Report On

Image Clustering Using CNN

Submitted in partial fulfillment of the requirements of the Course project in Semester VII of Fourth Year **Computer Engineering**

By

Khanjan Joshi - 26
Pooja Naskar - 30
Prathmesh Malvi - 38

Supervisor
Dr. Megha Trivedi

University of Mumbai

Vidyavardhini's College of Engineering & Technology

Department of Computer Engineering

(2023-24)

Vidyavardhini's College of Engineering & Technology
Department of Computer Engineering

CERTIFICATE

This is to certify that the project entitled "**Image Clustering Using CNN**" is a bonafide work of "**Prathmesh Malvi (38), Pooja Naskar (30), Khanjan Joshi (26**) " submitted to the University of Mumbai in partial fulfillment of the requirement for the Course project in semester VII of Fourth Year **Computer Engineering**.

Supervisor
Dr. Megha Trivedi

Dr. Megha Trivedi
Head of Department

Contents

ABSTRACT

The accuracy of locating the human eye is crucial to face recognition technology. From one viewpoint, the human eye as one of the features of the facial picture which conveys a lot of data is a higher priority than the body parts, for example, the mouth as mouth. Then again, human eye position in the facial picture is generally fixed, and the face distance is many times utilized as the reason for normalizing the element vector in face feature extraction.

There are as of now a few compelling strategies for locating the eye. According to the transform method, for example, the pupil itself must be detected in a circular shape, or the eyelid's elliptical shape must be detected, which requires a large amount of calculation; the eye is positioned roughly using the projection method, and the deformation template method is applied based on the projection method. It is more complicated to position the eye with the deformation template method, which requires a lot of calculations and is sensitive to the original image size and illumination. In order to locate the human eye with edge detection, the method determines the upper half of the face first, and then the eye coordinates are determined by edge detection.

We are trying to implement the technique of image clustering is to form the clusters of the images so that the similar images can be sorted according to requirements. This process include the using of various Machine Learning Algorithms. The CNN Algorithm is used to extract the features of face and encode them into numeric values do that the clusters can we formed using the K-Means. The Overall System will able to sort the similar images from the group of images as per user requirements.

1. INTRODUCTION

1.1 Introduction

Image clustering is a widely used technique in the field of computer vision and image analysis that involves grouping similar images together based on certain criteria. It is a process of unsupervised learning that enables machines to automatically classify images without human intervention. This technique has a wide range of applications, including image retrieval, object recognition, image segmentation, and content-based image retrieval.

The primary objective of image clustering is to group similar images together and distinguish them from dissimilar ones. To achieve this, image clustering algorithms use different features, such as color, texture, shape, and other visual properties, to measure the similarity between images. Once the similarity is measured, clustering algorithms group similar images together into clusters, which can be used to organize and analyze large collections of images. Image clustering algorithms can be broadly classified into two categories: hierarchical and partitional. Hierarchical clustering algorithms create a tree-like structure of clusters by recursively dividing the dataset into smaller clusters until a certain stopping criterion is reached. Partitional clustering algorithms, on the other hand, partition the dataset into a predetermined number of clusters, typically using an iterative approach that minimizes a distance metric.

The mini project's contribution is to provide a accurate and reliable Image Clustering technique. By implementing various machine learning algorithm’s we have achieved the clusters of image so that it can become possible to extract required information through it.The classification can be done using various shapes of face and other spatial information. Overall, the mini project's contribution is to provide a practical and reliable solution for users seeking to manage their Sorted Image which is achieved using clustering technique.

1.2 Problem Statement & Objectives

Image clustering is a challenging task in computer vision and image analysis, particularly when dealing with large datasets. One of the main problems with is to specify the features so that the clusters of the faces can be formed according to requirements. Also, it is difficult to manage that only the required facial features can be extracted so that similar clusters can be formed of the same faces.

The primary objective of our clustering system is to group similar images together based on certain criteria, such as visual features, semantic content, or contextual information. This process can help to automate the organization and analysis of large collections of images, making it easier to identify and sort the similar faces of the same person.

