

K-means and Mean Shift report

Daquan Lin 85610653
ShanghaiTech University
Shanghai, China
lindq@shanghaitech.edu.cn

Abstract

This report introduces two approaches to cluster The Extended Yale Face Database B[4], they are K-means and Mean Shift. Due to the original face image set has so many unrecognizable face images, and a classmate found each info file has list 22 easy to recognized images for the person order from NO.1 to NO.10. I implemented Local binary patterns(LBP)[2],[1] to increase the Euclidean distance among the images from different people. The codes shown in K-means.ipynb and MeanShift.ipynb.

1. Introduction

Both K-means and Mean Shift algorithms are unsupervised model. For K-means, there are a lot of people doing face recognition[6],[5]. For Mean Shift, [3] show some work in variabel lighting Mean Shift face recognition.

In section 2, I will show how to extract the two datasets and some pretreatment. In section 3, I will discuss how to implement K-means model and some problem I faced and how to fix them. Then, I will compare the performances of two datasets. In section 4, I will talk about how to implement simple Mean Shift. Last, come with a conclusion.

2. Dataset Treatment

We can find each info file has list 22(it should be 23, but the face image named with some person's name is too dark to recognized.) easy to recognized images for the person order from NO.1 to NO.10. I extracted these clear face images and implemented LBP to decrease the effect of background, save them in path: /data2 as type of NumPy Array. I called this dataset Mini-CroppedYaleB(MCYB) dataset, it has 220 face images in total. Also, for all face images, I just resize their resolution to 42×48 , where original resolution is $168 \times$

192 mostly. I save them in path: /data as type of NumPy Array.

3. K-means Model

First, suppose the training dataset is \mathbf{X} , which shape is $N \times D$. N is number of training face images and D is flatted image row by row. I generated K cluster centers, each one has the same length with the training data, which has D dimension. Then, calculate the Euclidean distance between the each sample $\mathbf{x}_i, i = 1, \dots, N$ and each center $\mathbf{k}_j, j = 1, \dots, K$. So we will have a distance matrix with shape of $N \times K$, each entry is $dis_{i,j}$. I clustered the \mathbf{x}_i in set \mathbf{X}^j , which get minimum distance in same \mathbf{k}_j . And, calculate the new center \mathbf{k}'_j through calculate the mean of \mathbf{X}^j . Repeat do it, until 10 iterates or more.

$$dis_{i,j} = \|\mathbf{x}_i - \mathbf{k}_j\|, i = 1, \dots, N, j = 1, \dots, K. \quad (1)$$

$$\mathbf{X}_j = \mathbf{X}[\argmin_{j \in K} dis_{i,j}] \quad (2)$$

$$\mathbf{k}'_j = \frac{1}{M} \sum \mathbf{X}_j, M = \text{len}(\mathbf{X}_j), j = 1, \dots, K \quad (3)$$

When calculate the accuracy, I search each clusters \mathbf{X}_j to find the most frequent original label $y_p, p = 1, \dots, K$ and replace y_pred_j with y_p . Finally, we get the accuracy by $equ(5)$.

$$y_pred'_j = \text{MAX}[y_pred_j == y_p], j, p = 1, \dots, K \quad (4)$$

$$\text{accuracy} = \frac{1}{N} [y_pred' == y] \quad (5)$$

3.1. Two Datasets Results

In MCYB dataset, I get the accuracy is approximately 75%. While in original full dataset, it just get about 31.6%.

4. Mean Shift Model

In implement the Mean Shift Model, I chose a new center from the point that hasn't been visited, randomly. And group the neighbor point whose distance

to the center is small than a radius as a class, along with mark them have been visited. Then, calculate the mean shift vector through calculate the mean of vectors that all points in the interesting areas point to the center, untill the Euclidean distance of mean shift vector is small than a threshold. Update the center by add the mean shift vector.

If a new center isn't overlap with existance centers, then add a new cluster, or union them with same label and update the new center as mean of both of two centers.

The calculation of accuracy is almost the same as K-means model. However, there is a bit different from K-means, since the number of cluster isn't decided before running the model.

4.1. Two Dataset Results

In MCYB dataset, I get the accuracy is approximately 76.4%. While in original full dataset, it just get about 4.0%. It seems very terrible accuracy

5. Conclusion

I implement two methods to cluster the face image from the CroppedYaleB dataset. They are K-means and Mean Shift, respectively. In MCYB dataset, both of two model get accuracy up to 75%. However, in original dataset, the accuracies of two model has a huge gap. Since the Mean Shift is more powerful than K-means, I think it maybe lack of tuning the Mean Shift model or preprocess to the face images.

References

- [1] T. Ahonen, A. Hadid, and M. Pietikäinen. Face recognition with local binary patterns. In European conference on computer vision, pages 469–481. Springer, 2004.
- [2] T. Ahonen, A. Hadid, and M. Pietikainen. Face description with local binary patterns: Application to face recognition. IEEE transactions on pattern analysis and machine intelligence, 28(12):2037–2041, 2006.
- [3] T. Chen, W. Yin, X. S. Zhou, D. Comaniciu, and T. S. Huang. Total variation models for variable lighting face recognition. IEEE transactions on pattern analysis and machine intelligence, 28(9):1519–1524, 2006.
- [4] A. Georgiades, P. Belhumeur, and D. Kriegman. From few to many: Illumination cone models for face recognition under variable lighting and pose. IEEE Trans. Pattern Anal. Mach. Intelligence, 23(6):643–660, 2001.
- [5] M.-C. Su and C.-H. Chou. A modified version of the k-means algorithm with a distance based on cluster symmetry. IEEE Transactions on pattern analysis and machine intelligence, 23(6):674–680, 2001.
- [6] K.-K. Sung and T. Poggio. Example-based learning for view-based human face detection. IEEE Transactions

on pattern analysis and machine intelligence, 20(1):39–51, 1998.