UDC 004.8

METHODS AND ALGORITHMS FOR IMAGE CLUSTERING IN MACHINE LEARNING

Zholdasbek Bibarys
3rd-year student, Educational Program 6B06118 – "Software Engineering"
M.Kh. Dulaty Taraz University, Taraz, Republic of Kazakhstan
Ozgeldi Nurbolat
9th-grade student, Secondary School No. 48 named after T.Ryskulov
Taraz, Republic of Kazakhstan
Scientific supervisor: Adilova Aknur
ak.adilova@dulaty.kz

Image clustering is one of the key tasks in the fields of computer vision and image processing. It is used to group similar images or objects within images and is widely applied in areas such as medical diagnostics, satellite image analysis, security (e.g., facial recognition), and image retrieval on the Internet.

With the advancement of machine learning technologies and the emergence of powerful approaches such as deep neural networks, the clustering task has become more accurate and efficient. However, despite significant progress in image processing, several challenges remain unresolved, including difficulties in handling large-scale datasets, selecting optimal methods for different types of images, and improving the speed and accuracy of algorithms.

Therefore, the study of new image clustering methods and the development of efficient algorithms remain relevant and important research directions.

Modern approaches in machine learning, such as the use of convolutional neural networks (CNNs) for feature extraction from images, open up new opportunities for clustering. They enable the automation of the feature extraction process and significantly improve clustering quality, especially in complex scenarios with high variability in images.

However, despite considerable progress, a number of challenges remain, including issues related to the scalability of methods, the need for processing large volumes of data, and the difficulty of selecting optimal hyperparameters.

Image clustering is the process of grouping images based on their visual features, such as color, texture, shape, contours, or more complex features extracted using machine learning techniques. In this task, it is important to identify methods and algorithms that can effectively handle visual data while taking into account the high dimensionality of images. Let us consider the main methods of image clustering.

The K-means method is one of the most popular and simplest clustering techniques used to group data, including images, into several clusters. Its main objective is to minimize intra-cluster variance while maximizing inter-cluster differences. Let us examine in detail how the K-means algorithm works and how it is applied in image clustering.

Algorithm Steps:

Initialization. The algorithm begins by selecting initial centroids for the clusters. This can be done randomly by choosing *k* objects from the dataset as initial centroids, or by using more advanced initialization techniques such as the *k-means++* method, which helps distribute the initial centroids more evenly across the data space.

Assignment of objects to clusters. In the second step, each object (in our case, an image) is assigned to the nearest centroid. This is done using a distance metric that measures the "similarity" between objects (most commonly, the Euclidean distance is used). Each object is assigned to the cluster whose centroid is closest to it.

Centroid update. After all objects have been assigned to clusters, new centroids are recalculated for each cluster. The centroid of a cluster is the mean value of all points (or objects)

within that cluster. In the case of images, this may correspond to the average values of color channels or other image features.

Iteration. Steps 2 and 3 are repeated until the centroids no longer change significantly (or the changes become negligible), indicating convergence of the algorithm. At this point, the clustering process is considered complete.

Let us assume we have a dataset of images. Each object (image) is represented by a set of features, for example, feature vectors obtained through feature extraction (such as color histograms, texture features, or even convolutional features). The K-means algorithm iteratively groups these images into *k* clusters based on these features.

Advantages and Disadvantages of the K-means Method

Advantages:

- *Simplicity and speed:* The K-means method is relatively easy to implement and works quickly even with large datasets.
- *Efficiency:* The algorithm performs well on large datasets, especially when the data is linearly separable.
- *Versatility:* This method can be applied to various types of data, not only images but also, for example, textual or numerical data.

Application of K-means in Image Clustering. The K-means algorithm can be applied to images in various domains:

- *Image segmentation:* K-means can be used to divide an image into several regions (segments), each containing pixels with similar characteristics such as color or texture. For example, in a natural landscape image, the algorithm can separate segments such as the sky, land, and vegetation.
- *Similar image retrieval:* In image search systems, the algorithm can cluster images based on similarities in features such as color or texture. When a user searches for an image, the system can first cluster images by features and then search within the most relevant cluster.
- *Content-based image grouping:* The K-means algorithm can be used to group images into categories. For instance, images of animals can be separated from landscape images. Features can be extracted using computer vision techniques such as HOG descriptors or convolutional neural networks.
- *Anomaly detection:* In image quality control tasks (e.g., in manufacturing processes), K-means can be used to detect anomalous images. Images that do not belong to any cluster may indicate defects or irregularities.

