



СОФИЙСКИ УНИВЕРСИТЕТ „СВ. КЛИМЕНТ ОХРИДСКИ“
ФАКУЛТЕТ ПО МАТЕМАТИКА И ИНФОРМАТИКА

КУРСОВ ПРОЕКТ
ПО
РАЗМИТИ МНОЖЕСТВА И ПРИЛОЖЕНИЯ

Тема:

Сегментация на изображения от ЯМР скенер за откриване
на туморни образувания в мозъка

Студент:

Семир Балджиев, 0MI3400731, ИИОЗ

София 2026 г.

1. Въведение

Ядрено-магнитният резонанс (ЯМР) е широко приложим и използван метод за детайлна визуализация на тъкани и органи в човешкото тяло. Поради това му свойство, той става един от препоръчителните методи за изследване на главния мозък. При използването на ЯМР диагностика за изследване на човешкия мозък се генерират огромен обем данни, което поставя предизвикателства пред рентгенолозите. Ръчната сегментация на туморни образувания е трудоемка и изискваща време задача, която също може да бъде податлива на субективните виждания на специалистите, които я извършват, което може да доведе до грешки. Автоматизираната сегментация предлага много възможности, но честно се сблъсква с ограниченията от ЯМР, като нисък контраст, съществуващ шум и други, които могат да повлияят върху ясното разбиране и сегментиране на изображението. В този случай стандартните методи, които не са насочени към използването на невронни мрежи и дълбоко обучение, а клъстериране на данните от изображенията в отделни области (клъстери), имат недостатъци, защото те твърдо асоциират конкретен пиксел към даден клъстер. Този проект се фокусира върху прилагането на размити множества/логика, като подход , който да преодолее неяснотите и ограниченията на стандартните подходи, използвайки го за по-прецизно моделиране на преходните зони.

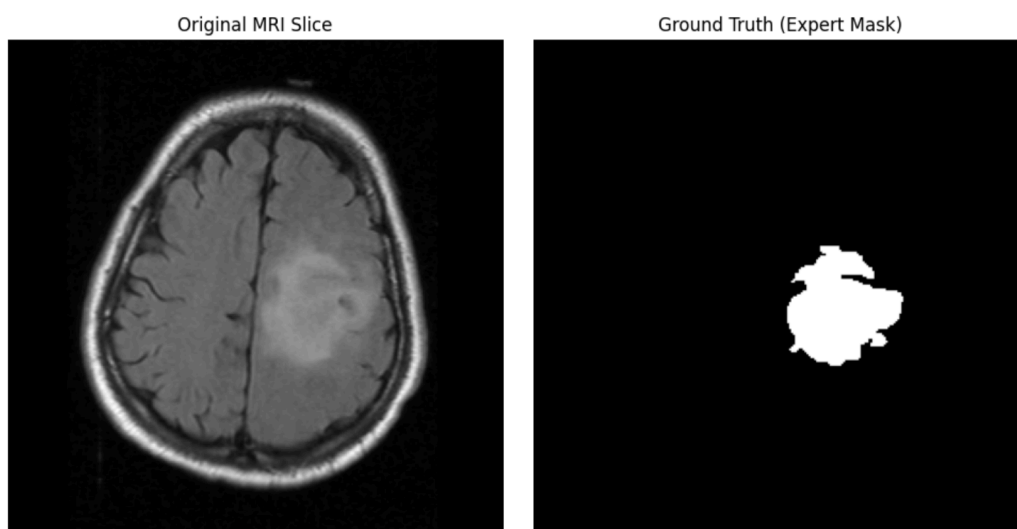
2. Формулировка на задачата

Основната задача на този проект е да се реализира алгоритъм, който автоматично да намира и сегментира туморни образувания в осеви ЯМР отрезки, които отрезки формират двуизмерно изображение на мозъка. Задачата се свежда до това, да се раздели множеството от пиксели на изображението в отделни смислово дефинирани области (клъстери). Трудността на задачата идва от факта, че туморните клетки често нараняват съседна здрава тъкан, чрез проникване и натрупване на течност, клетки или чужди вещества, което често води до подуване или възпаление на тази тъкан (едем). Това води създаване на неясна граница между туморната тъкан и здравата такава. Задачата изисква не само създаване и прилагане на алгоритъма върху данните, но и обработка на изображенията. Това ще помогне на алгоритъма като премахне някои участъци, като например черепната кост, която може да се появи със същия интензитет като на тумор. Алгоритъмът трябва да може да разграничи здрава и туморна тъкан, както и фон и едем. Освен това алгоритъмът трябва ефективно да филтрира малките шумови участъци в изображението, които често имитират туморна тъкан в здрави области от тъканта на мозъка.

