на негово место се додадени два слоја со 128 неврони и 4 неврони за класификација соодветно (Слика 3.3.1).



Слика 3.3.1: VGG16 модел

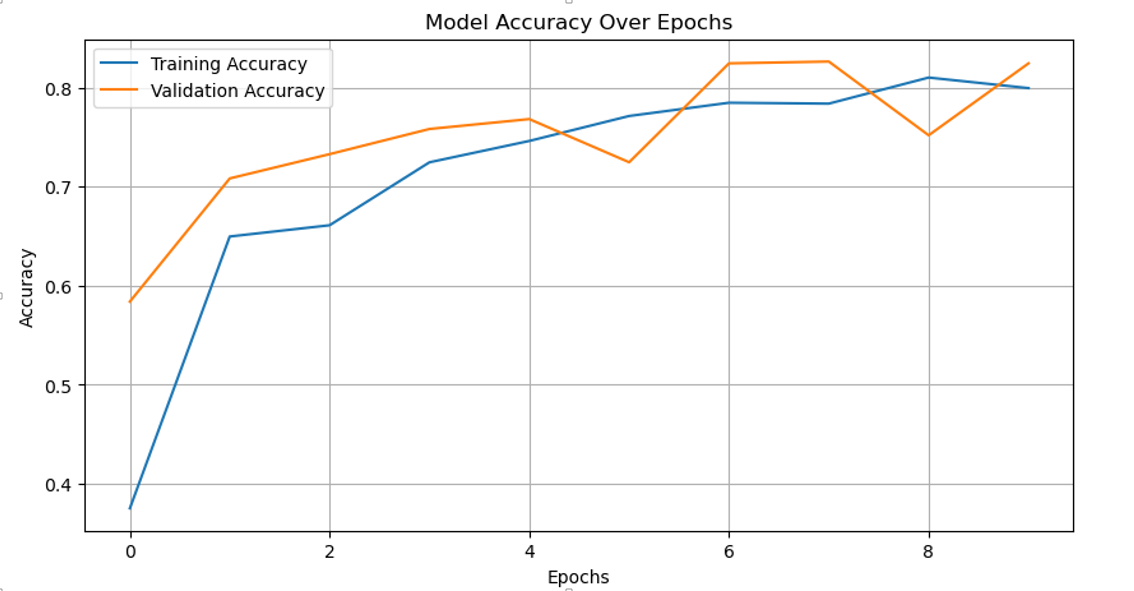
Дополнително, додаден е код за конверзија на лабелите со „One-Hot-Encoding“ во лабели со четири вредности кои ги ознчуваат припадностите на соодветните класи. Ова е потребно бидејќи моделот ги бара излезните вредности во овој формат (Слика 3.3.2).



Слика 3.3.2: „One-Hot-Encoding“ лабелирање на снимките, доколку една снимка припаѓа на дадена категорија, вредноста на таа категорија станува 1, а другите 0.

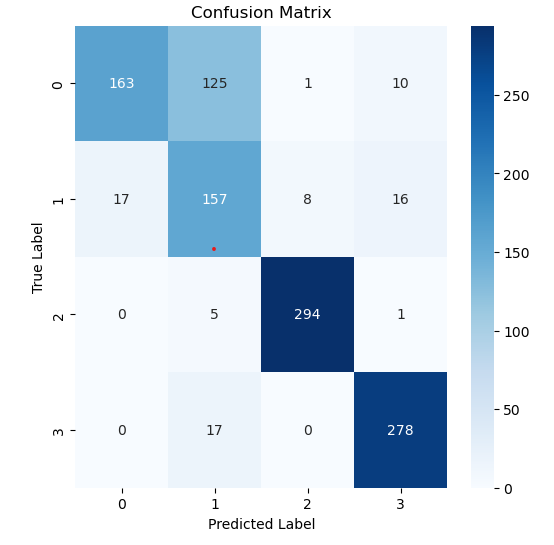
Тренирањето на моделот се одвива на истиот принцип како конволуциската невронска мрежа опишана погоре, меѓутоа влезните снимки треба да се конвертираат во RGB формат.