The K-means method is a powerful and versatile clustering tool widely used in various fields, including image processing. Despite its simplicity, this method has several limitations. To improve clustering results, various enhancements of the algorithm can be applied, such as more effective centroid initialization using the *k-means++* method, or the use of algorithms like DBSCAN, which do not require a predefined number of clusters.

Hierarchical clustering is a method that constructs a hierarchy of clusters by initially treating each object as an individual cluster and then gradually merging or splitting them. This approach is widely used in data analysis, including image clustering, as it allows the identification of data structures at different levels of abstraction and does not require a predefined number of clusters.

Hierarchical clustering can be divided into two types: agglomerative clustering (bottom-up approach) and divisive clustering (top-down approach).

In agglomerative hierarchical clustering, the process begins by treating each object (or image) as an individual cluster. Then, at each step, the two closest (or most similar) clusters are merged into a single cluster. This process continues until only one cluster remains that contains all objects.

Steps of Agglomerative Clustering:

- *Initialization:* each object is considered as a separate cluster;
- *Distance computation:* a similarity or distance measure is calculated for every possible pair of clusters (e.g., Euclidean distance, Manhattan distance, etc.);

- *Cluster merging:* at each step, the two clusters with the smallest distance or highest similarity are merged;
- *Iteration:* this process is repeated until all objects are combined into a single cluster.

If we apply the agglomerative method to image clustering, we start by representing each image as an individual cluster. At each step, the algorithm merges the two most similar images (for example, based on color characteristics or texture features). As merging continues, the number of clusters gradually decreases until a single cluster containing all images is formed.

Methods for Measuring Distance Between Clusters. At each step, several methods can be used to calculate the distance between clusters:

- *Single Linkage (minimum distance):* the distance between two clusters is defined as the minimum distance between any pair of objects from the two clusters. This method may lead to "chaining" effects, where clusters become elongated like chains.
- *Complete Linkage (maximum distance):* the distance between two clusters is defined as the maximum distance between any pair of objects from the two clusters.
- *Average Linkage (average distance):* the distance between two clusters is calculated as the average distance between all possible pairs of objects from the two clusters.
- *Centroid Linkage:* the distance between clusters is determined based on their centroids (the mean of all objects within a cluster). After merging clusters, a new centroid is computed for the resulting cluster.

Unlike the agglomerative approach, divisive clustering works in the opposite manner. In this case, the process begins with a single cluster that contains all objects, and then this cluster is recursively divided into smaller subsets (subclusters) until the desired level of detail is achieved.

Steps of Divisive Clustering:

- *Initialization:* all objects are considered as one cluster;
- *Cluster splitting:* the cluster is divided into two smaller clusters. This is done by computing a distance measure between objects and partitioning them in such a way that objects within each subgroup are as similar as possible;
- *Iteration:* this process continues until each object forms its own cluster or until the required level of partitioning is reached.

Advantages and Disadvantages of Hierarchical Clustering:

- *No need to predefine the number of clusters:* unlike methods such as K-means, hierarchical clustering does not require specifying the number of clusters in advance, making it convenient for exploring the structure of data;
- *Sensitivity to noise and outliers:* especially in agglomerative clustering, the presence of noise or outliers may lead to the formation of "chained" clusters;
- *Difficulty with large datasets:* due to high computational complexity and the need to store a distance matrix between all objects, hierarchical clustering can be challenging to apply to very large datasets.

Application of Hierarchical Clustering in Image Processing

Image Segmentation. Hierarchical clustering can be utilized to divide an image into multiple regions (segments) based on visual features such as color, texture, or shape.

Grouping of Similar Images. Hierarchical clustering is effective for classifying images into different categories (e.g., landscapes, portraits, architecture) by identifying and grouping visually similar objects.

Object Detection and Extraction in Images. The algorithm can be applied to identify and extract objects or specific regions within an image based on feature similarity; for instance, to isolate different anatomical structures in medical images.

Hierarchical Clustering. Hierarchical clustering is a powerful tool for analyzing data and images, enabling the identification of inherent structure and hierarchical relationships among objects. Despite its computational complexity, it offers several advantages, such as not requiring a predefined number of clusters and providing the ability to visualize data structure through a dendrogram.