3. Описание на данните

За реализацията и тестване на алгоритъма е използвано специализирано множество от данни, които са изтеглени от платформата Kaggle. Това множество от данни се използва за различни изследвания и експерименти в биомедицинската област. Данните се състоят от двуизмерни ЯМР отрезки, които са цветни но в конкретния случай за тази задача не се нуждаем от изображения с цветни канали, затова се преобразуват към степени на сивото. Изображенията са в растерен формат .tif / .tiff (Tagged Image File Format). Броя на изображенията са 7858, като всеки един запис в множеството от данни се характеризира с две изображения:

- **Оригинално изображение:** Това представлява двуизмерен срез на главния мозък на който са показани различни анатомични структури от него. Тук е важно да се спомене, че са използвани ЯМР срезове в последователност FLAIR (Fluid-Attenuated Inversion Recovery), която е характерна за визуализиране на едема и туморния регион.
- **Маска:** Това е ръчно анотирано изображения от експерти, където пикселите в региона който е тумора са в бял цвят а всичко останало в черен. Тези изображения служат за референтни стойности при оценката на алгоритъма и сравняване на сегментацията.



Фиг. 1. Примерни изображения от множеството от данни на които са показани двата типа изображения

4. Описание на алгоритъма

Процесът по сегментация на ЯМР изображение се осъществява в няколко стъпки, които комбинират различни методи за предварителна обработка на изображенията с цел премахване на шум или намаляне на интензитета на някои региони от изображението. След това се осъществява съществена част където се извършва реалната размита клъстеризация върху вече обработените изображения. Като основната цел на алгоритъма е да трансформира суровите данни за интензитетите на пикселите в структурирана информация за тъканите и различните области от анатомичната структура на мозъка.

4.1. Предварителна обработка

Преди преминаване към същинската част на алгоритъма която е сегментацията, входното изображение преминава през редица трансформации за подобряване на съотношението между сигнала и шума.

- **Gaussian Smoothing:** Прилага се филтър на Гаус за потискане на високочестотния шум, който е характерен за ЯМР диагностиката. При този филтър всеки пиксел от изображението се преизчислява се взема среднопретеглената стойност на съседните му пиксели. Като тук за разлика от обикновенното заглаждане е че стойността на съседите не е еднаква, тези които са по-далеч са с по-малка, а тези които са по-близо до центъра по-голяма. Като един вид следва нормалното разпределение. За двуизмерни изображения с каквито разполагаме формулата е следната:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

- **CLAHE (Contrast Limited Adaptive Histogram Equalization):** Тази трансформация се използва за адаптивно подобряване на локалния контраст. Това е важна стъпка, тъй като областите около тумора (едема) често имат близки нива на интензитет с този на тумора. Адаптивното изравняване на хистограмата (АНЕ) подобрява това, като трансформира всеки пиксел с функция за трансформация, получена от област в съседство. В най-простата си форма всеки пиксел се трансформира въз основа на хистограмата на квадрат, обграждащ пиксела. Извеждането на функциите за трансформация от хистограмите е точно същото като при обикновенното изравняване на хистограмата: Функцията за трансформация е пропорционална на кумулативната функция на разпределение (CDF) на стойностите на пикселите в съседство. Пикселите близо до границата на изображението трябва да се третират

специално, защото тяхната съседство не би лежало изцяло в изображението.

4.2. Клъстеризация чрез Fuzzy C-means

Това е основната част от алгоритъма която е базирана на размита клъстеризация. За разлика от стандартните подходи за клъстеризация тук всеки пиксел може да принадлежи към няколко пиксела едновременно с различна степен на принадлежност. Размитата клъстеризация има за цел да минимизира следната целева функция:

$$J(W, C) = \sum_{i=1}^n \sum_{j=1}^c w_{ij}^m \| \mathbf{x}_i - \mathbf{c}_j \|^2,$$

Където:

- X_i е стойността на пиксела
- m е параметър на размитост в случая е зададен на 2
- C_j е центъра на клъстера

Алгоритъмът итеративно обновява центровете на клъстерите и степените на принадлежност, докато не стигне стабилно състояние. В този случай алгоритъма използва 4 клъстера които са разделени смислово по следния начин: фон, здрава тъкан, туморно ядро и едем. Като резултат получаваме матрица на принадлежност, която е доста подробна и има информация за всеки пиксел с каква степен на принадлежност принадлежи към даден клъстер. В този случай ще имаме 4 стойности на принадлежност за всеки пиксел, защото сме задали 4 клъстера. По този начин получаваме не чак толкова твърди стойности, за това кой пиксел, към кой клъстер принадлежи, а моделираме по прецизно данните. Също така получаваме и координатите на центровете на клъстерите, обаче при изображения със скала на сивото каквито имаме ние, това са конкретните нива на сивото, които най-добре представят съответната тъкан или обект.