1.3 SCOPE

The scope of image clustering is to form the clusters of the images so that the similar images can be sorted according to requirements. This process include the using of various algorithms such as CNN and K-Means. The CNN Algorithm is used to extract the features of face and encode them into numeric values do that the clusters can we formed using the Density Based Algorithm. The Overall System will able to sort the similar images from the group of images as per user requirements.

2. PROPOSED SYSTEM

2.1 Introduction

The system would need to perform feature extraction on each image in the dataset. This would involve identifying relevant visual features, such as texture, shape, and extracting them using appropriate algorithms and techniques. The system would need to calculate similarity between pairs of images based on their visual features. Similarity measurement could be achieved using techniques such as K-Means.Once similarity measurements have been obtained, the system would need to select an appropriate clustering algorithm that can effectively group similar images together based on the similarity measurements.

Common clustering algorithm K-Means clustering will be used for image clustering would that involve several key steps, including feature extraction, By effectively grouping similar images together, such a system could facilitate image retrieval, object recognition, and other applications in image analysis and computer vision. In the case of image clustering, this involves transforming the raw image data into a more manageable format, such as a feature vector. One popular approach is to use a convolutional neural network to extract features from the images, and then use these features as inputs to the clustering algorithm.

2.2 Architecture

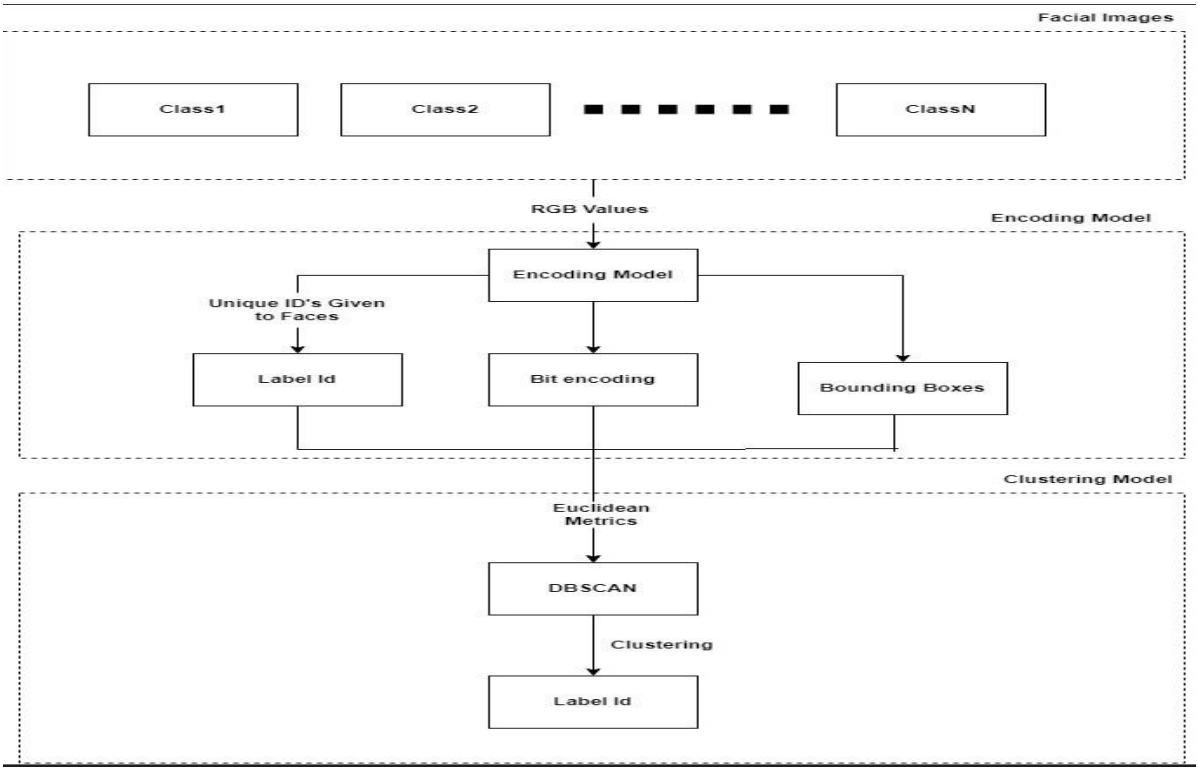


Fig. Architecture Of Model

1. Facial Images

This model takes input as images and to convert it into the various classes so that the images can be divided into various dimension. This classification is done by using RGB method. Then by forming the clusters the required features can we extracted to recognize the face. This output is transferred to the Encoding Model.

