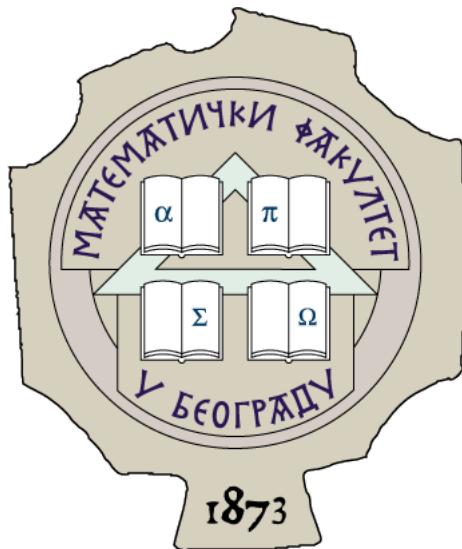


# Фази кластеровање

Маја Џрномарковић 21/2017  
Марко Бабић 77/2017

25. јун 2021.

Семинарски рад у оквиру курса  
Рачунарска интелигенција



# Садржај

<b>1 Увод</b>	<b>3</b>
1.1 Дефиниција кластеровања . . . . .	3
1.2 Врсте кластеровања . . . . .	3
1.3 Различити типови кластеровања . . . . .	4
1.4 Фази кластеровање . . . . .	4
<b>2 Сегментација слика</b>	<b>4</b>
2.1 Фази ц-средина . . . . .	5
2.1.1 Наша имплементација алгоритма фази ц-средина . . . . .	6
2.1.2 Резултати алгоритма фази ц-средина . . . . .	8
2.2 К-средина . . . . .	15
2.2.1 Наша имплементација алгоритма к-средина . . . . .	16
2.2.2 Резултати алгоритма к-средина . . . . .	17
2.3 Функција <i>threshold()</i> и представљање пиксела једном вредношћу . . . . .	23
2.4 Поређење резултата . . . . .	24
<b>3 Закључак</b>	<b>25</b>
<b>4 Литература</b>	<b>26</b>

# 1 Увод

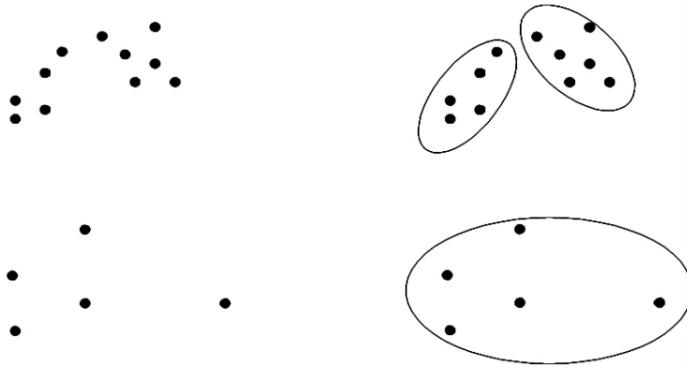
Сегментација слика је један од најпрострањенијих начина за коректно класификовање пиксела у апликацијама заснованим на одлучивању. Сегментација је техника која партиционише слику на униформне и непреклапајуће делове, засновано на некој мери сличности. Ова техника има велики број примена у анализи слика, медицинском процесирању слика, географском информационом систему, и др. Последњих година, исказано је велико интересовање за анализу слика, и с временом се ова област све више развија. Наш рад се бави проблемом сегментације слика коришћењем технике фази кластеровања. Највише пажње ће бити посвећено проблему детекције тумора на мозгу сегментовањем рендгенских снимака.

## 1.1 Дефиниција кластеровања

Не постоји формална дефиниција кластеровања. Кластер анализа је проналажење група објеката таквих да су објекти у једној групи међусобно сличнији у односу на објекте у различитим групама. **Кластеровање** се односи на поступак издвајања кластера. Проблем кластеровања се може дефинисати на следећи начин: Дат је жељени број кластера  $K$ , скуп података од  $N$  тачака и функција за мерење растојања. Потребно је пронаћи партиције скупа података тако да се минимизује вредност функције за мерење.

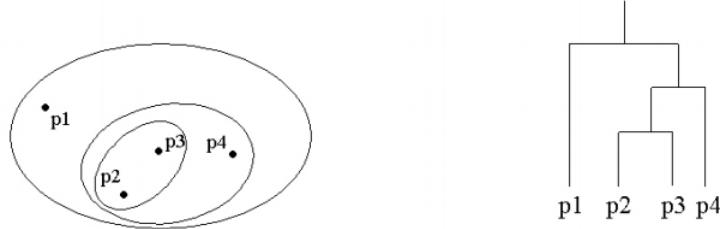
## 1.2 Врсте кластеровања

- **Партиционо кластеровање:** Подела скупа података у непреклапајуће подскупове (кластере) такве да је сваки податак тачно у једном подскупу.



Слика 1: Лево - почетни подаци. Десно - Партиционо кластеровање.

- **Хијерархијско кластеровање:** Скуп угњеждених кластера организован у облику хијерархијског стабла.



Слика 2: Лево - хијерархијско кластеровање. Десно - денгограм.

### 1.3 Различити типови кластеровања

- Ексклузивно/неексклузивно кластеровање
- Фази (расплинуто/нерасплинуто) кластеровање
- Делимично/комплентно кластеровање
- Хетерогено/хомогено кластеровање

### 1.4 Фази кластеровање

Код класичног кластеровања, подаци су подељени у одређен број дисјунктних кластера, где један елемент може припадати само једном кластеру. У фази кластеровању тачка може припадати већем броју кластера са неком тежином између нула и један. Збир свих тежина је једнак 1. Сличне карактеристике има вероватносно кластеровање. Фази кластеровање се још назива и расплинуто кластеровање.

## 2 Сегментација слика

Постоје многобройне методе и разноврсна литература за издавање информација са слике и њену поделу на различите регионе. Свака од тих метода сусреће се са одређеним ограничењима која се огледају у временској сложености или тачности. Разлог за то је што не постоје јасне границе између објекта на слици. Фази кластеровање показало се као веома добар начин за превазилажење овог проблема.

Сегментација слика коришћењем фази кластеровања била је предмет великог интересовања кроз године. Неки од алгоритама који се баве овом темом су: Фази ц-средина, Густафон-Кесе, Гаусова разградња смеше, Фази ц-сорте, Фази ц-прстенови, Прилагодљиве фази ц-сорте, Фази ц-омотачи,

Фази ц-сферни омотачи, Фази ц-правоугаони омотачи (енг. Fuzzy C-Means (FCM), Gustafson-Kesse (GK), Gaussian Mixture Decomposition (GMD), Fuzzy C-Varieties (FCV), Adaptive Fuzzy C-varieties (AFC), Fuzzy C-Shell (FCS), Fuzzy C-Spherical Shells (FCSS), Fuzzy C-Rings, Fuzzy C-Quadric Shells (FCQS), Fuzzy C- Rectangular Shells (FCRS) и други. Међу свим горе наведеним алгоритмима, метод фази ц-средина је најприхваћенији начин за сегментацију слика јер омогућава мањи губитак информација у односу на алгоритме класичног кластеровања.

У наставку ћемо приказати сегментацију слика коришћењем алгоритама к-средина (енг. k-means) и фази ц-средина (енг. fuzzy c-means).

## 2.1 Фази ц-средина

Први пут га је представио Дун а потом га је модификовао Бездек. Фази ц-средина алгоритам је итеративни алгоритам који покушава поделити скуп података у унапред задат број подгрупа(кластера) с тим што овде сваки елемент припада сваком кластеру са одређеним степеном припадности који је између нула и један.

Следи детаљан опис алгоритма:

- Улазни параметри: Подаци које желимо да кластеријемо (скуп елемената  $x$  димензије  $n$ ) и број који означава колико кластера желимо да добијемо ( $k$ ).
- Излазни параметри: Матрица припадности кластерима (димензије  $n \times k$ ) и матрица центроида кластера(димензије  $k \times d$  где је  $d$  димензија сваког појединачног елемента из скупа  $x$ ).

Кораци алгоритма:

1. Насумично додељујемо вредности за све тежине у ознаци:  $w_{ij}$ ,  $1 \leq i \leq n$ ,  $1 \leq j \leq k$  уз услове:

$$(a) \sum_{j=1}^k w_{ij} = 1, \quad \forall i \in 1, 2, \dots, n$$

$$(b) 0 < \sum_{i=1}^n w_{ij} < n, \quad \forall j \in 1, 2, \dots, k$$

2. Рачунање центроида за све кластере у ознаци  $c_j$  помоћу формуле:

$$c_j = \frac{\sum_{i=1}^n w_{ij}^p x_i}{\sum_{i=1}^n w_{ij}^p} \quad (1)$$

Напомена: ако је  $p = 0$ , имаћемо понашање класичног к-средина алгоритма.

3. Ажурирање вредности матрице припадности користећи формулу:

$$w_{ij} = \frac{\frac{1}{dist(x_i, c_j)}^{\frac{1}{p-1}}}{\sum_{q=1}^k \frac{1}{dist(x_i, c_q)}^{\frac{1}{p-1}}} \quad (2)$$

4. Понављати кораке 2. и 3. док центроиди не остану исти у две итерације за редом.

