الجانب العملي :

تطبيقنا عبارة عن كشف الورم الدماغي Brain Tumour عن طريق ال color-based segmentation using clusterings algorithm اي عن طريق تجزئة الصورة حسب الالوان باستخدام خوارزميات ال clusterings.

تجزئة الصورة حسب الون لديه قضيتين هامتين وهما : اي مجال الالوان سنستخدم مثل RGB, HSI, CMY, CMYK, YIQ, CIE L∗a∗b∗, etc., ايضا ماهي خوارزمية ال clustering التي سوف نستخدمها في عملية التجزئة .

brain tumor segmentation يتحقق باستخدام الطرق التالية : thresholding , region growing and clustering.

في مشروعنا سنستخدم ال clustering التي تعتبر من أفضل الطرق.

ومن الخوارزميات التي سوف نستخدمها في هذا التطبيق k-means ,c-means,SOM,GMM والتي سوف نطبقها على بكسلات الصورة ونقارن بينها حسب النتائج.

ال datasets في مشروعنا عبارة عن مجموعة من الصور لدماغ اما مصاب بالورم او غير مصاب بالورم.فعن طريق تطبيق color-based segmentation يمكننا من معرفة النتيجة اي نكشف عن الورم الدماغي.

النظام يمر بالمراحل التالية:

Input original image\*

Pseudo Color Translation\*

Color Space Translation\*

Implementation of Clustering Algorithms\*

Cluster Selection \*

performance measures\*

Pseudo Color Translation

هنا يتم تحويل الصورة الى صورة ملونة ان لم تكن ملونة اي الى صورة RGB التي تحتوي على قيم R و G و B اي مجال القيم من 0 الى 255.

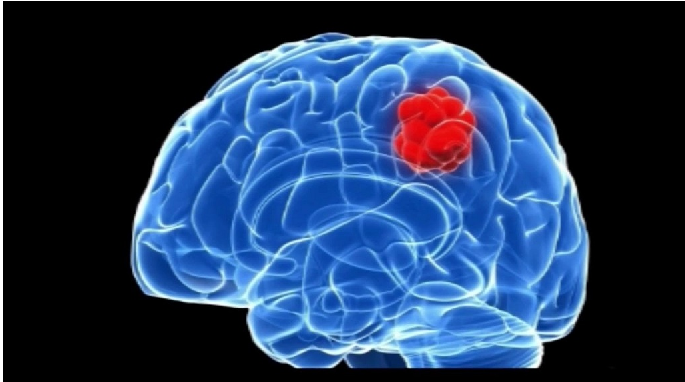
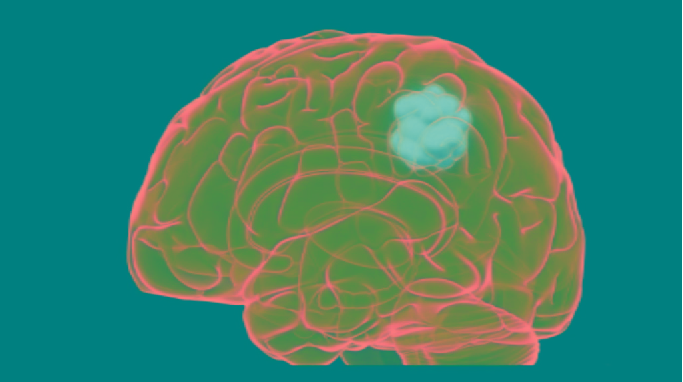


image afterPseudo Color Translation

Color Space Translation :

هنا يتم تحويل صورة RGB الى صورة L\*a\*b اي CIELab كي يتم تطبيق الخوارزميات عليها.

حسث L هي luminosity layer طبقة للاضائة و a يمثل اسقاط الاخضر والاحمر اما ال b يمثل اسقاط الازرق والاصفر.



*L*\**a*\**b*\* image after Color Space Translation

Implementation of Clustering Algorithms :

سوف نطبق الخوارزميات الاربعة التالية : k-means ,c-means,SOM,GMM على بكسلات الصورة للكشف عن الورم.

k-means :

بعد تحويل الصورة من صورة RGB الى صورة Lab نطبق خوارزمية k-means عليها.