2. Encoding Model

The Encoding Model performs the encoding of various features into numerical values so that is can be easier to K-Means to form the clusters of images. This model assigns the unique Id to each given face.

- By using an ordinal encoder, the categorical feature is converted into numerical data using binary-encoding, which is a combination of one-hot encoding and Hash encoding.
- The bounding box is an imaginary rectangular box that contains an object or a set of points which bounding box refers to the border's coordinates that enclose an image.

3. Clustering Model

This model takes input as encoded values from encoding model of the various features of the face and forms the clusters using K-Means Algorithm. Through this clusters we can able to extract the various required features.

2.3 Algorithm

1. **Convolution Neural Network (CNN):**

For the recognition and analysis of images and videos, CNNs (Convolutional Neural Networks) are a type of deep learning algorithm. Conceptually, it relies on convolution, which involves processing small portions of an image, called kernels or filters, to extract features like edges, shapes , textures, and corners.

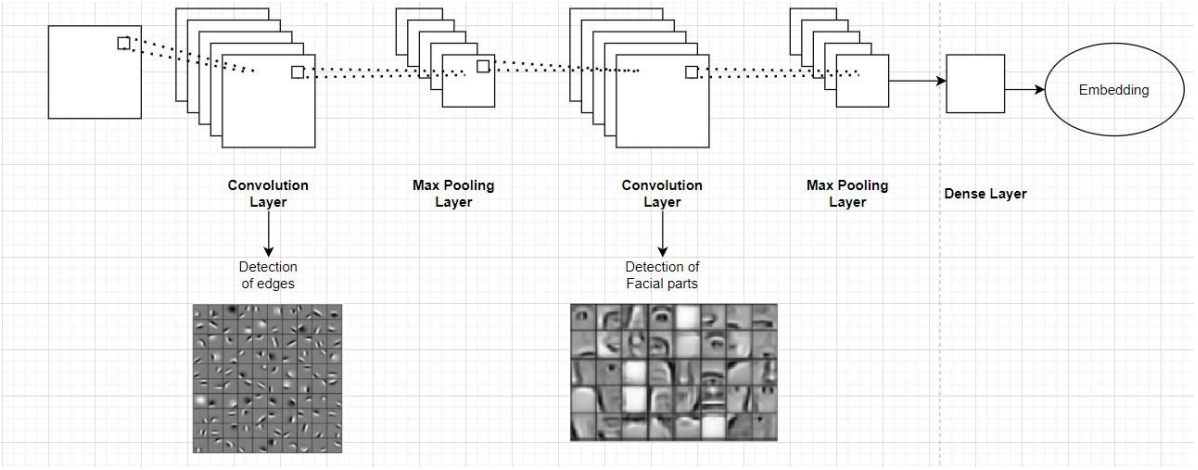


Fig. CNN Algorithm

In a CNN, the input layer takes in an image and passes it through several layers of convolution, pooling, and activation functions, where each layer is responsible for detecting and learning different features of the image. The output of the last layer is then flattened and fed into one or more fully connected layers that produce the final classification or prediction.

STEP 1

Convolutional Layer: The input image is passed through a CNN layer(convolution) that has a set of learnable filters to detect features in the picture. Each filter produces a two-dimensional activation map that highlights the existence of a feature in a picture.