4.3. Дефъзуфикация

След приключване на размитата клъстеризация данните които имаме са от ползва, но трябва и всеки пиксел все пак да го съпоставим с някой клъстер за да може да визуализираме маската която се е получила при сегментацията. За тази цел се извършва фъзификация, като тук се прилага твърда сегментация и се избира клъстера с най-висока степен на принадлежност за всеки пиксел. След тази трансформация получаваме едно представяне в което имаме сегментация на пикселите към различните клъстери. За да видим този, който е само за тумора трябва да ги филтрираме и да покажем пикселите, които са се присъединили към този клъстер.

Въпреки предварителната обработка на изображенията може да има някои области с висок интензитет и те да бъдат присвоени към клъстера за тумор. Поради тази причина се прилага морфологичен филтър, като например Connected Components Analysis, който идентифицира всички отделни "острови" от пиксели в туморния клъстер. Това позволява да се премахнат такива учатъци които не са централни а са по периферията на изображението, защото те най-вероятно са част от черепа.

5. Резултати

За да се оцени обективно работата на алгоритъма, е необходимо използването на количествени метрики, които измерват припокриването между автоматично генерираната сегментация и аотираната маска, предоставен от експерти.

5.1. Коефициент на сходство на Дайс (Dice Similarity Coefficient - DSC)

Това е една подходяща метрика която може да се използва при такъв тип задачи защото, тя измерва пространственото припокриване между два набора от данни. Стойностите на DSC варират от 0 (липса на припокриване) до 1 (перфектно съвпадение).

$$DSC = \frac{2|X \cap Y|}{|X| + |Y|}$$

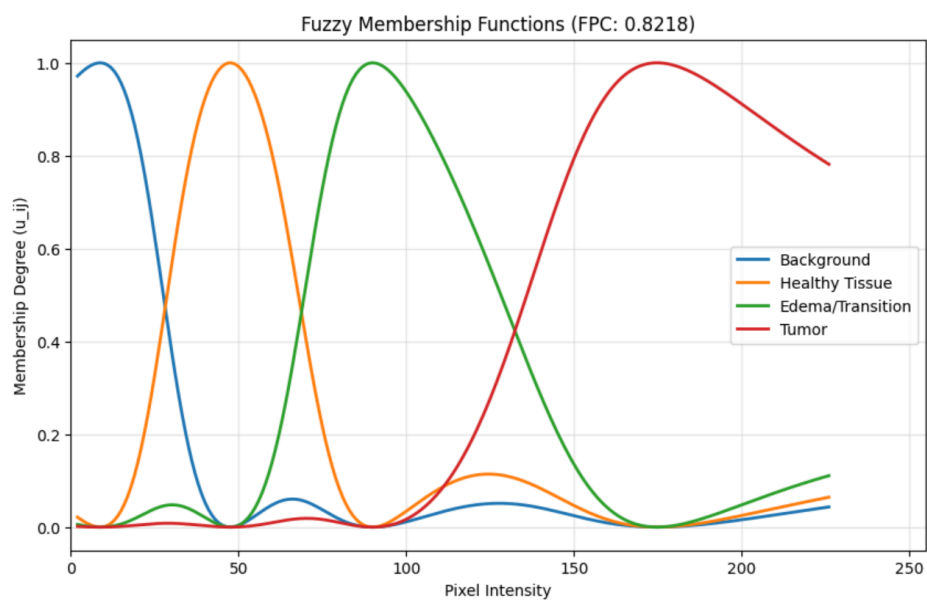
Където:

- X е маската генерирана от алгоритъма
- Y е референтната маска
- Сечението им представлява броя на пикселите кито са правилно асоциирани като части от тумора.

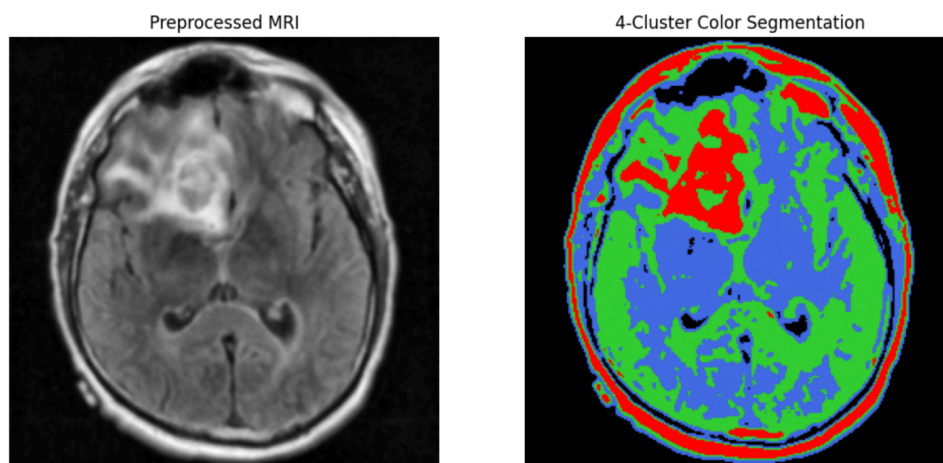
Друга метрика която сеизмерва е Fuzzy Partition Coefficient - FPC.