Мера квалитета кластеровања је сума квадратне грешке(енг. Sum of Squared Error):

$$SSE = \sum_{j=1}^k \sum_{i=1}^n w_{ij}^p dist(x_i, c_j), \quad p \in 1, \dots, \infty \quad (3)$$

### 2.1.1 Наша имплементација алгоритма фази ц-средина

Решење проблема сегментације слика уз помоћ алгоритма фази ц-средина смо имплементирали у програмском језику Пајтон. Пајтон библиотеке ко-ришћене у нашем решењу су:

- numpy
- matplotlib.pyplot
- os
- cv2
- time
- math

Algoritam je implementiran u функцији `fuzzy_c_means`. Ona као аргументе прима:

- *data* - низ података(to је уствари матрица која је димензије  $n \times d$ );
- *n* - цео број који означава број података које желимо да клasterujemo;
- *k* - цео број који означава број klastera;
- *d* - цео број који означава димензију pojedinačног податка из скупа по-датака које желимо да klasterujemo;
- *p* - цео број који означава параметар за фази формулу којом одређујемо степен припадности неког податка за сваки од klastera;
- *max\_iter* - цео број који означава максимални број итерација zbog bezbednosti,

dok као повратну вредност враћа:

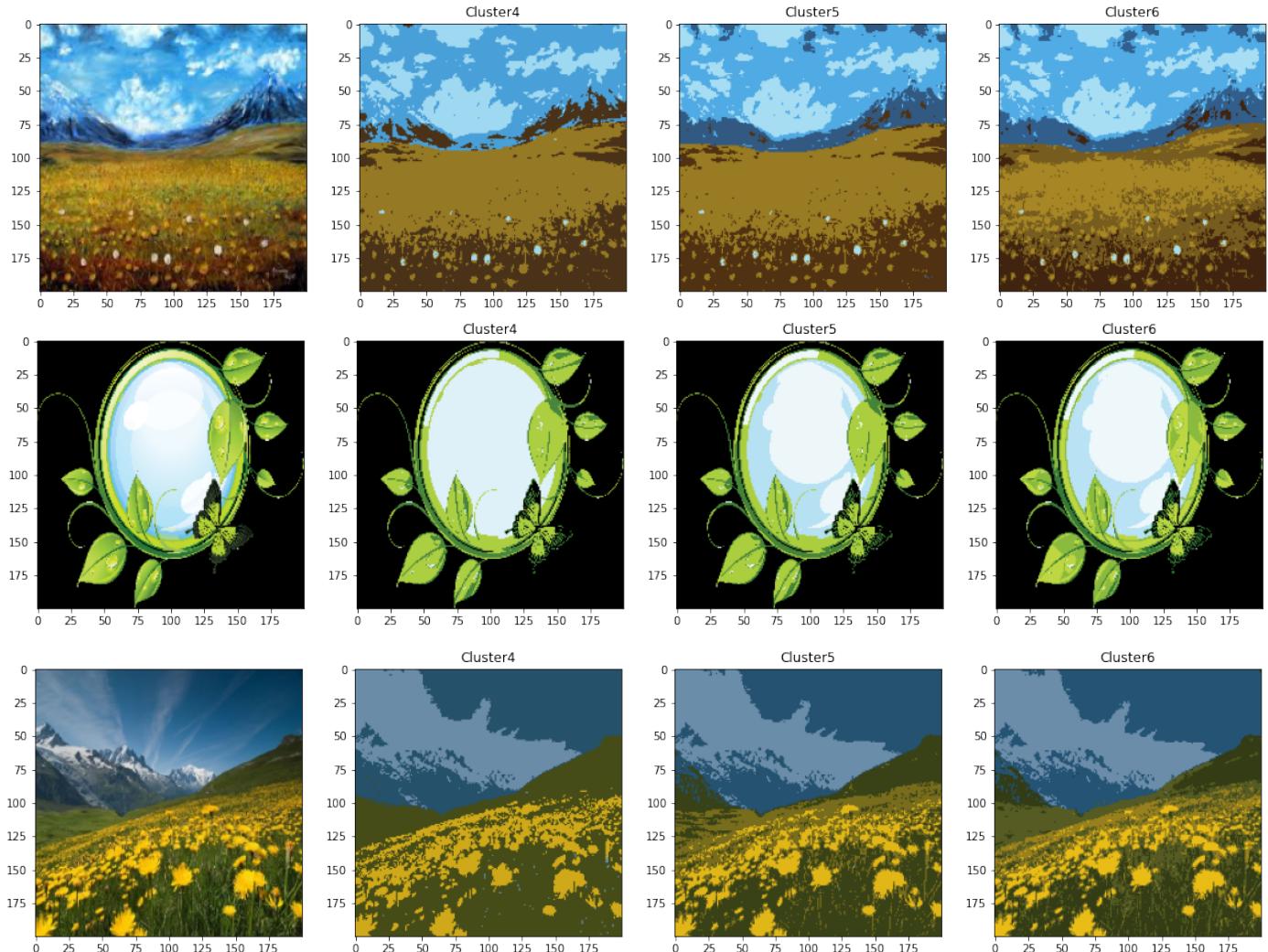
- матрицу  $n \times k$  у којој ће елемент на позицији бити  $w_{ij}$ , tj. težina са којом i-ти елемент припада j-tom klasteru;
- матрицу  $k \times d$  у којој ће се чувати centroidi svih klastera.

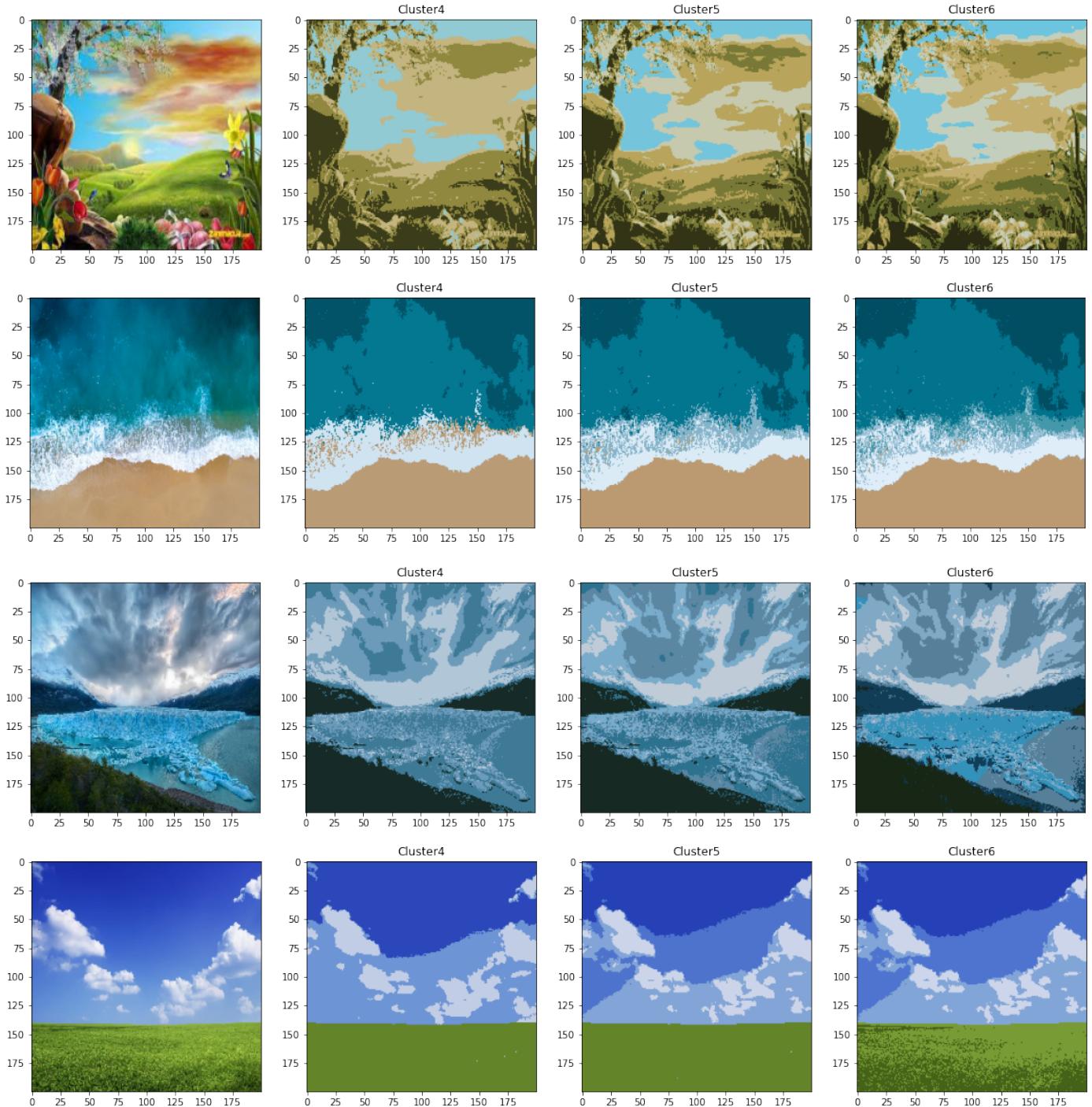
Dakle, kako je jedna slika predstavljena kao matrica piksela gde je svaki piksel dimenzije 3, nju parsiramo u niz, korišćenjem funkcije *reshape()* iz Pajton biblioteke *numpy*, kako bismo je mogli, kao argument, proslediti našoj funkciji. Nakon što naša funkcija kao povratnu vrednost vrati matricu pripadnosti podataka(u našem slučaju piksela dimenzije 3) klasterima, svakom od piksela dodeljujemo vrednost centroida klastera za koji je vrednost najveća u matrici pripadnosti. Zatim niz piksela vraćamo u matricu polaznih dimenzija korišćenjem funkcije *reshape()* i posmatramo je kao sliku.