При тренирање на моделот се забележува пораст на тренинг и валидациската точност на моделот. По шестата епоха моделот покажува мала нестабилност. Меѓутоа нема голема дискрепанца во точностите, моделот покажува добри резултати (Слика 3.3.3). Крајната точност изнесува 0.8.



Слика 3.3.3: Графици на точностите на VGG16

Според конфузионата матрица на Слика 3.3.4, моделот има потешкотии да ги препознае снимките во кои има менингиома. Другите класи ги предвидува со голема прецизност.

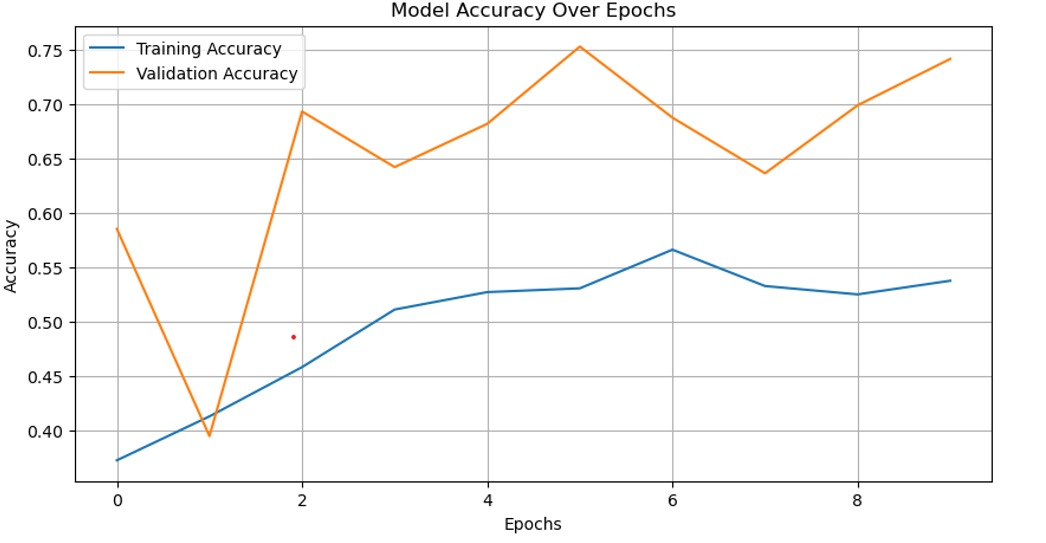


Слика 3.3.4: Конфузиона матрица на VGG16

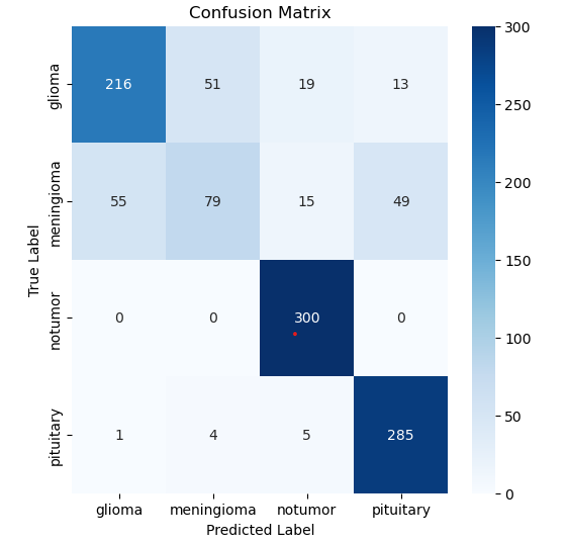
**III.IV) Fine-Tuning на VGG16**

Бидејќи на овој модел потребни му се 12 часа за целосно тренирање, дел од конволуциските слоеви во основниот модел ќе бидат „замрзнати“ (тежините во тие слоеви нема да може да се менуваат при тренирање). Оваа постапка се вика Fine-Tuning на моделот.

На Слика 3.4.1 дадени се графици за тренинг и валидациските точности на VGG16 со „замрзната“ конволуциска база. Се забележува дека валидациската точност е поголема од тренинг точноста. Тоа се должи на фактот дека тренинг множеството поминува низ Data Augmentation процедури и сликите може да претрпат дисторзија.