Тъй като използваме Fuzzy C-Means, измерваме и FPC. Тази метрика показва колко „категоричен“ е бил алгоритъмът при разделянето на данните в клъстери.

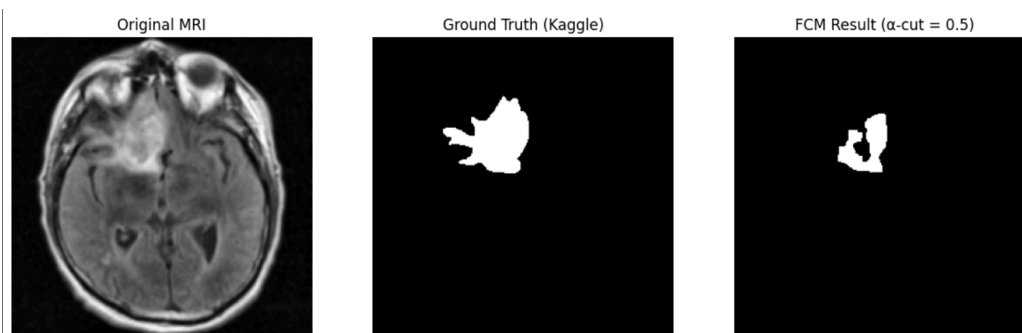
- Стойности на FPC близо до 1 означават, че клъстерите са ясно дефинирани и няма голямо разминаване между тях.
- По-ниските стойности на FPC показват, че в изображението има много „шумни“ или смесени пиксели, които алгоритъмът трудно класифицира.



Фиг. 2. Функциите на принадлежност за различните клъстери и интензитет на пиксела



Фиг. 3. Различните сегменти след клъстеризацията



Фиг. 4. Сравнение на резултата от клъстеризацията с референтната стойност

6. Програмна реализаци

Алгоритъмът е разработен на езика Python, избран поради многообразието от библиотеки за научни изчисления включително такива и за размити множества и обработката на изображения.

Основните компоненти и библиотеки които са използвани са:

- Обработка на изображенията чрез OpenCV и различни модули от него които ни позволяват да извършим предварителната обработка и прочитане на изображенията като например функциите GaussianBlur и createCLAHE, imread
- Всякакъв вид изчисления за сумиране или матрични се извършват чрез библиотеката numpy за оптимизирани и скоростни изчисления.
- Библиотеката skfuzzy която се използва за размитата клъстеризация (C means) с която се пресмята матрицата на принадлежност U.
- Библиотеката Matplotlib за визуализация на изображенията в различните етапи от изпълнение на програмата. За началното визуализиране на изображението и маската към него, както и в последствие за показване на откритите сегменти от алгоритъма.
- Библиотеките kagglehub и glob за изтегляне на множеството изображения от Kaggle и последваното им зареждане за да може да се използват от програмата.

Средата в която се изпълнява програмата и всичките тестове е Jupyter Notebook това е една среда която позволява съвместно писане на код и текст по формата на markdown, което помага за неговото документиране, и се визуализират лесно данни в тази среда, което улеснява извършването на експерименти и изпълнение на код.

7. ИСТОЧНИЦИ

1. Mateusz Buda, Ashirbani Saha, Maciej A. Mazurowski "Association of genomic subtypes of lower-grade gliomas with shape features automatically extracted by a deep learning algorithm." Computers in Biology and Medicine, 2019.
2. Dibya Jyoti Bora, and Mrinal Kanti Mishra. 2025. "Comparative Evaluation of Hard and Soft Clustering for Precise Brain Tumor Segmentation in MR Imaging". Journal of Advances in Mathematics and Computer Science 40 (9):127–141. <https://doi.org/10.9734/jamcs/2025/v40i92050>.
3. [Sci-Kit Fuzzy Documentation](#)
4. [CLAHE \(Contrast Limited Adaptive Histogram Equalization\)](#)
5. [Gaussian Smoothing](#)
6. [Dice Similarity Coefficient](#)