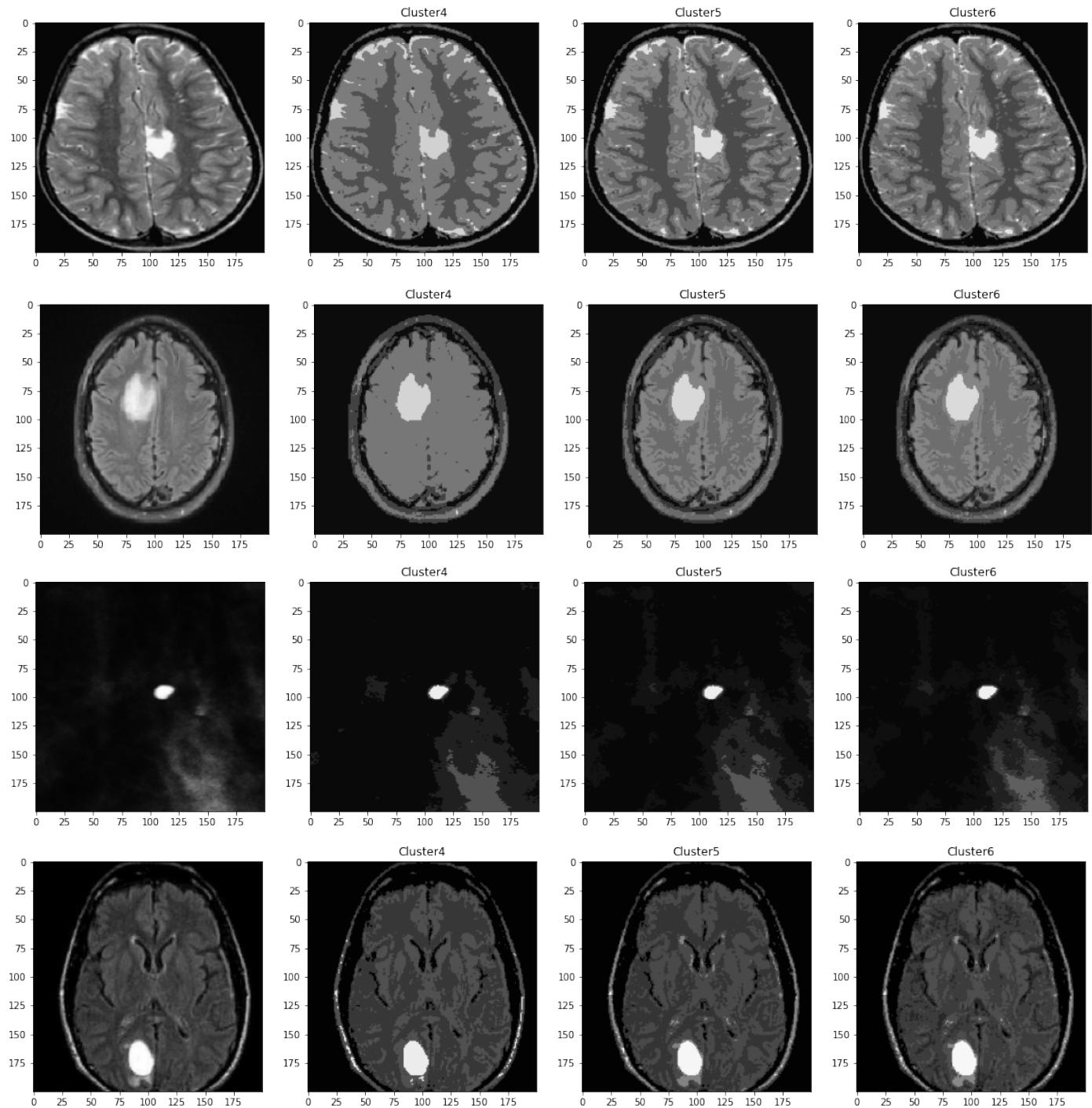
### 2.1.2 Резултати алгоритма фази ц-средина

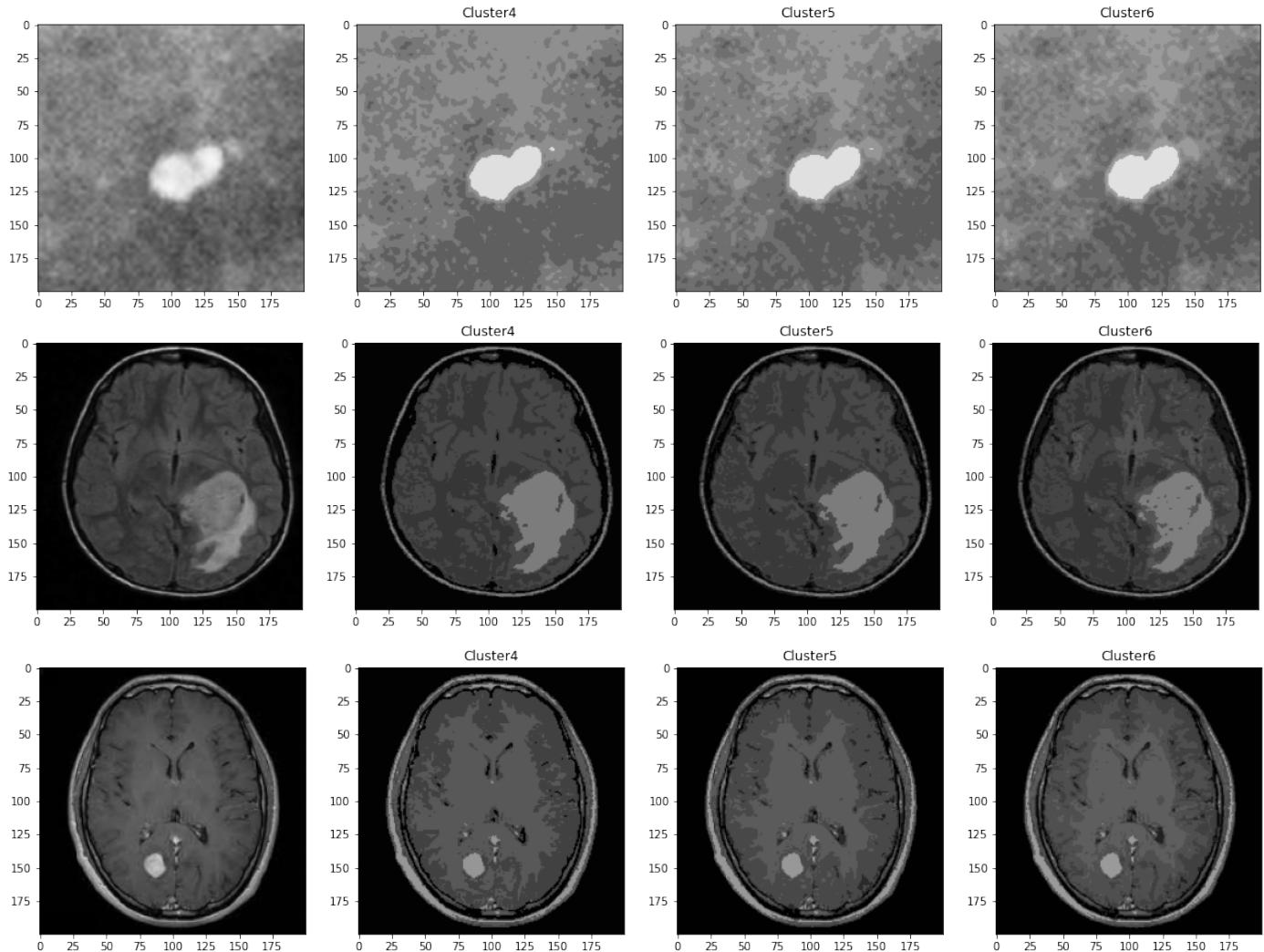
Након описаног поступка слике исписујемо коришћењем функција `subplot()` и `imshow()` из Пајтон библиотеке `cv2`.