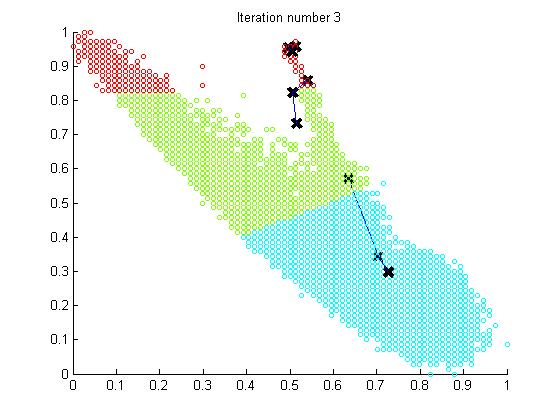
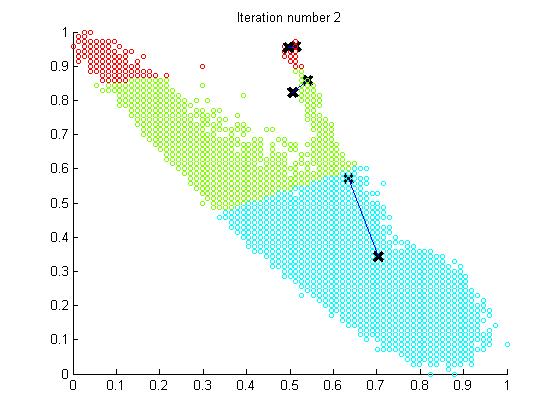
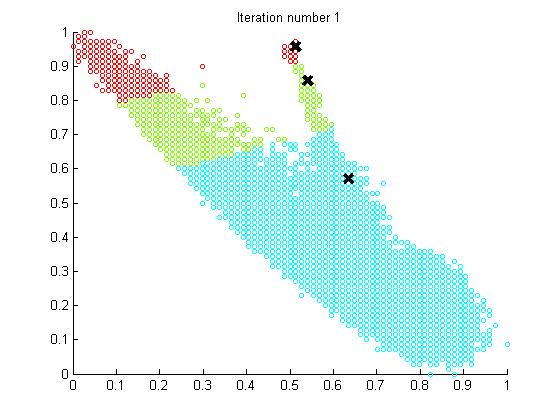
سيكون تطبيق الخوارزمية على 2D dimension والمحورين الذين سنتعامل معهما هما a و b و

سوف نرسم بكسلات الصورة اعتمادا على قيم a و b لبكسلات الصورة.

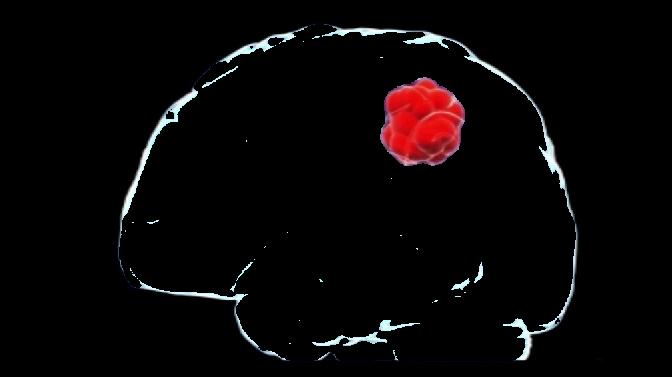
وتعمل الخوارزمية كما يلي :

