

КУРБАТОВ Д. И., ЕГОРОВА Д. К.

**ПРИМЕНЕНИЕ НЕЧЕТКОГО АЛГОРИТМА КЛАСТЕРИЗАЦИИ C-MEANS
ДЛЯ СЕГМЕНТАЦИИ ИЗОБРАЖЕНИЙ**

Аннотация. В статье рассматривается применение нечеткого алгоритма кластеризации c-means для анализа изображений. Приведена программная реализация алгоритма на языке C#.

Ключевые слова: кластеризация, нечеткий метод c-means, итерации, цвет, пиксели.

KURBATOV D. I., EGOROVA D. K.

**APPLICATION OF FUZZY C-MEANS CLUSTERING
ALGORITHM FOR IMAGE SEGMENTATION**

Abstract. Application of fuzzy clustering algorithm with c-means for image analysis is considered in this article. A software implementation of the algorithm in C# is provided.

Keywords: clustering, fuzzy c-means method, iteration, color, pixels.

Введение. Разбиение изображения на однородные области является ключевым элементом компьютерного (технического) зрения. Для этих целей могут быть использованы методы кластерного анализа, так как они позволяют осуществить сегментацию пикселей на основе сходства по цвету или уровня интенсивности серого.

В работе приведена программная реализация сегментации цветного изображения методом c-means, относящегося к методам «мягкой» кластеризации и позволяющего более точно вычислить принадлежность элемента кластеру. Этот алгоритм успешно используется для кластеризации изображений и неконтролируемой сегментации медицинских, геологических, спутниковых изображений и т.п. [1].

Описание алгоритма. Нечеткий алгоритм кластеризации c-means был разработан (для случая $m=2$) J. C. Dunn в 1973 г. и усовершенствован (для случая $m>2$) J. C. Bezdek в 1981 г. Данный метод предполагает, что входные данные могут принадлежать более чем одному кластеру одновременно, то есть первоначально для каждого вектора, описывающего исследуемый объект, случайным образом определяется вероятность принадлежности заданным кластерам. Алгоритм получает на входе набор кластеризуемых векторов, количество кластеров, коэффициент неопределенности m и коэффициент $\varepsilon > 0$, определяющий точность алгоритма. Затем запускается итерационный процесс, состоящий в выполнении следующей последовательности действий: 1) расчет центров кластеров; 2) расчет расстояния от каждого вектора до центра каждого кластера; 3) расчет и нормализация коэффициентов принадлежности векторов кластерам; 4) расчет значения матрицы нечеткого

разбиения и сравнение этого значения со значением матрицы нечеткого разбиения на предшествующей итерации. На выходе получают матрицу вероятностей принадлежности каждого входного вектора каждому кластеру.

Таким образом, нечеткий алгоритм c-means минимизирует величину

$$\sum_{i=1}^{|X|} \sum_{j=1}^C u_{i,j}^m \|x_i - c_j\|^2, \quad 1 \leq m \leq \infty, \quad (1)$$

где $m \in R$, $u_{i,j}$ – коэффициент принадлежности вектора x_i кластеру c_j , x_i – i -ый компонент $|X|$ -мерного вектора X , C – количество кластеров, c_j – центр j -ого кластера, а $\| * \|$ – норма, определяющая расстояние от вектора до центра кластера. Нечеткое разбиение входных данных на кластеры производится итеративной оптимизацией функции (1), пересчетом коэффициентов принадлежности $u_{i,j}$ и переопределением центров кластеров c_j . [2]

Вычисляемые величины. На каждой итерации вычисляются следующие величины.

1. Центры кластеров:

$$c_j = \frac{\sum_{i=1}^n u_{i,j}^m x_i}{\sum_{i=1}^n u_{i,j}^m}, \quad (2)$$

где $u_{i,j}$ – коэффициент принадлежности x_i вектора к кластеру c_j .

2. Коэффициент принадлежности:

$$u_{i,j} = \frac{1}{\sum_{i=1}^n \left(\frac{\|x_i - c_j\|}{\|x_i - c_k\|} \right)^{\frac{2}{m-1}}}. \quad (3)$$

3. Решающая функция:

$$\max(|u_{i,j}^{(k)} - u_{i,j}^{(k-1)}|), \quad (4)$$

где k – номер итерации алгоритма.

4. Проверка завершения итерационного процесса.

Реализация алгоритма. Далее приведена программная реализация алгоритма c-means для сегментации цветных изображений на языке C# в программной среде VisualStudio 2017 [3].

Графический интерфейс представлен на рисунке 1. Для загрузки сегментируемого изображения предусмотрено меню «File».