STEP 2

Activation Function: The activation maps produced by the convolutional layer are passed through a non-linear activation function, such as ReLU, to introduce non-linearity into the model and improve its representational power.

STEP 3

Pooling Layer: Down sampling of activation maps is then performed using a pooling layer to reduce the spatial dimensions of the activation maps while preserving their important features. One of the common pooling technique is Max pooling that selects the maximum value in each pool.

STEP 4

Fully Connected Layer: The output of the pooling layer is passed through multiple fully connected layers and flattened into a one dimensional vector that apply a set of weights to produce a final output. These layers are also known as dense layers.

STEP 5

Softmax Activation: An Softmax activation function that converts the output into a probability distribution over the different classes is applied to the final output of the fully connected layer.

STEP 6

Loss Function: The output of the softmax activation is compared with the ground truth label using a loss function, such as categorical cross-entropy, to measure the difference between the predicted and actual class probabilities.

STEP 7

Backpropagation: The error signal produced by the loss function is backpropagated through the network to update the weights of the filters and fully connected layers using an optimization algorithm, such as stochastic gradient descent (SGD).

2.4 Hardware & Software requirements

2.4.1 Software requirements:

1. Jupyter Notebook
2. Visual Studio Code
3. Google Collab

2.4.2 Hardware requirements:

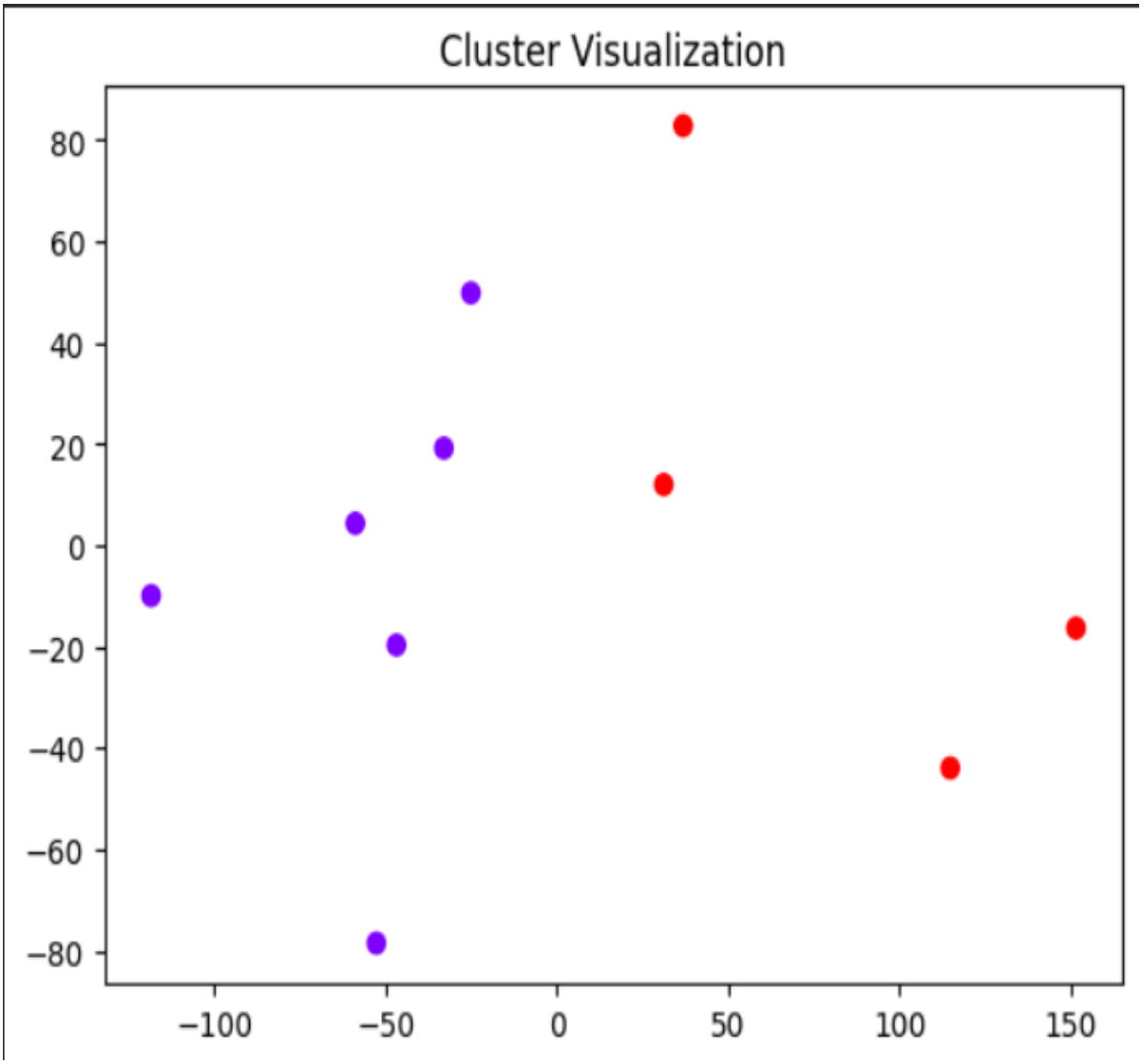
1. Windows 10 or higher (version 2004 or higher). Build 19041.xxx or higher
2. 64-bit machine (System type x64 based PC)