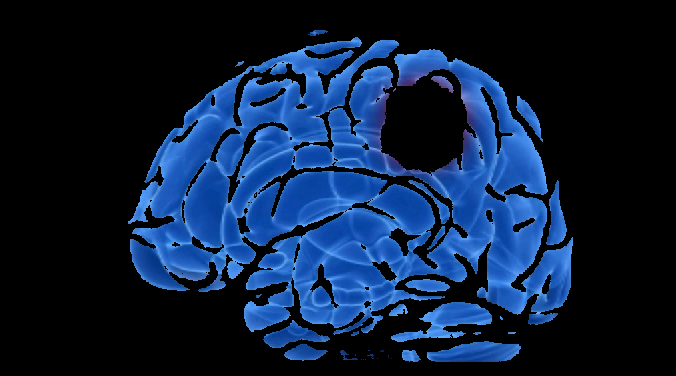
* يتم أختيار مراكز عشوائية(centroid) في البداية حسب قيم k .
* كل بكسل في الصورة يتم مقارنته مع المراكز اعمادا على Euclidian Distance ويتم اختيار المركز الاقرب وتصنيف البكسل له .
* بعد تحديد كل عنقود وتحديد البكسلات التابعة له يتم بعد ذلك حساب مراكز جديدة للعنقود عن طريق المتوسط mean .
* بعد اختيار مراكز جديدة للعناقيد,تكرر الخوارزمية نفسها .
* يتم تكرار الخوارزمية الى ان يتم الاستنتاج ان مراكز العناقيد لاتتغير .

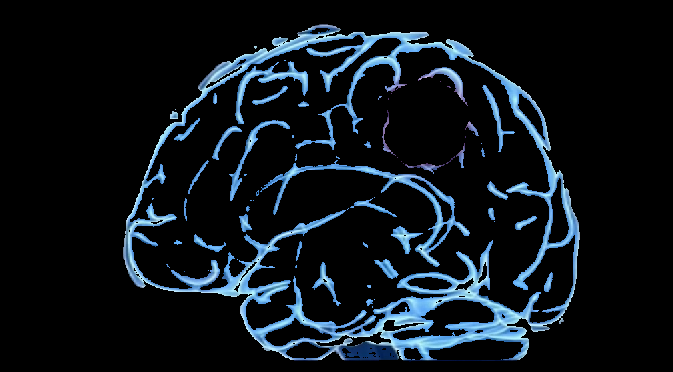
وهذه الطريقة موضحة كما يلي :



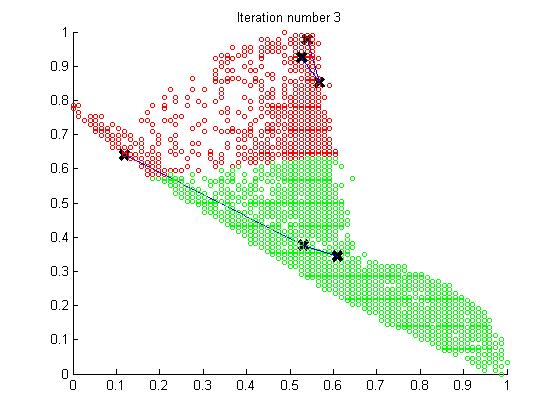
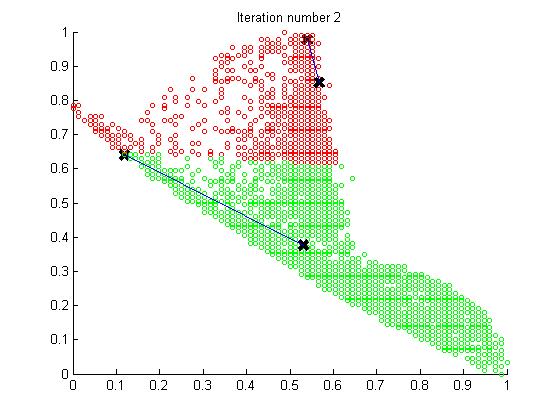
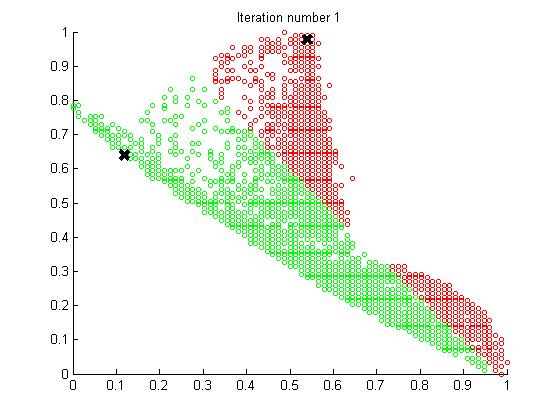
والخرج هو:







اما في حال كان عدد ال cluster هو : k=2

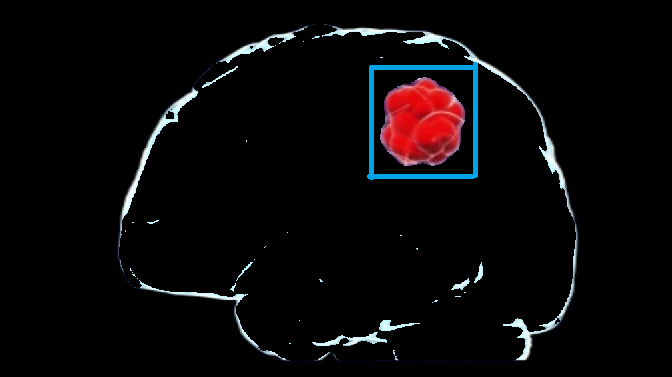


ويتم الحصول على الصورتين بعد تنفيذ الخوارزمية:



حساب مساحة الورم في الصورة :

يتم حساب المساحة عن طريق اختيار مساحة الورم وحساب قيم البكسلات ضمن المساحة وهذه الطريقة تعطي نتائج جيدة .



objective function of k-means clustering :

تابع الخطأ لخوارزمية ال k-means :

https://home.deib.polimi.it/matteucc/Clustering/tutorial_html/images/image009.gif

**complexity of K-Means algorithm :**

تعقيد الخوارزمية هو :

O(n\*K\*I\*D)

حيث :

n : عدد بكسلات الصورة الكلي .

K : عدد المراكز اي centroid .

I : عدد تكرار الخوارزمية .

D : squared euclidean distance .

الايجابيات :

* خوارزمية بسيطة .
* اذا كان عدد المتحولات كبير سوف تعمل الخوارزمية بشكل اسرع بشرط ان يكون عدد ال k قليل .
* تكون فعالة اذا كان شكل العناقيد كروي .

السلبيات :

* لايمكن التعامل مع بيانات ذات احجام ضخمة ومختلفة الكثافات.
* غير قادرة على تحديد القيم المتطرفة .
* محددة على بيانات لها مفهوم ال (centroid) .
* لا تعمل جيدا مع داتا كبيرة .

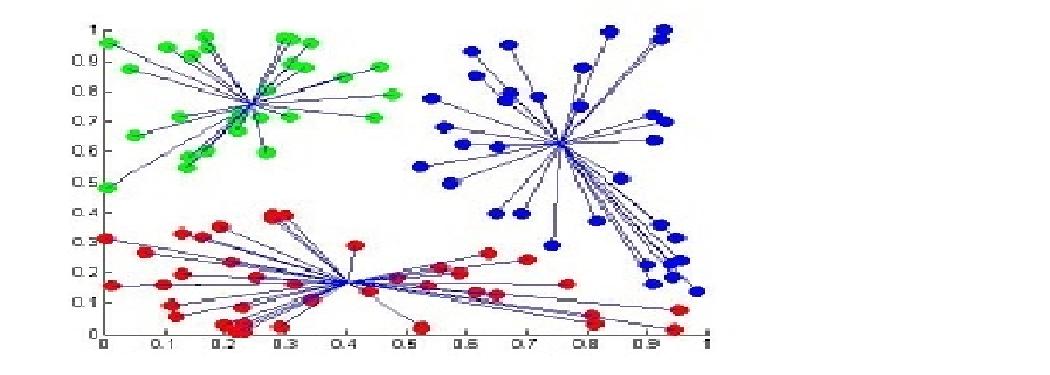
c-means :

في خوارزمية ال k-means كل بكسل في الصورة يكون تابع ل centroid واحد فقط .

اما في هذه الخوارزمية يمكن للبكسل ان يتبع لأكثر من centroid حسب علاقة بينهما .

