Multimedia Tools and Applications (2021) 80:4719–4727
https://doi.org/10.1007/s11042-020-09822-5

Gabor face clustering using affinity propagation and structural similarity index

Issam Dagher[1] · Sandy Mikhael[1] · Oubaida Al-Khalil[1]

Received: 16 March 2020 / Revised: 31 July 2020 / Accepted: 2 September 2020 /
Published online: 1 October 2020

Abstract
Clustering is an important technique in data mining. It separates data points into different groups or clusters in such a way that objects in the same group are more similar to each other in some sense than with the objects in other groups. Gabor face clustering using affinity propagation and structural similarity index is composed of: A representation based on Gabor filters which has been shown to perform very well in face features, Affinity propagation clustering algorithm which is flexible, high speed, and does not require to specify the number of clusters, and structural similarity index which is a very powerful method for measuring the similarity between two images. Experimental results on two benchmark face datasets (LFW and IJB-B) show that our method outperforms well known clustering algorithms such as k-means, spectral clustering and Agglomerative.

Keywords Face clustering · Affinity propagation · Gabor · Structural similarity index

1 Introduction

Face clustering problem has found many applications nowadays in many emerging areas like forensic and surveillance applications, smart phones, and social media. It can be described as follows: Given a set of data or faces, clustering techniques separate them into different group. Each group shares some common characteristics (faces of the same person). The greater the similarity within the group (same person) and the greater the difference between groups (different persons), the better the clustering is. This problem can be tackled by different approaches. Different face representations can be considered and different cluster algorithms can be applied. Good model should be used to represent a face and good cluster technique with powerful similarity measure should be be applied on this model in order to get the best possible clustering of a set of faces.

✉ Issam Dagher
dagheri@balamand.edu.lb

[1] Computer Engineering Department, University of Balamand, Tripoli, Lebanon

1.1 Related work

In this paper, we have attempted to handle the face clustering problem by using a set of Gabor filters as a face representation method and by using an effective face clustering algorithm (Affinity propagation) which is capable of automatically grouping the faces. We replaced the popular Euclidian similarity measure by a more powerful one which is the Structure Similarity Measure.

Different clustering techniques use different similarity measures. K-means [19] algorithm is the simplest clustering algorithm. It minimizes the within-cluster sum of squares (Euclidian similarity measure). The number of clusters should be specified. Hierarchical clustering algorithms fall into 2 categories: top-down or bottom-up. Bottom-up algorithms treat each data point as a single cluster at the outset and then successively merge (or *agglomerate*) pairs of clusters until all clusters have been merged into a single cluster that contains all data points. Bottom-up hierarchical clustering is therefore called *hierarchical agglomerative clustering* or *HAC*. This hierarchy of clusters is represented as a tree (or dendrogram). In Spectral clustering [26] the problem of clustering is reformulated using the similarity graph: We want to find a partition of the graph such that the edges corresponding to different clusters have a very low weight (which means that points in different clusters are dissimilar from each other) and the edges within a group have higher weight (which means that points within the same cluster are similar to each other). Different clustering techniques were applied to face clustering.

In [6] spectral clustering with local gradients was developed. They used Yale-B and PIE-66 datasets. 2D-HMM and hierarchical clustering were used in [23]. They used a database of 1500 face images of 8 persons. LBP face-features and color and texture, body-features were used in [3]. They used spectral clustering. They used a database of 400 face images of 5 persons. In [15] the authors used probabilistic clustering. Hierarchical clustering using rank-order distance function was applied in [27]. Learning base descriptors are developed in [2]. Subspace clustering was applied in [16]. They used the extended Yale-B database. Wang et al. K-NN graph construction method was done in [18]. [1] and [11] used hierarchical clustering to organize the face database. [14] applied their face clustering on video frames. Deep learning was used in [12]. Clustering billion of images with large scale nearest neighbor was used in [9].

1.2 Contributions

Our contributions in the Face clustering problem are the following:

- Selection of a fast, and an accurate clustering algorithm: Affinity propagation algorithm which does not require to specify a priori the number of clusters.
- Selection of an accurate image similarity measure: structural similarity index (SSIM) which is a perception-based model that considers image degradation as perceived change in structural information, while also incorporating important perceptual phenomena, including both luminance masking and contrast masking terms.
- Selection of an accurate face features: Gabor filters which model the responses of simple cells in the primary visual cortex by extracting important features in an image from different orientation and different scales.

2 Background

2.1 Affinity propagation

Affinity propagation (AP) is a clustering algorithm based on the concept of "message passing" between data points [4]. It can also support similarities that are not symmetric or do not satisfy the triangle inequality. It is deterministic and does not require the number of clusters that must be determined and estimated before running the algorithm. A collection of real-valued similarities between data points is considered as input to the "affinity propagation" approach. The most common similarity measure is the negative squared error (Euclidean distance). For points 'x_i' and 'x_k', $s(i,k) = -||x_i - x_k||^2$ is taken as inputs. The values of s(k,k) are usually taken the median of the input similarities .

There are two kinds of message exchanged between data points:

- The "responsibility" matrix **R** has values $r(i, k)$ that quantify how well-suited x_k is to serve as the exemplar for x_i, relative to other candidate exemplars for x_i.
- The "availability" matrix **A** contains values $a(i, k)$ that represent how "appropriate" it would be for x_i to pick x_k as its exemplar, taking into account other points' preference for x_k as an exemplar.

Both matrices are initialized to all zeroes.