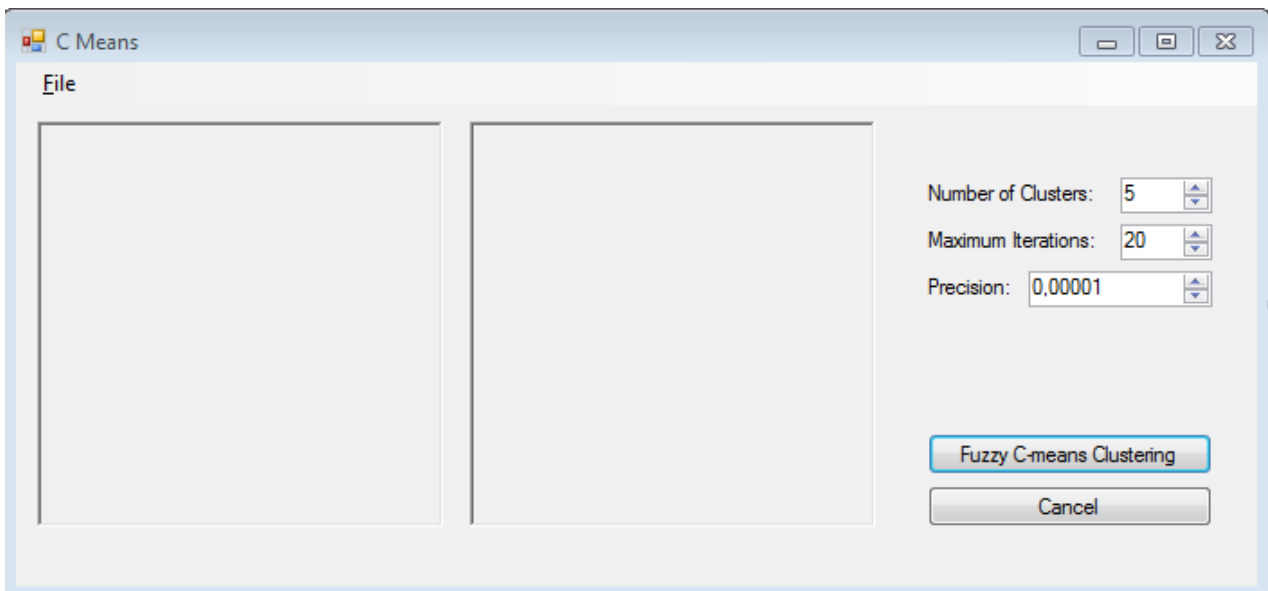


Рис. 1. Графический интерфейс.

Параметры, доступные для изменения, приведены в правой верхней части интерфейса программы – это количество кластеров, максимальное количество итераций и точность.

При активации «Fuzzy C-means Clustering» запускается процесс создания точечных объектов кластера для каждого пикселя изображения (см. код, приведенный ниже).

```
List<ClusterPoint> points = new List<ClusterPoint>();
for (int row = 0; row < originalImage.Width; ++row)
{
    for (int col = 0; col < originalImage.Height; ++col)
    {
        Color c2 = originalImage.GetPixel(row, col);
        points.Add(new ClusterPoint(row, col, c2));
    }
}
```

Затем создается заданное количество «ClusterCentroid» объектов кластера. Центры кластеров (или центроиды) на первой итерации выбираются случайным образом и корректируются алгоритмом (см. код).

```
List<ClusterCentroid> centroids = new List<ClusterCentroid>();
//Create random points to use as the cluster centroids
Random random = new Random();
for (int i = 0; i < numClusters; i++)
```

```

{
    int randomNumber1 = random.Next(sourceImage.Width);
    int randomNumber2 = random.Next(sourceImage.Height);
    centroids.Add(new ClusterCentroid(randomNumber1, randomNumber2,
        filteredImage.GetPixel(randomNumber1, randomNumber2)));
}

```

Далее создается FCM-объект и запускаются итерации (см. код).

```

FCM alg = new FCM(points, centroids, 2, filteredImage, (int)numericUpDown2.Value);
k++;
alg.J = alg.CalculateObjectiveFunction();
alg.CalculateClusterCentroids();
alg.Step();
double Jnew = alg.CalculateObjectiveFunction();
Console.WriteLine("Run method i={0} accuracy = {1} delta={2}",
    k, alg.J, Math.Abs(alg.J - Jnew));
backgroundWorker.ReportProgress((100 * k) / maxIterations, "Iteration " + k);
if (Math.Abs(alg.J - Jnew) < accuracy) break;

```

После завершения итераций алгоритм выполняет процесс построения результирующего изображения, назначая каждый пиксель кластеру, для которого он имеет наибольший коэффициент принадлежности.

Для тестового примера было выбрано изображение, представленное на рисунке 2. Сегментация осуществлялась на 2 кластера.



Рис. 2. Исходное изображение.

Результирующее изображение представлено на рисунке 3.



Рис.3. Результат выполнения алгоритма.

ЛИТЕРАТУРА

1. Jain A. K. Data clustering: 50 years beyond K-means // Pattern Recognition Letters. – 2010. – Vol. 31. – P. 651–666.
2. Барсегян А. А., Куприянов М. С., Холод И. И., Тесс М. Д., Елизаров С. И. Анализ данных и процессов: учеб. пособие / под ред. А. А. Барсегян. – 3-е изд., перераб. и доп. – СПб.: БХВ–Петербург, 2009. – 512 с.
3. Gauge С. Cluster analysis [Электронный ресурс]. – Режим доступа: <https://ru.scribd.com> (дата обращения 08.05.2019).