في نهاية تنفيذ الخوارزمية ينتج لدينا مصفوفة ال membership والتي تمثل علاقة كل بكسل بكل centroid.

وبشكل تقليدي يمكننا اختيار winning اي لاي centroid تابع البكسل في الصورة .



**complexity of C-Means algorithm :**

تعقيد هذه الخوارزمية هو O(n \* d \* c^2 \* i) .

حيث :

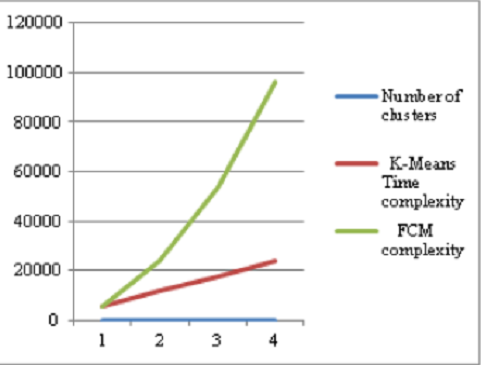
n : عدد بكسلات الصورة الكلي .

d :بارمتر

c : عدد ال clusters .

i : عدد التكرارات للخوارزمية .

نلاحظ المقارنة بين الخوارزميتين k-means و FCM من حيث التعقيد :



FCM تعمل بشكل ابطئ من الخوارزمية k-means .

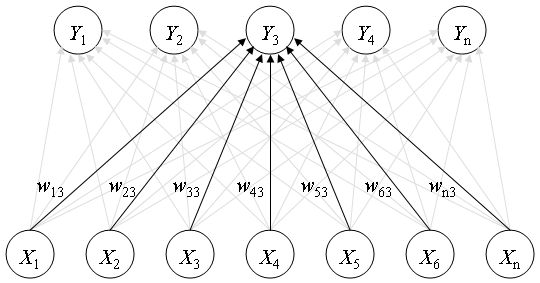
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  | | --- | | **Objective function minimization status** | | |  | | --- | | **No. of iterations** | | image |
|  | 15 |  |
| C:\Users\wade\AppData\Local\Microsoft\Windows\INetCache\Content.Word\3.jpg | 12 | C:\Users\wade\AppData\Local\Microsoft\Windows\INetCache\Content.Word\image004.jpg |
| C:\Users\wade\AppData\Local\Microsoft\Windows\INetCache\Content.Word\2.jpg | 15 | C:\Users\wade\AppData\Local\Microsoft\Windows\INetCache\Content.Word\image003.jpg |

SOM :

وهي شبكة عصبية ذات اوزان وعصبونات,وقد نفذنا الخوارزمية في مجال الالوان Lab ومجال الالوان RGB حيث ال SOM لديها طبقة دخل تمثل ال futureوطبقة خرج تمثل عدد العناقيد k.

في مجال الالوان Lab يوجد لدينا اثنان futureوهما a و b , اما في RGB لدينا ثلاثة وهم R و G و B .

وهذه الخوارزمية تحل مشكلة تعدد ال feutre التي تعاني منها k-means حيث تحتاج الى بيانات كبيرة لتحقيق أفضل اوزان للشبكة .



في حال مجال الالوان RGB وk=3 :

X1=R

X2 =G

X3=B

Y1 =1

Y2 =2

Y3 =3

في حال مجال الالوان Lab و k=3 :

X1=a

X2=b

Y1=1

Y2=2

Y3=3

نلاحظ ان للشبكة طبقة دخل وطبقة خرج , في تطبيقنا طبقة الدخل عبارة عن اثنان او ثلاثة من ال future حسب نوع مجال الالوان المطبق في الخوارزمية , اما طبقة الخرج هي عبارة عن

k=2,3,4.. .

وايضا هذه الخوارزمية يمكن ان تعمل على اي topology .