DBSCAN (Density-Based Spatial Clustering of Applications with Noise). DBSCAN is a density-based clustering algorithm that is used to identify clusters in data by grouping objects located in high-density regions and separating them from objects in low-density regions, which are treated as noise. DBSCAN does not require a predefined number of clusters, making it particularly useful when the number of clusters is unknown in advance.

Key Principles of the DBSCAN Algorithm:

- *Density*. The algorithm is based on the concept of density. Clusters are defined as regions in the feature space where objects are located sufficiently close to each other.
- *Identification of Clustered Objects.* Objects in high-density regions form clusters, whereas objects located in low-density regions are considered noise or outliers.

Advantages of DBSCAN:

- *No Requirement for a Predefined Number of Clusters.* Unlike methods such as K-means, DBSCAN does not require the number of clusters to be specified in advance. Clusters are automatically determined based on the density of the data.
- *Ability to Detect Arbitrary-Shaped Clusters.* DBSCAN can identify clusters of arbitrary shapes, which is particularly useful for datasets that do not follow predefined geometric structures (e.g., circles or ellipses).
- *Effective Handling of Noise.* DBSCAN efficiently manages noise and outliers by classifying low-density objects as noise rather than forcing them into clusters.

Disadvantages of DBSCAN:

- *Sensitivity to Varying Density.* DBSCAN performs poorly when clusters have significantly different densities. A single global density parameter may fail to detect low-density clusters or may incorrectly merge distinct clusters.
- *Inefficiency in High-Dimensional Data.* In tasks involving high-dimensional data (e.g., high-resolution images or text data with many features), DBSCAN can be computationally expensive and less effective.

Applications of DBSCAN in Image Clustering:

- *Grouping Images by Visual Similarity.* DBSCAN can be used to cluster images based on similarities in visual features, for example, grouping images containing similar objects such as portraits, landscapes, or urban scenes.
- *Analysis of Arbitrary-Shaped Structures.* DBSCAN is particularly useful for analyzing images with complex structures or irregular shapes, such as in medical imaging or satellite image processing tasks.
- *Handling Noisy Data.* In tasks where images contain significant noise (e.g., object recognition or image retrieval systems), DBSCAN can effectively distinguish noise from meaningful clusters.

Application of DBSCAN for Image Clustering. Suppose we have a collection of images, and our task is to classify them based on the similarity of visual features (e.g., color or texture). By applying DBSCAN, we can:

- *Extract image features,* such as color histograms, HOG descriptors, or CNN-based descriptors.
- *Define the parameters ϵ (epsilon)* and *minPts* to adapt the algorithm to the density of the data.
- *Apply DBSCAN,* which groups images into clusters based on similarity and separates noise (if present).

DBSCAN is a powerful and flexible clustering method that enables the automatic detection of clusters in data without requiring the number of clusters to be specified in advance. It is particularly well-suited for handling data with arbitrary shapes and noise, making it highly effective for image clustering tasks, where images may significantly vary in content and noise levels.

Image Features in Clustering. Image features are characteristics that represent the visual content of an image. These features can be classified as global (describing the entire image) or local (describing specific regions or key points within the image). Prior to performing image

clustering, it is essential to extract informative features that effectively capture the visual content of the images. Subsequent clustering is then carried out based on these features.

One of the simplest and most widely used features is the color histogram. It represents the distribution of pixel intensities across different color channels (e.g., red, green, and blue in the RGB color model, or alternative color spaces such as HSV or Lab). Color histograms can be utilized to cluster images according to the similarity of their color compositions.

References

1. Asambaev, A. Zh. *Fundamentals of Artificial Intelligence.* Almaty: Evero, 2017. 168 p.
2. Baimukhamedov, M. F., Baimukhamedova, A. M., Boranbaev, S. N. *Artificial Intelligence: Modern Theory and Practice. Study Guide, Part 1.* Almaty: Bastau, 2020. 248 p.
3. Bartalev, S. A., Khovratovich, T. S. Assessment of satellite image segmentation methods for forest change detection. *Current Problems in Remote Sensing of the Earth from Space*, 2011, vol. 8, no. 1, pp. 44–62.
4. Gorbachenko, V. I. *Machine Learning: учебное пособие.* Moscow: IPR Media, 2023. 218 p.
5. Zapechnikov, S. V. *Fundamentals of Data Mining and Machine Learning: Lecture Notes. Study Guide.* Moscow: National Research Nuclear University MEPhI, 2022. 136 p.