Слика 3.4.1: График на тренинг и валидацика точност



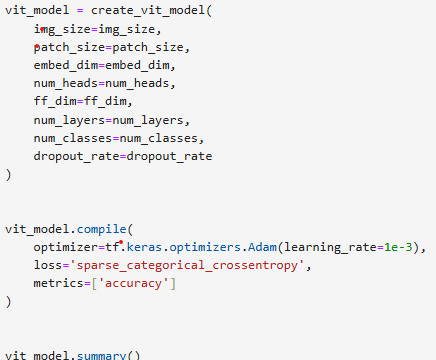
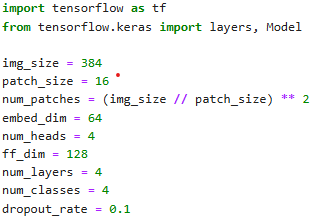
Слика 3.4.2: Конфузиона матрица на Fine-Tuned VGG16

Од Слика 3.4.2 може да се забележи дека моделот греши при предвидување менингиом со 119 промашувања. Незанемарливи грешки се забележуваат и при предвидување на глиом, речиси 83 снимки се погрешно класифицирани.

**III.V) Трансформери за компјутерски вид**

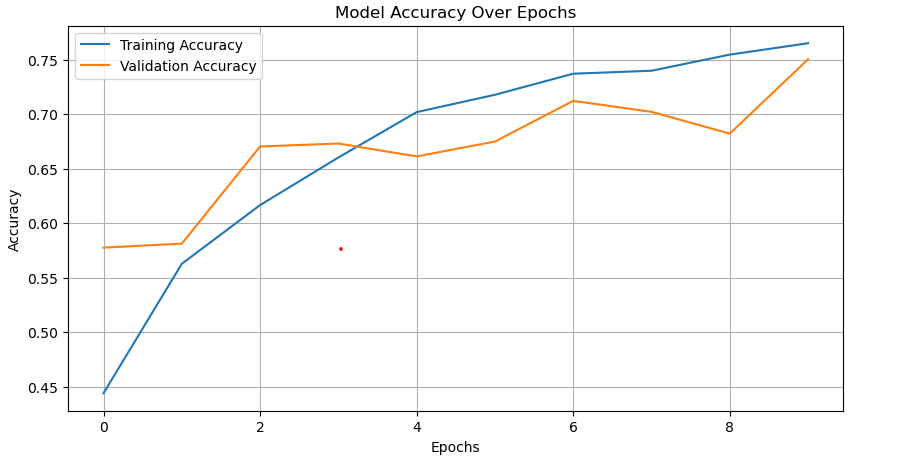
Vision Transformers (ViTs) користат self-attention за обработка на слики, наместо класични конволуции како CNNs. Тие ги делат сликите на мали парчиња (patches)**,** ги претвораат во вектори и анализираат како тие парчиња се поврзани меѓусебно. Ова им овозможува подобро да ги разберат долгорочните зависности во сликите, за разлика од CNNs кои главно гледаат локални карактеристики. За разлика од CNNs, ViTs немаат вградени претпоставки за структурата на сликите, туку учат директно од податоците**.** Иако нудат подобра глобална анализа, ViTs бараат големи количини на податоци и компјутерски ресурси за да постигнат високи перформанси. Тие се користат во класфикација на слики, детекција на објекти, медицинска анализа и автономни возила. На големи податочни множества ViTs можат да ги надминат CNNs, но за помали сетови CNNs често се поефикасни. Развојот на оптимизирани модели како Swin Transformer и DeiT помогна да се намалат овие ограничувања и да се подобри нивната употребливост.