**complexity of SOM algorithm :**

من أجل شبكة مكونة من N عصبون دخل و K عصبون خرج وM داتا( نقطة ) يكون تعقيدها :

1 – من اجل حساب المسافة بين الدخل الاوزان : O(N\*M\*k)

2 – من اجل اختيار الخلية الرابحة : O(N\*(K\*log(k)))

3 – من اجل تحديث الاوزان : O( N\*M\*K)

GMM :

تحل هذه الخوارزمية مشكلة بكسلات الصورة التي لانستطيع معرفة اي عنقود تابعة .

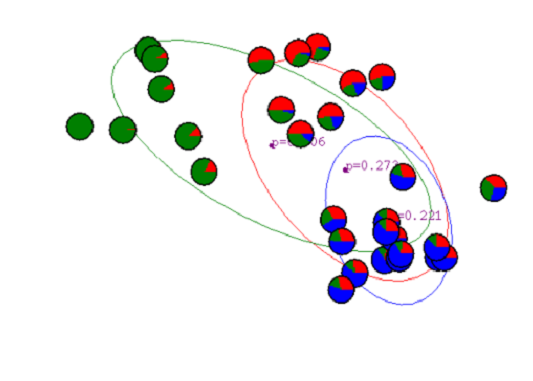
وهي تعتمد على الاحمالات وتستخدم خوارزمية EM لتحديث parameters .

حيث الخوارزمية EM لديها خطوتين :

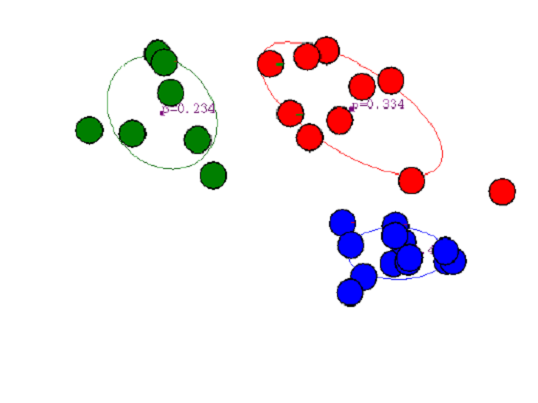
الخطوة E : لمعرفة وحساب احتمال كل بكسل في الصورة ان يتبع ل cluster معين .

الخطوة M : لتحديث parameters الخوارزمية GMM وما المتوسط والتباين .

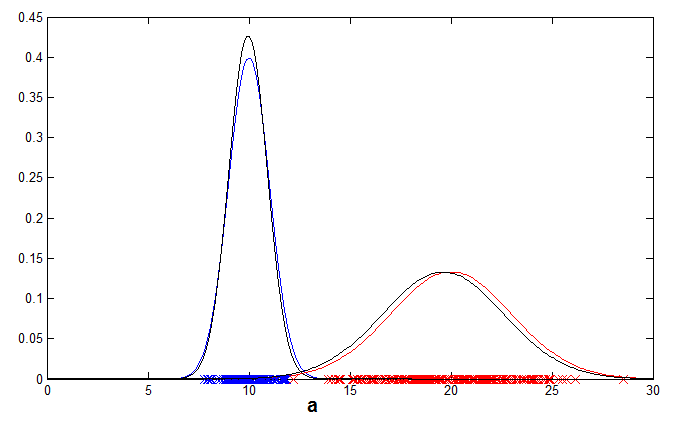
هنا نلاحظ تم اختيار دوائر بشكل عشوائي حسب متوسط وتباين عشوائي للخوارزمية :



وبعد تنفيذ ال GMM وتكرار الخوارزمية EM عدد من المرات :



في تطبيق الخوارزمية على صورة في مجال الالوان Lab هنا نأخذ المجال a فقط من اجل تحسين النتائج .



**complexity of GMM algorithm :**

نلاحظ ان لهذه الخوارزمية خطوتين هما E وM لذلك تعقيدها هو تعقيد الخطوة E يضاف اليها تعقيد الخطوة M :

من اجل E تعقيدها O(n \*k)

من اجل M تعقيدها O( k )

مضروب بعدد تكرارات الخوارزمية I

فيصبح التعقيد الكلي O( n \* k \* I )

لكن هذه الخوارزمية بطيئة في تحقيق ال convergence اي تحقيق cover لجميع بكسلات الصورة بعد عدد من التكرارات .