Слике које је сегментовао наш алгоритам фази ц-средина изгледају овако:

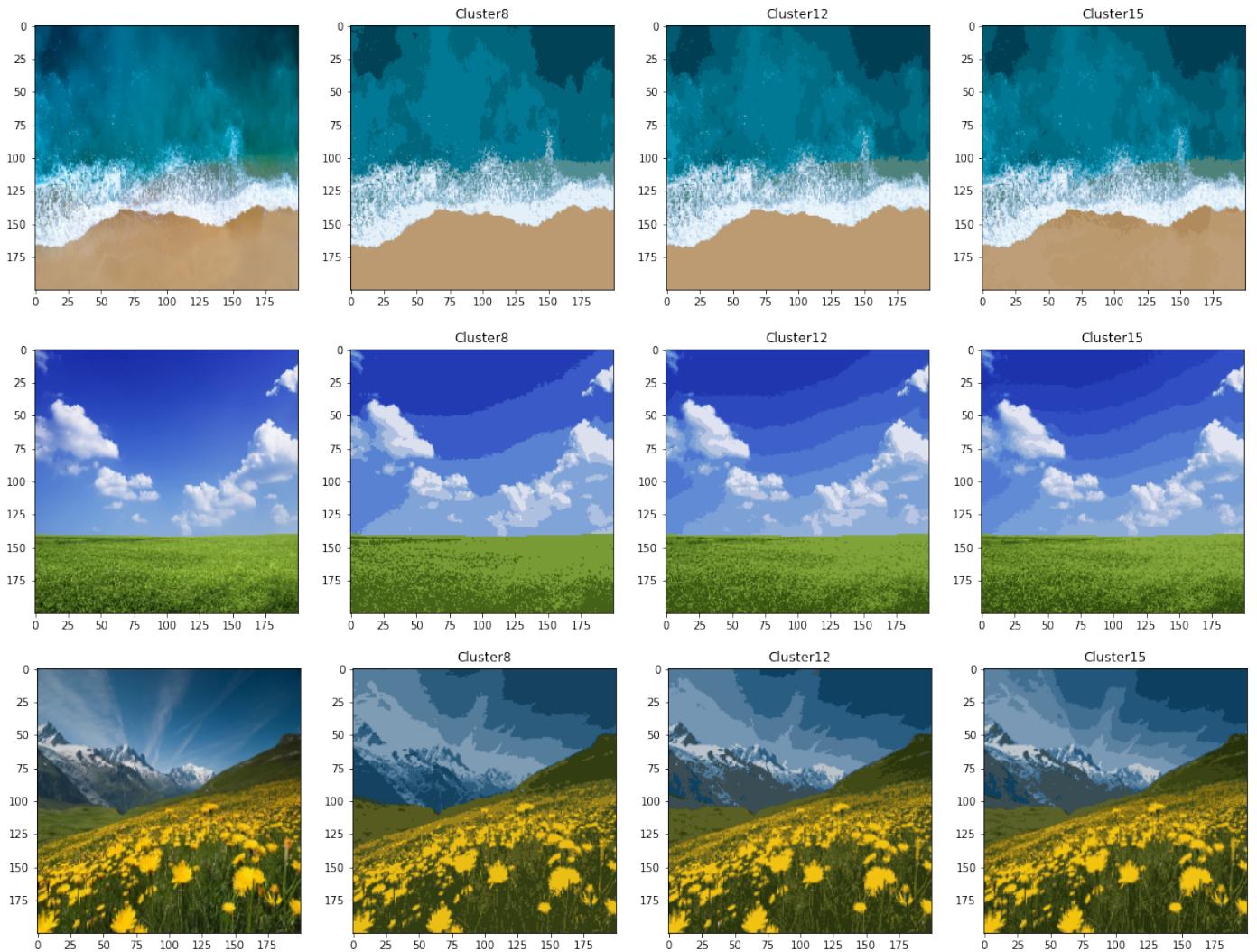




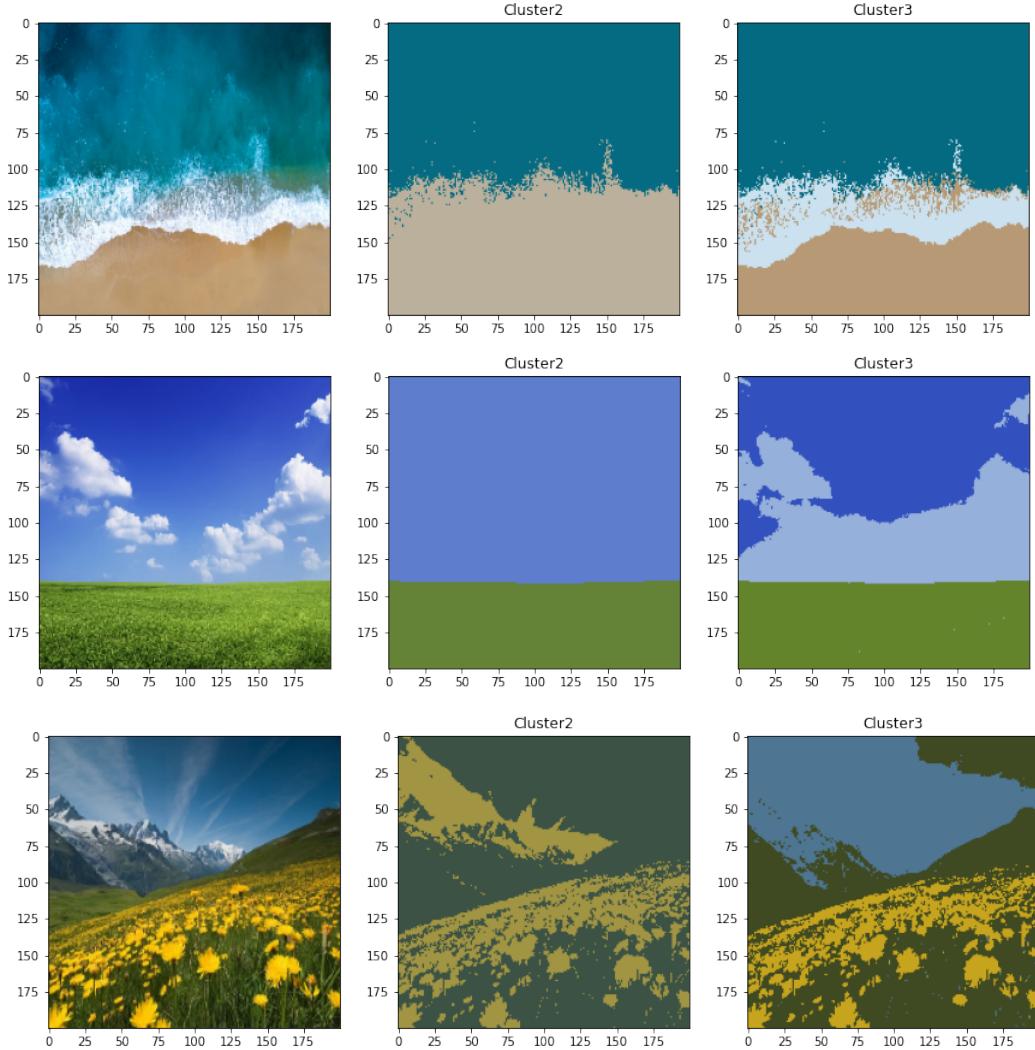




Коришћење више кластера:

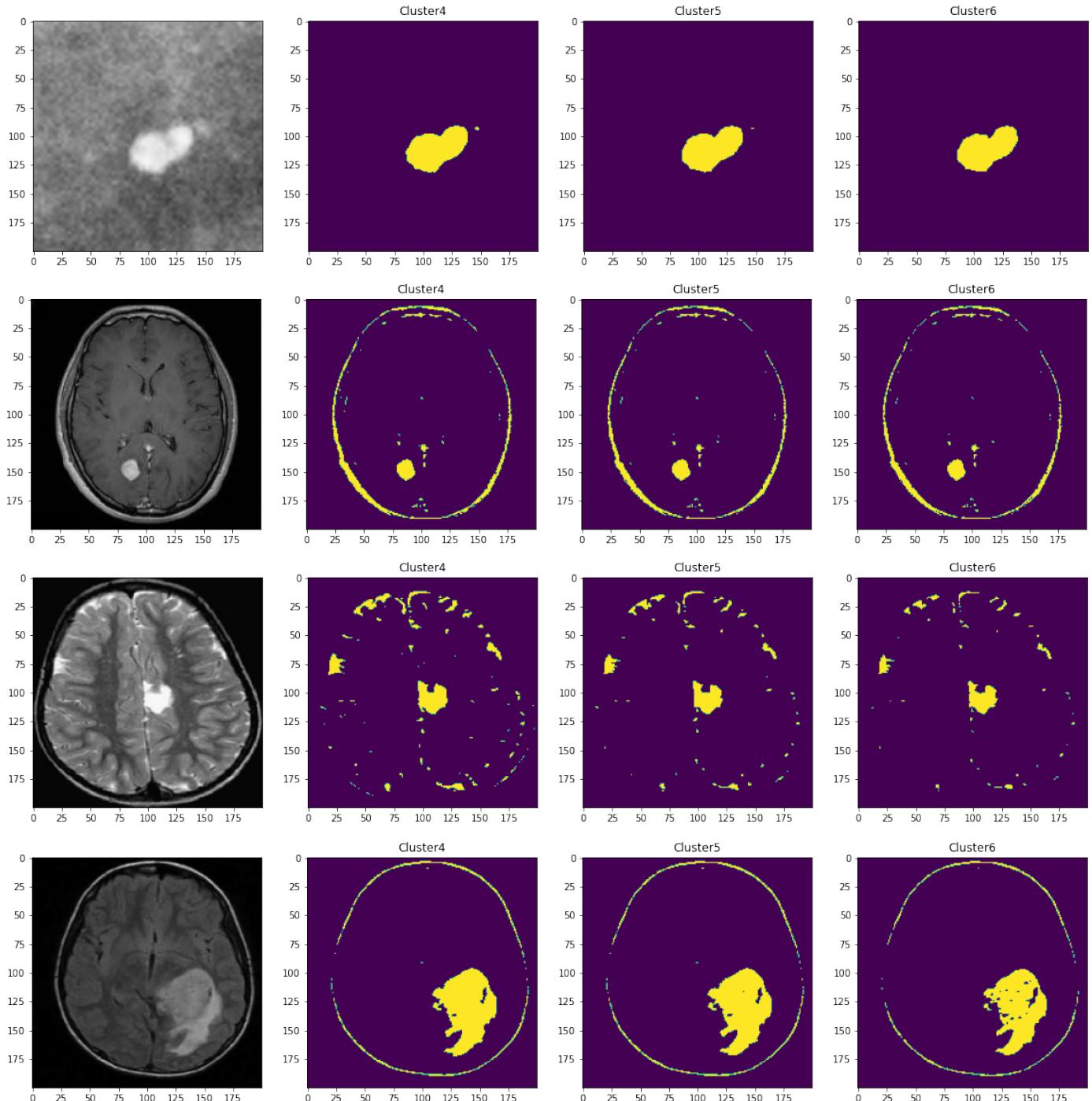


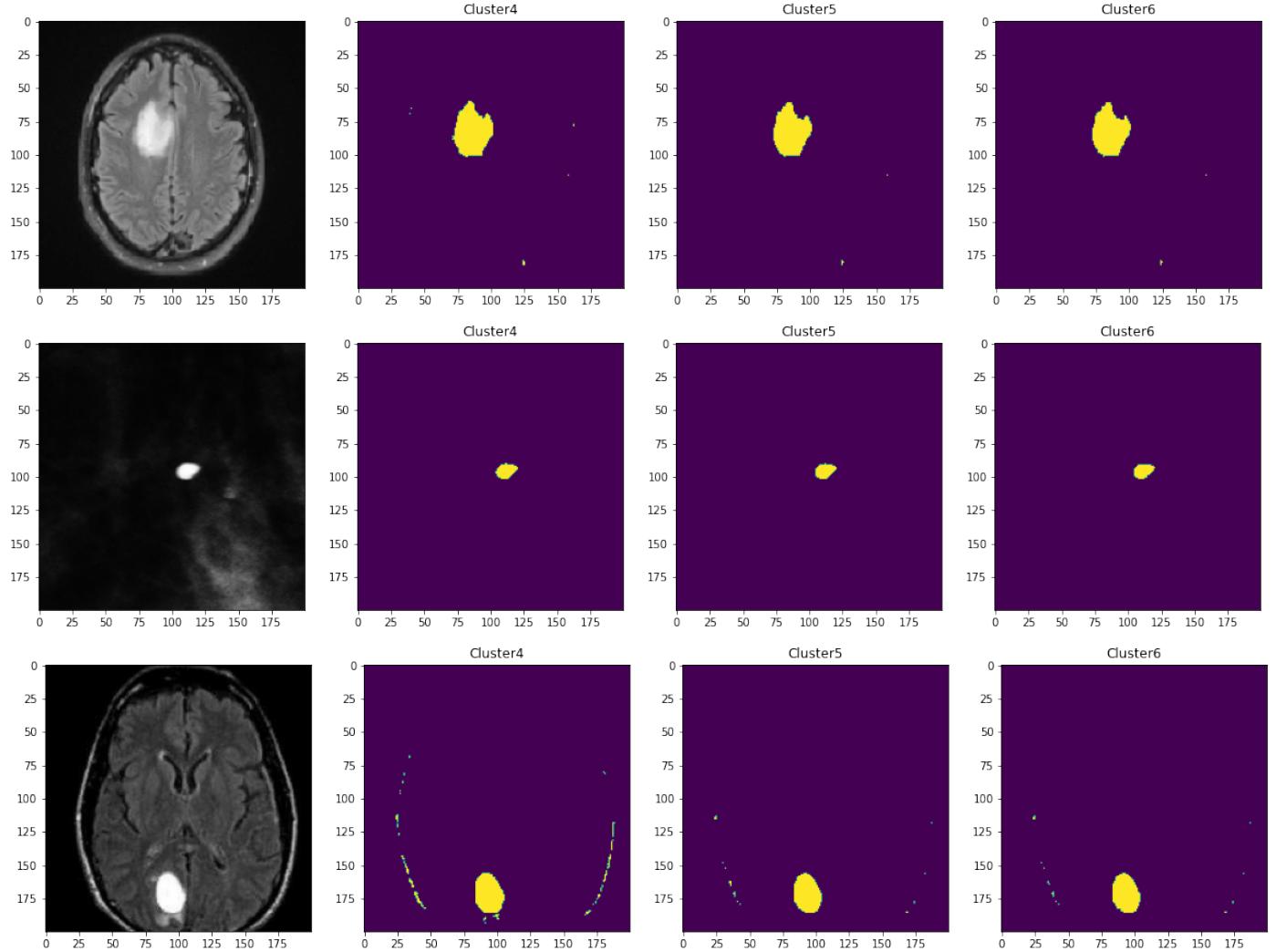
Коришћење мање кластера:



У случају рендгенских снимака мозга врло је битно да се јасно сегментује одређени детаљ на слици, односно тумор ако он постоји, и због тога након горе описаног поступка бинаризујемо слику коришћењем функције *threshold()* из библиотеке *cv2* и пикселе представљамо једном целобројном вредношћу, односно представљамо слику са само две боје. Овај поступак је детаљно описан у одељку 2.3.

Након примене нашег алгоритма фази ц-средина, функције *threshold()* и парсирања слике у матрицу пиксела који су димензије један на рендгенске снимке тумора мозга на излазу добијамо слике сегментоване на следећи начин:





## 2.2 К-средина

К-средина алгоритам је итеративни алгоритам који покушава поделити скуп података у унапред задат број засебних подгрупа(кластера).

Следи детаљан опис алгоритма:

- Улазни параметри: Подаци које желимо да кластеријемо (скуп елемената  $x$  димензије  $n$ ) и број који означава колико кластера желимо да добијемо ( $k$ ).
- Излазни параметри: Матрица припадности кластерима (димензије  $n \times k$ )

$2$ ) и матрица центроида кластера(димензије  $k \times d$  где је  $d$  димензија сваког појединачног елемента из скупа  $x$ ).

Кораци алгоритма:

1. Наводимо жељени број кластера.
2. Насумично додељујемо вредности за центроиде свих кластера у означи  $c_j$ .
3. Рачунање центроида за све кластере као аритметичке средине свих елемената у кластеру.
4. За сваки податак ажурирамо кластер ком он припада тако да припада кластеру чији центроид му је најближи.
5. Понављамо кораке 2. и 3. док центроиди не остану исти у две итерације за редом.

Мера квалитета кластеровања је сума квадратне грешке(енг. Sum of Squared Error).

### 2.2.1 Наша имплементација алгоритма к-средина

Решење проблема сегментације слика уз помоћ алгоритма к-средина смо имплементирали у програмском језику Пајтон (енг. Python).

Пајтон библиотеке које коришћене у нашем решењу су:

- *numpy*
- *matplotlib.pyplot*
- *os*
- *cv2*
- *time*
- *math*

Алгоритам је имплементиран у функцији *k\_means*. Она као аргументе прима:

- *data* - низ података(то је уствари матрица која је димензије  $n \times d$ );
- *n* - цео број који означава број података које желимо да кластеријемо;
- *k* - цео број који означава број кластера;
- *d* - цео број који означава димензију појединачног податка из скупа података које желимо да кластеријемо;
- *max\_iter* - цео број који означава максимални број итерација због безбедности,

док као повратну вредност враћа:

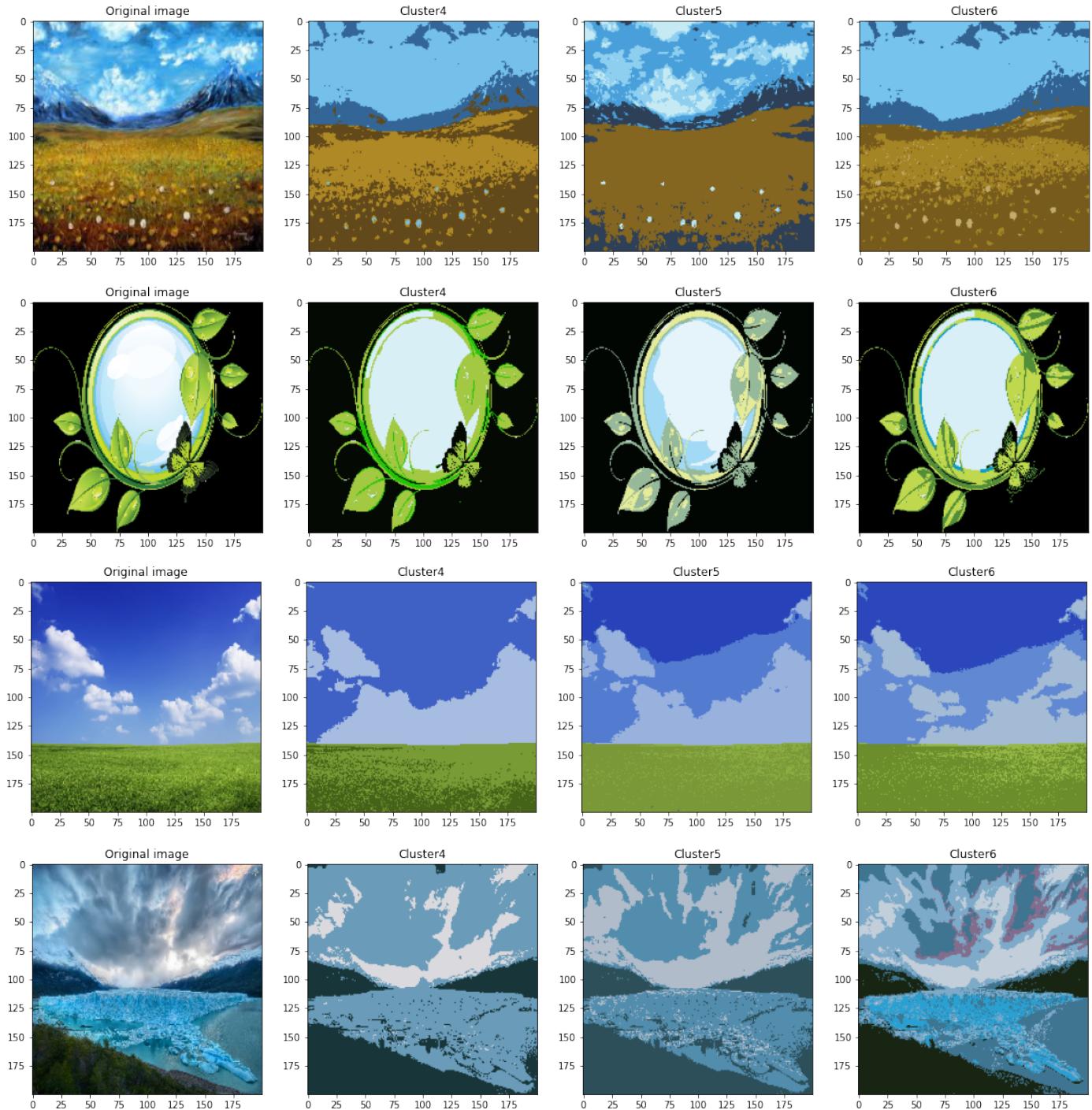
- матрицу  $n \times 2$  у којој ће се уз елемент налазити ознака кластера ком припада;
- матрицу  $k \times d$  у којој ће се чувати центроиди свих кластера.

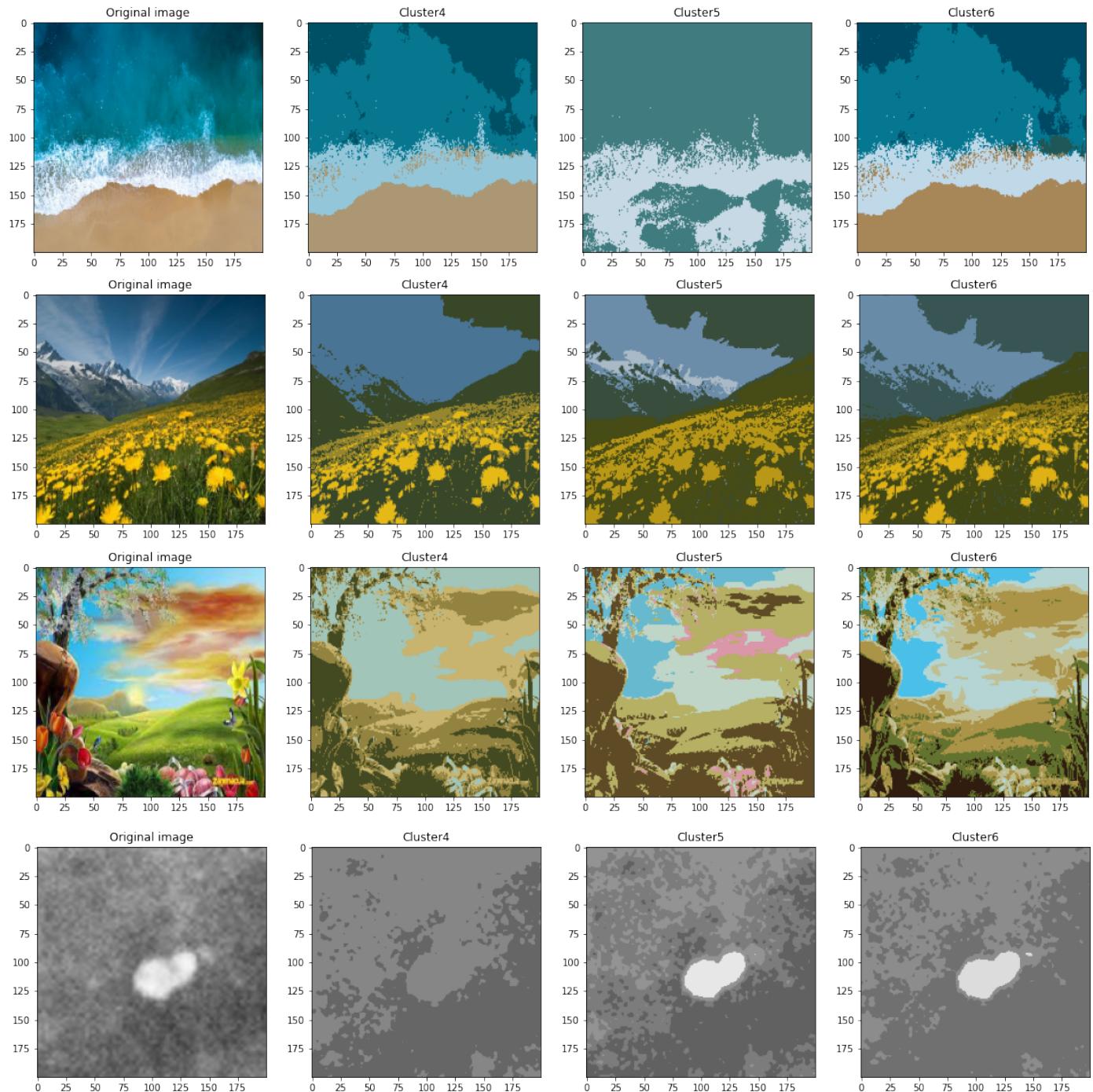
Дакле, како је једна слика представљена као матрица пиксела где је сваки пиксел димензије три, њу парсирамо у низ, коришћењем функције *reshape()* из Пајтон библиотеке *numpy*, како бисмо је могли, као аргумент, проследити нашој функцији. Након што наша функција као повратну вредност врати матрицу припадности података(у нашем случају пиксела димензије 3) кластерима сваком од пиксела додељујемо вредност центроида кластера ком припада. Затим низ пиксела враћамо у матрицу полазних димензија коришћењем функције *reshape()* и посматрамо је као слику.