2.5 RESULT

Code:

```
import os
import cv2
import numpy as np
from sklearn.cluster import KMeans
from sklearn.decomposition import PCA
import matplotlib.pyplot as plt
from sklearn.metrics import silhouette_score
n_clusters = 2
image_dir = '/content/sample_data/final'
def load_and_preprocess_images(image_dir):
    images = []
    for filename in os.listdir(image_dir):
        if filename.endswith(".jpeg"):
            image_path = os.path.join(image_dir, filename)
            img = cv2.imread(image_path)
            img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
            img = cv2.resize(img, (224, 224))
            img = img / 255.0  # Normalize the pixel values
            images.append(img)
    return np.array(images)

# Load and preprocess images
images = load_and_preprocess_images(image_dir)

# Flatten the images to vectors
images_flat = images.reshape(images.shape[0], -1)

# Apply PCA for dimensionality reduction
pca = PCA(n_components=2)
images_pca = pca.fit_transform(images_flat)

# Apply K-Means clustering
kmeans = KMeans(n_clusters=n_clusters)
cluster_labels = kmeans.fit_predict(images_pca)

# Visualize the clusters
plt.scatter(images_pca[:, 0], images_pca[:, 1], c=cluster_labels, cmap='rainbow')
plt.title('Cluster Visualization')
plt.show()
```

Clusters:


Cluster Visualization
80
60
40
20
0
−20
−40
−60
−80
−100
−50
0
50
100
150

2.6 CONCLUSION AND FUTURE WORK

In conclusion, image clustering is a crucial technique in image analysis, computer vision, and machine learning. It provides a way to group similar images together based on visual features, semantic content, or contextual information, which can facilitate object recognition, image retrieval and image segmentation. Process of image clustering involves several steps, including There are still some challenges in image clustering, such as determining appropriate features to extract, selecting an appropriate clustering algorithm, and dealing with large datasets. Further research is needed to address these challenges and improve the performance of image clustering.

3. REFERENCES

1. Face Gesture Recognition Based on Clustering Algorithm | IEEE Conference Publication | IEEE Xplore.8833105

2. Tian Quan, Wang Yunzhe, Human Eye Positioning and State Recognition Based on Image Gray Feature, Computer Knowledge and Technology,2018,14(03):170-172.

3. https://saturncloud.io/blog/a-comprehensive-guide-to-convolutional-neural-networks-theeli5-way/

4. https://scikit-learn.org/stable/modules/generated/sklearn.cluster.DBSCAN