في هذا الجدول نلاحظ زمن تنفيذ كل خوارزمية بالترتيب حسب حجم الصورة .

|  |  |  |
| --- | --- | --- |
| Computation Time  (sec) | size | image |
| 3.1463   5.5647  34.7564    10.1244 | 285 x 210 |  |
| 2.4353   6.7435  36.7564  11.2453 | 500 x 535 | C:\Users\wade\AppData\Local\Microsoft\Windows\INetCache\Content.Word\image005.jpg |
| 1.2445   4.5636  29.8675  9.2453 | 135 x 132 | image003 |

وبعد تعديل حجم الصورة اي انقاص عدد البكسلات نلاحظ ان زمن التنفيذ قد نقص .

|  |  |  |
| --- | --- | --- |
| Computation Time  (sec) | size | image |
| 0.2352   4.5441  12.7564    8.3455 | 200 x 147 | C:\Users\wade\AppData\Local\Microsoft\Windows\INetCache\Content.Word\image001.jpg |
| 1.4045   5.1431  20.6554  9.8675 | 350 x 375 | C:\Users\wade\AppData\Local\Microsoft\Windows\INetCache\Content.Word\image005.jpg |
| 0.5435   0.6546  10.7564  5.7654 | 95 x 93 | image003 |

مقارنة الخوارزميات حسب عدد ال feuture .

|  |  |  |  |
| --- | --- | --- | --- |
| 4-feuture | 3-feuture | 2-feuture | Cluster algorithm |
| \_ | تعمل بشكل بطئ | تعمل جيدا | k-means |
| \_ | تعمل جيدا.. | تعمل جيدا | C-measn |
| تعمل جيدا | تعمل جيدا | تعمل | SOM |
| \_ | تعمل | تعمل جيدا | GMM |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| performance | | | | | Number of  Clusters(K) | |
| EM | SOM | FCM | k-means |  | |
| 63 | 62 | 63 | 59 | 2 | |
| 74 | 67 | 71 | 67 | 3 | |
| 87 | 84 | 84 | 78 | 4 | |
| 97 | 89 | 89 | 87 | 5 | |

في هذا الجدول نلاحظ زمن تنفيذ كل خوارزمية في حال عدد ال clusters : k=2

|  |  |  |
| --- | --- | --- |
| Computation Time  (sec) | Clustering Algorithm | S.No |
| 3.4543 | k-means | 1 |
| 5.0812 | c-means | 2 |
| 45.9632 | SOM | 3 |
| 11.4353 | GMM | 4 |

في هذا الجدول نلاحظ زمن تنفيذ كل خوارزمية في حال عدد ال clusters : k=3

|  |  |  |
| --- | --- | --- |
| Computation Time  (sec) | Clustering Algorithm | S.No |
| 6.5436 | k-means | 1 |
| 9.9235 | c-means | 2 |
| 45.7476 | SOM | 3 |
| 16.4014 | GMM | 4 |

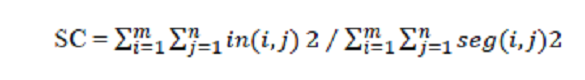
في هذا الجدول نلاحظ زمن تنفيذ كل خوارزمية في حال عدد ال clusters : k=4

|  |  |  |
| --- | --- | --- |
| Computation Time  (sec) | Clustering Algorithm | S.No |
| 7.5436 | k-means | 1 |
| 11.6554 | c-means | 2 |
| 46.6720 | SOM | 3 |
| 20.4353 | GMM | 4 |

المقارن بين الخوارزميات حسب :

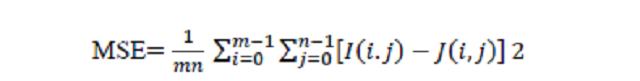
**Structural Content :**

هذا المعيار يعكس نوعية الصورة التي تمت تطبيق التجزئة عليها .



قيم صغيرة ل SC يعني نوعية جيدة للصورة .