Кодот на Сликата 3.5.1 имплементира едноставен Vision Transformer (ViT) модел преземен од TensorFlow и Keras. Прво, снимките се делат на мали „печови“ (patches) кои се претвораат во вектори преку „PatchEmbedding“ слојот. Трансформер блоковите се состојат од „attention head“ механизам на повеќекратна глава (multi-head) за внимание и feed-forward мрежа со нормализација и испуштање (dropout) за регуларизација. Позиционите вгнездувања (positional embeddings) се додаваат за да се зачува редоследот на печовите. По неколку трансформер слоеви, резултатите се компресираат преку глобално просечно групирање (GlobalAveragePooling1D), и на крај се додава класификациски слој за предвидување на класите. Моделот е извршен со Adam оптимизатор и крос-ентропија загуба за класификација. Оваа структура овозможува високо ефективна анализа на слики со минимално претходно процесирање.

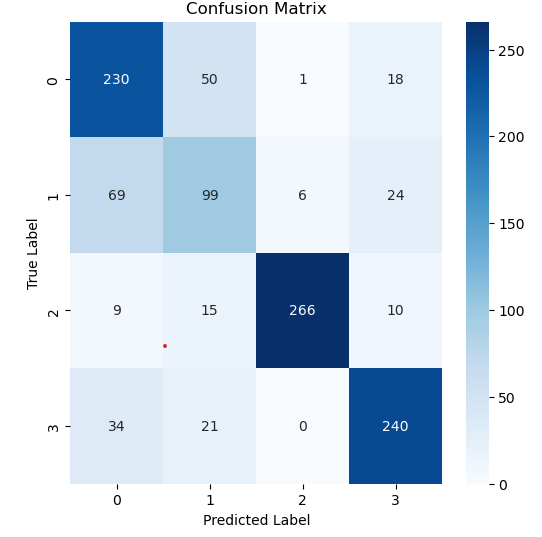


Слика 3.5.1:Трансформер модел

За да се испита точноста на моделот истиот се повикува со истите слики но во два различни формати. Првиот пат моделот работи со сликите во RGB формат со сиви канали, сместени во листи (“images\_array\_rgb ”, “val\_images\_rgb” и “testing\_images\_rgb”). Потоа истиот модел работи со RGB формат со додадена сина нијанса ( “training\_sini”,”validation\_sini”,”testing\_sini”).



Слика 3.5.2: График на тренинг и валидацика точност



Слика 3.5.3: Конфузиона матрица на Трансформер, сиви

Од Слика 3.5.3 може да се види дека моделот греши при предвидување менингиом со 99 промашувања. Незанемарливи грешки се забележуваат и при предвидување на глиом, речиси 69 снимки се погрешно класифицирани. Точноста на овој модел е 0.75.

|  |
| --- |
|  |
| Слика 3.5.4: Тренинг множество на Трансформер, сини |

Второто стартување на моделот со RGB слики е неуспешно. Тоа може да се забележи на Слика 3.5.4. каде точноста на валидациското и тренинг множеството е само 25%. Тоа се случѕва поради појава на шум кај сликите поради додадањето на сина нијанса. Што докажува дека додавањето на боја не му помага на моделот полесно да ги препознае разликите во класите.

# **IV. Заклучок**

Според резултатите од сите модели се забележува дека конволуциските модели даваат најдобри резултати, Слика 4.1 . Тоа е така бидејќи тие модели детално ги анализираат сликите со филтрите вградени во истите. Исто така голема улога има и оригиналната боја на сликите, црно-бела со која и работат овие модели. Како напомена, треба да се нагласи дека при делење на датасетот на подмножества за тренирање, валидација и тестирање, влечењето на снимките се прави по случаен избор, што придонесува до различни вредности на точностите при тренирање и тестирање.

|  |
| --- |
|  |
| Слика 4.1 Споредба на точноста на сите модели |

Оваа семинарска работа е еден од примерите што покажува дека машинското учење наоѓа голема примена во медицината, особоено во диферцијалната дијагностика. Овој и слични примери од литературата покажуваат дека со добар датасет може да се добијат модели за класификација со висока точност, а со тоа да го намалат времето на поставување на дијагнозата. Но сепак во медицината сеуште преовладува ставот дека второто мислење односно последниот збор го имаат докторите и дека овие алатки служат само како помагала.

# **VI. Референци**

1. [Kaggle](https://www.kaggle.com/datasets/cm037divya/pcos-dataset) - https://www.kaggle.com/datasets/masoudnickparvar/brain-tumor-mri-dataset
2. Scikit learn