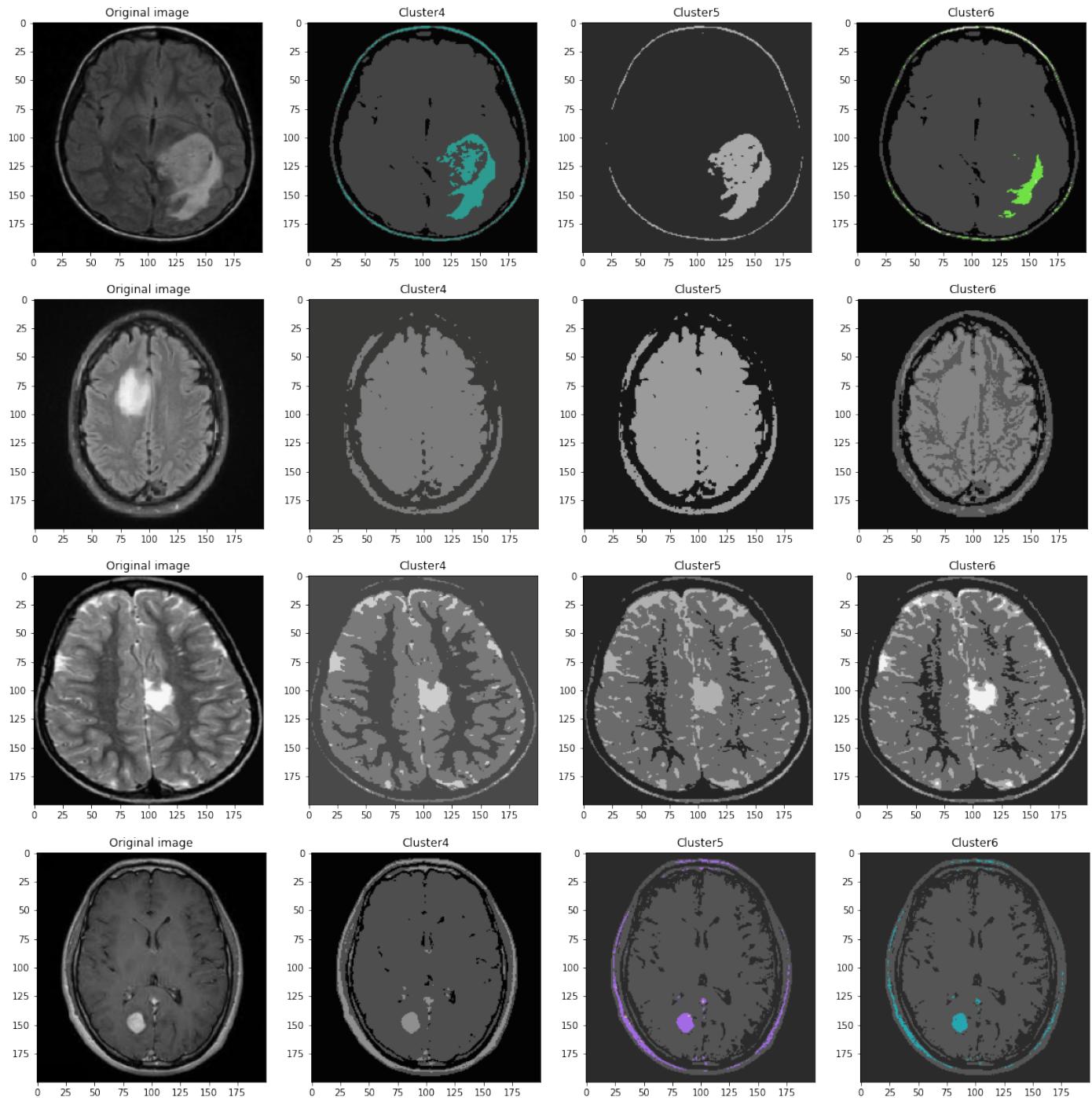
### 2.2.2 Резултати алгоритма к-средина

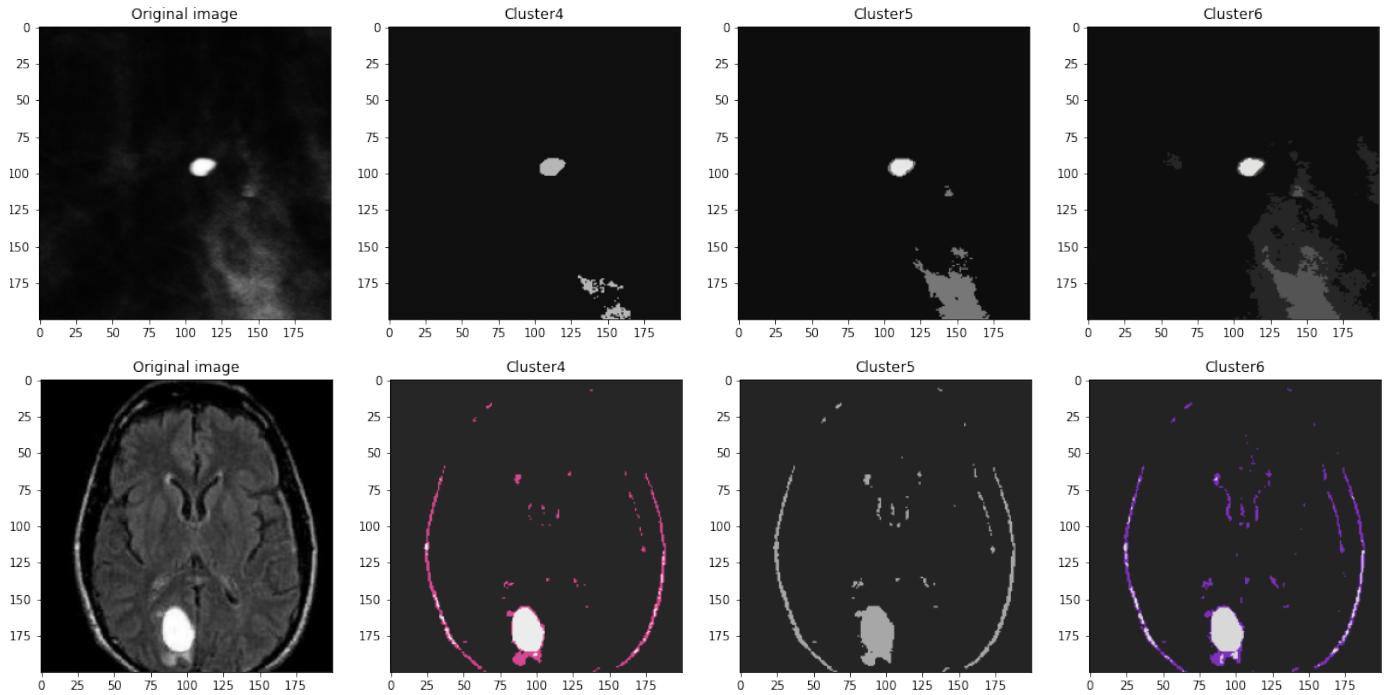
Након описаног поступка слике исписујемо коришћењем функција *subplot()* и *imshow()* из Пајтон библиотеке *cv2*.

Слике које је сегментовао наш алгоритам к-средина изгледају овако:



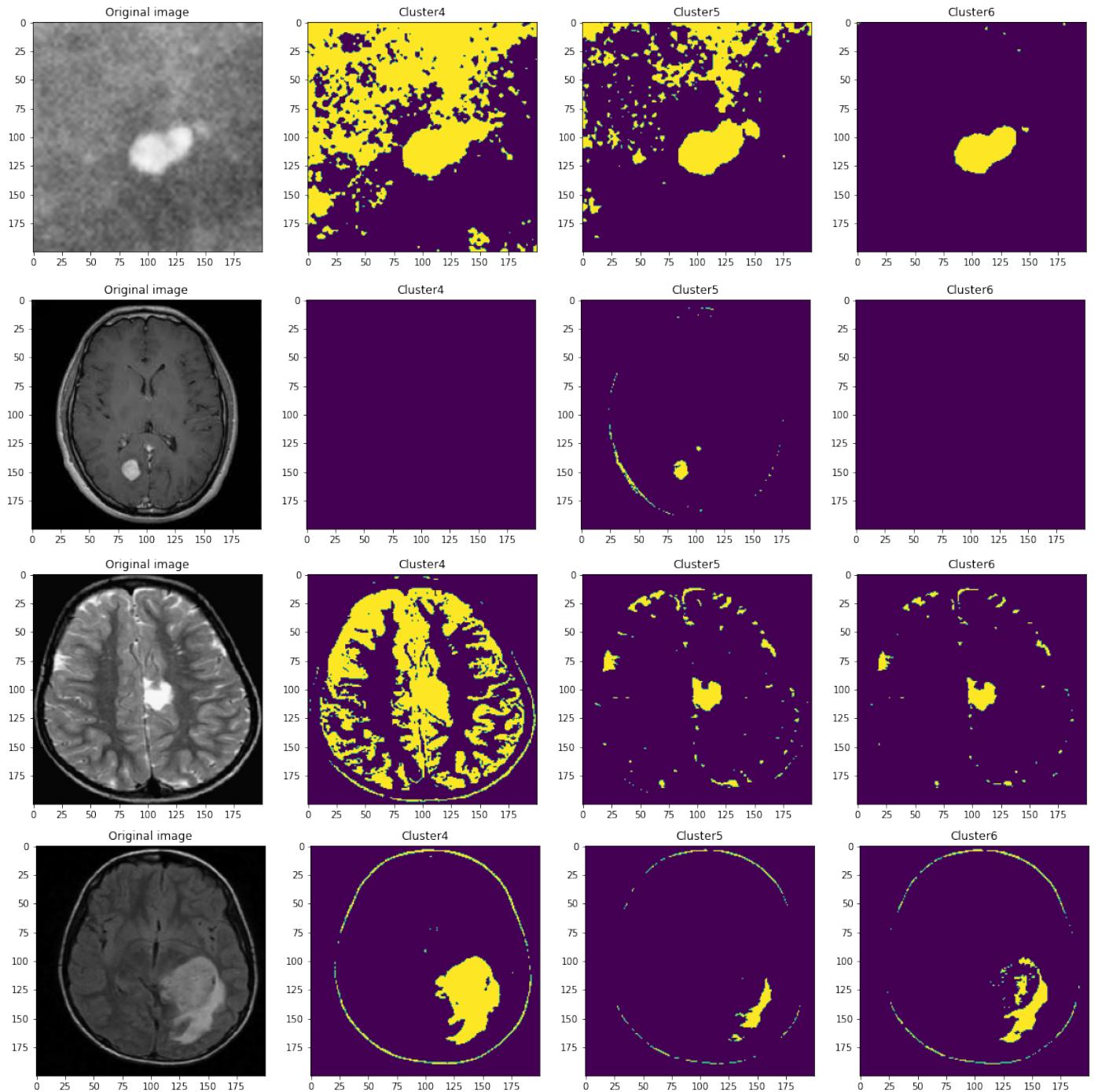


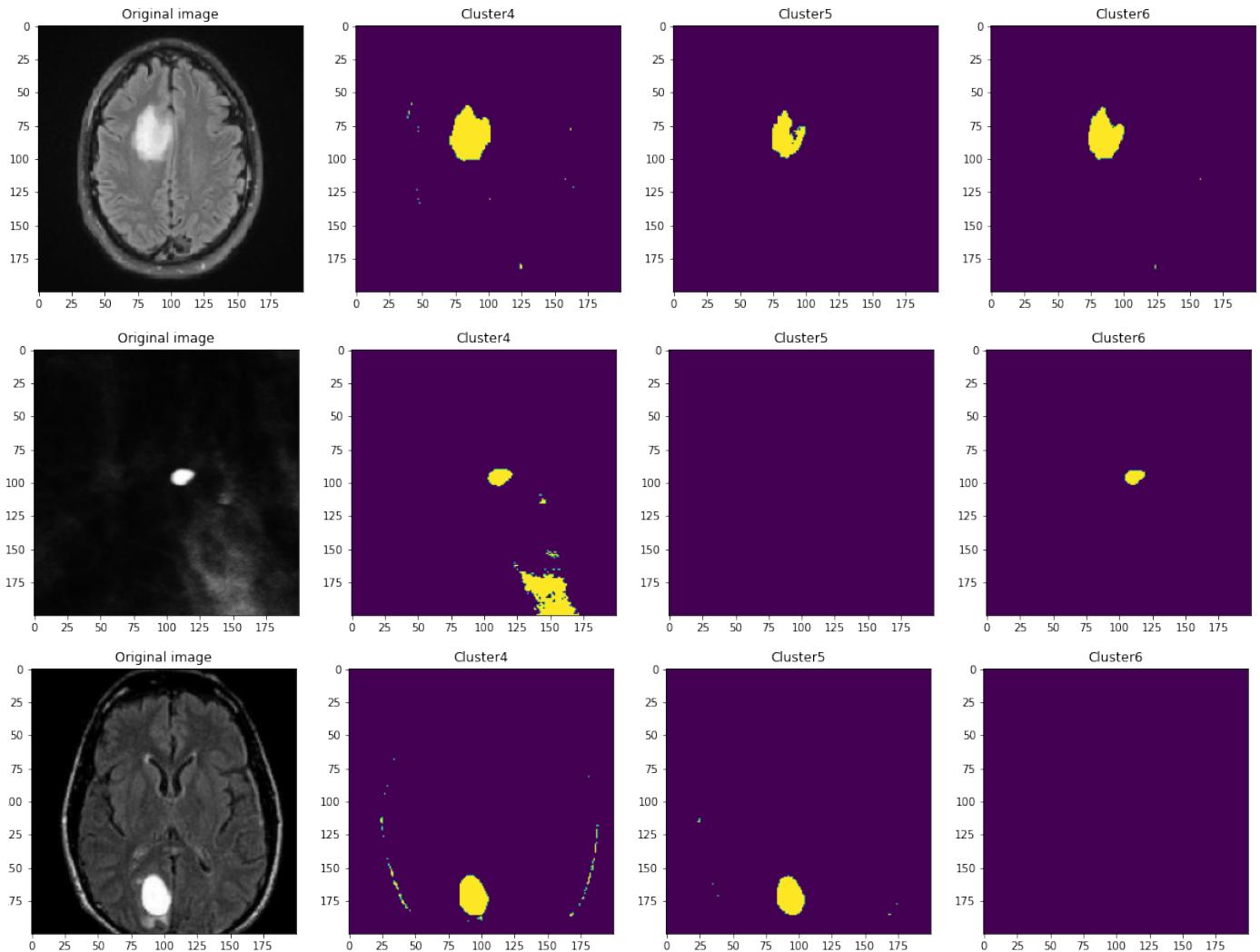




У случају рендгенских снимака мозга врло је битно да се јасно сегментује одређени детаљ на слици, односно тумор ако он постоји, и због тога након горе описаног поступка бинаризујемо слику коришћењем функције *threshold()* из библиотеке *cv2* и пикселе представљамо једном целобројном вредношћу, односно представљамо слику са само две боје. Овај поступак је детаљно описан у одељку [2.3](#).

Након примене нашег алгоритма к-средина, функције *threshold()* и парсирања слике у матрицу пиксела који су димензије један на рендгенске снимке тумора мозга на излазу добијамо слике сегментоване на следећи начин:





### 2.3 Функција `threshold()` и представљање пиксела једном вредношћу

За рендгенске снимке тумора, након горе описаног поступка (за оба алгоритма), примењена је функција `threshold()` из Пајтон библиотеке `cv2` са циљем да тумор који смо издвојили буде обојен једном бојом, а остатак слике буде обојен потпуно другачијом бојом. Ово радимо јер нам је битно да тумор са снимка буде јасно издвојен. Функција `threshold()` као аргументе прима редом:

- слику коју желимо да обрадимо (пожељно је да слика буде сиве боје,

односно да сваки од пиксела буде представљен са три исте вредности, како бисмо добили смислене резултате)

- целобројна вредност која представља референтну вредност, користи се за класификацију вредности које чине пикселе
- целобројна вредност
- параметар дефинисан на нивоу библиотеке *cv2* који одређује како ће радити функција *threshold()*

За наше потребе користићемо *THRESH\_BINARY* као четврти аргумент функције и у том случају функција ради на следећи начин: Свака од три вредности сваког пиксела се упоређује са референтном вредношћу, уколико је вредност мања или једнака од референтне онда се она замењује са нулом, а ако је већа онда се замењује са вредношћу која је задата као трећи аргумент функције.

У нашем случају као референтну вредност узимамо *max\_value - 1* где *max\_value* представља највећу вредност у нашој слици, дакле у том случају само пиксели који имају баш све три вредности *max\_value* ће отићи у трећи аргумент док ће сви остали отићи у нула. Другим речима само један кластер који има највећу вредност центроида ће бити издвојен док ће се сви остали стопити у један. Овај поступак нам је веома користан јер је тумор на рендгенским снимцима светлији од остатка слике па из тог разлога желимо да издвојимо само оне пикселе који имају највећу вредност јер што је већа вредност пиксела то је он светлији.

Како је слика и на излазу из функције *threshold()* сива уколико је пролеђена слика сива пикセル можемо претворити у само једну вредност (јер су сивим сликама сви пиксели представљени са три исте вредности). Слике на излазу су љубичасто жуте јер пиксели сада нису представљени са три вредности (којима бисмо ми одређивали боју) већ са једном и у том случају се користи подразумевана мапа боја (уколико имамо две различите вредности које одређују пикселе на нивоу целе слике мања ће бити представљена љубичастом, а већа жутом бојом).

## 2.4 Поређење резултата

Поређењем резултата које враћају алгоритми к-средина и фази ц-средина можемо видети да су слике приметно боље сегментоване алгоритмом фази ц-средина. Његова сложеност је већа од сложености алгоритма к-средина па сам процес сегментације траје нешто дуже. Такође предност алгоритма фази ц-средина је то што се у пракси показало да даје скоро потпуно исте резултате за исти број кластера на истим сликама, док код алгоритма к-средина то није случај. Разлог томе је што алгоритам к-средина веома зависи од псевдо-случајно задатих почетних вредности центроида кластера.

### 3 Закључак

Кластеровање је један од најкоришћенијих начина за сегментацију слика. Огромна предност фази кластеровања у односу на остале технике сегментације слика је то што је врло отпоран на неправилне границе. Током рада користили смо више различитих слика које смо сегментовали уз испробавање различитих вредности одређених параметара и различитог броја кластера. У већини случајева, код фази ц-средина алгоритма, за издвајање тумора са ренгенских снимака најбоље се показало коришћење 5 кластера, а за сегментацију слика у боји број кластера који је најбоље користити зависи од тога колико детаљна сегментација нам је потребна, у обе ситуације најбоље се показало да је параметар фази формуле ( $p$ ) једнак 2. Код к-средина алгоритма се не може једноставно утврдити за који број кластера добијамо најбоље резултате с обзиром на то да смо у неким ситуацијама при раду са експерименталним подацима добијали најбољу сегментацију за један број кластера, а онда када бисмо опет покренули алгоритам најбоља сегментација би се добила за други број кластера разлог томе је зависност алгоритма к-средина од почетних псевдо-случајних вредности центроида. Код оба алгоритма вредности центроида смо поредили на две децимале јер у том случају добијамо приближно једнако добре резултате као да смо поредили на више децимала, али много раније се заустављамо. Сегментација слика уз помоћ алгоритама фази кластеровања је своју примену између осталог нашла у медицини где је веома битна прецизност резултата. Наравно, фази кластеровање има широк спектар примене поред сегментације слика у разним другим областима неке од њих су економија, вештачка интелигенција и друге.

## **4 Литература**

- M.S. YANG - A Survey of Fuzzy Clustering, October 1993.
- XL Xie, G Beni - A validity measure for fuzzy clustering, 1991.
- Donald E. Gustafson, William C. Kessel - Fuzzy clustering with a fuzzy covariance matrix, 1979.
- Imad Dabbura - K-means Clustering: Algorithm, Applications, Evaluation Methods, and Drawbacks, September 2018
- Мирјана Маљковић - Скалабилни кластер алгоритми, 2008.
- Ненад Митић - Предавање о кластер анализи на Математичком факултету, 2020.