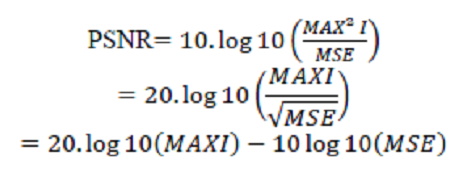
**Means Squared Error :**



قيم صغيرة له تعني خطأ صغير وجودة عالية .

MSE و PSNR يتناسبان بشكل عكسي.

**Peak Signal to Noise Ratio :**



عادة يتم تقديره كمقياس ديسبل ويقاس بين صورتين . الصورة الاصلية والصورة التي الناتجة عن ال تجزئة .

اذا كانت قيمتها فوق ال 30 يعني الخرج الناتج لديه وضوح , وتأخذ قيمها من 0 الى الانهاية.

قيم كبيرة تعني كفائة عالية للصورة .

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| PSNR | MSE | SC |  | image |
| 9.8715 | 0.3377 | 4.7466 | k-means |  |
| 8.8385 | 8.4964 | 6.3501 | FCM |
| 20.730 | 549.53 | 1.2279 | SOM |
| 7.3347 | 122.66 | 2.4456 | GMM |
| 9.8715 | 0.3377 | 4.7466 | k-means |  |
| 8.8385 | 8.4964 | 6.3501 | FCM |
| 20.730 | 549.53 | 1.2279 | SOM |
| 7.3347 | 122.66 | 2.5456 | GMM |
| 9.8715 | 0.3377 | 4.7466 | k-means |  |
| 8.8385 | 8.4964 | 6.3501 | FCM |
| 20.730 | 549.53 | 1.2279 | SOM |
| 7.3347 | 152.66 | 2.5666 | GMM |
| 9.8715 | 0.3377 | 4.7466 | k-means |  |
| 8.8385 | 8.4964 | 6.3501 | FCM |
| 20.730 | 549.53 | 1.2279 | SOM |
| 7.3347 | 152.66 | 2.5664 | GMM |

|  |  |  |  |
| --- | --- | --- | --- |
| **True condition** | | |  |
| Condition negative | Condition positive |  | **Predicted condition** |
| [**False positive**](https://en.wikipedia.org/wiki/False_positive) ([Type I error](https://en.wikipedia.org/wiki/Type_I_error)) | [**True positive**](https://en.wikipedia.org/wiki/True_positive) | Predicted condition positive |
| [**True negative**](https://en.wikipedia.org/wiki/True_negative) | [**False negative**](https://en.wikipedia.org/wiki/False_negative) ([Type II error](https://en.wikipedia.org/wiki/Type_II_error)) | Predicted condition negative |

Accuracy = (TP+TN)/(TP+TN+FP+FN)

Precision = TP/(TP+FP)

Recall = TP/(TP+FN) **true positive rate**

specificity = TN/(FP+TN) **true negative rate**

سنقارن اداء الخوارزميات في مجال الالوان Lab و RGB :

في حال كان مجال الالوان RGB :

|  |  |  |  |
| --- | --- | --- | --- |
| Precision | Accuracy | **Recall%** | **Type of Clustering** |
| 100 | 93.4 | 95.2 | k-means |
| 100 | 80.7 | 87.4 | c-means |
| 100 | 94.5 | 89.2 | SOM |
| 100 | 84.6 | 80.9 | GMM |

في حال كان مجال الالوان Lab :

|  |  |  |  |
| --- | --- | --- | --- |
| Precision | Accuracy | **Recall** | **Type of Clustering** |
| 100 | 91.9 | 94.5 | k-means |
| 100 | 90.7 | 88.4 | c-means |
| 100 | 80.2 | 89.2 | SOM |
| 100 | 85.4 | 79.8 | GMM |

بعد العمل على تحسين الخوارزميات اعطت نتائج جيدة ..

|  |  |
| --- | --- |
| Specificity% | Clustering algorithm |
| 91 | k-means |
| 89.1 | c-means |
| 79.1 | SOM |
| 82.1 | GMM |