The algorithm then performs the following updates recursively (Eqs. 1 and 2).

$$r(i,k) \leftarrow s(i,k) - \underset{k' \neq k}{max}\left\{a\left(i,k^{'}\right) + s\left(i,k^{'}\right)\right\} \tag{1}$$

$$a(i,k) \leftarrow \min\left(0, r(k,k) + \sum\nolimits_{i' \notin \{i,k\}} \max\left(0, r\left(i^{'},k\right)\right)\right) \tag{2}$$
$$\textit{for } i \neq k \textit{ and}$$

To limit the influence of strong incoming positive responsibilities, the total sum is thresholded so that it cannot go above zero. The "self-availability" a(k,k) is uploaded differently (Eq. 3):

$$a(k,k) \leftarrow \sum_{i' \neq k} \max\left(0, r\left(i^{'},k\right)\right) \tag{3}$$

2.2 Structural similarity index

SSIM [17] is primarily used in measuring the similarity between two images. This method allows the metric to adapt to the local statistical characteristics at different image locations. It is characterized by very good accuracy and low complexity.

The Structural Similarity Index quality assessment index is a computation of three terms, namely the luminance term, the contrast term and the structural term. It is given by Eq. (4).

$$\mathrm{SSIM}(x,y) = l(x,y)^{\alpha} \cdot c(x,y)^{\beta} \cdot s(x,y)^{\gamma} \tag{4}$$

$$\text{Where } \ l(x,y) = \frac{2\mu x\mu y + C1}{\mu^2 x + \mu^2 y + C1} \tag{5}$$

$$c(x,y) = \frac{2\sigma x\sigma y + C2}{\sigma^2 x + \sigma^2 y + C2} \tag{6}$$

$$s(x,y) = \frac{\sigma\mathrm{xy} + \mathrm{C3}}{\sigma\mathrm{x}\sigma\mathrm{y} + \mathrm{C3}} \tag{7}$$

If $\alpha = \beta = \gamma = 1$ and $\mathrm{C3} = \mathrm{C2}/2$ the index simplifies to the Eq. (8):

$$SSIM(x,y) = \frac{(2\mu x\mu y + \ \ C1)(2\sigma xy + C2)}{(\mu^2 x + \mu^2\ y + C1)(\sigma^2 x + \sigma^2 y + C2)} \tag{8}$$

2.3 Gabor filtering

Gabor filters [5] are a set of band pass filters used to extract important features in an image. The idea is to convolve the image with a set of filters to extract structures with different sizes and orientations. Gabor filter is a sinusoidal plane of a particular frequency and orientation, modulated by a Gaussian envelope. It can be represented by the Eq. (9)

$$h(x,y) = g\left(x^{'},y^{'}\right)\exp\left(2\pi i f x^{'}\right) \tag{9}$$

Where $g(x', y')$ is the 2d Gaussian given by Eq. (10)

$$g\left(x^{'},y^{'}\right) = \exp\left[-\frac{1}{2}\ {}^{x'^2+y'^2}\!\Big/_{\sigma^2}\right] \tag{10}$$

$(x', y') = (x\cos\theta + y\sin\theta, -x\cos\theta + y\sin\theta\ )$ are the rotated coordinates (x, y) by θ.

The parameters f and θ represent the frequency and orientation of the signal. In the time domain, Gabor function is represented by Eq. (11):

$$h(x,y) = \frac{1}{2\pi\overset{\text{Ç}}{\sigma}}\exp\left[-\frac{1}{2}\left(\frac{\overset{\text{Ç}}{x} + \overset{\text{Ç}}{y}}{\overset{\text{Ç}}{\sigma}}\right)\right]\cos(2\pi fx) \tag{11}$$

The Fourier transform of the Gabor filter is given by Eq. (12)

$$H(u,v) = \exp\left[-2\overset{\text{Ç}}{\pi}\overset{\text{Ç}}{\sigma}\left((u{-}f) + \overset{\text{Ç}}{v}\right)\right] + \exp\left[-2\overset{\text{Ç}}{\pi}\overset{\text{Ç}}{\sigma}\left((u+f) + \overset{\text{Ç}}{v}\right)\right] \tag{12}$$

When a Gabor filter is applied to an image, it gives the highest response at the edges and at points where the texture changes.

The θ parameter decides what kind of features the filter responds to. For example, giving theta a value of zero means that the filter is responsive only to horizontal features only. So, in order to obtain features at various angles in an image, we divide the interval between 0 and 180 into 16 equal parts, and compute a Gabor kernel for each value of theta thus obtained.

The wavelength governs the width of the strips of Gabor function. Increasing the wavelength produces thicker stripes and decreasing the wavelength produces thinner stripes. Keeping other parameters unchanged and changing the lambda to 60 and 100, the stripes get thicker.

Figure (1) shows the Gabor filters for 8 orientations and 5 wavelengths.

2.4 F-measure

To assess the performance of the clustering algorithms, we have used the F-measure [22]. It can be described as follows:

- Positive (P): same cluster.
- Negative (N): different clusters.
- True (T): same class.
- False (F): different classes.
- TP: Number of item pairs that are in the same cluster and belong to the same class.
- FP: Number of item pairs that are in the same cluster but belong to different classes.
- TN: Number of item pairs that are in different clusters and belong to different classes.
- FN: Number of item pairs that are in different clusters, but belong to the same class.

Precision (Eq. (13)) is calculated as the fraction of pairs correctly put in the same **cluster**, **recall** (Eq. (14)) is the fraction of actual pairs that were identified, and F-measure (Eq. (15)) is the harmonic mean of **precision and recall**.

$$precision = \frac{TP}{TP + FP} \tag{13}$$

$$recall = \frac{TP}{TP + FN} \tag{14}$$

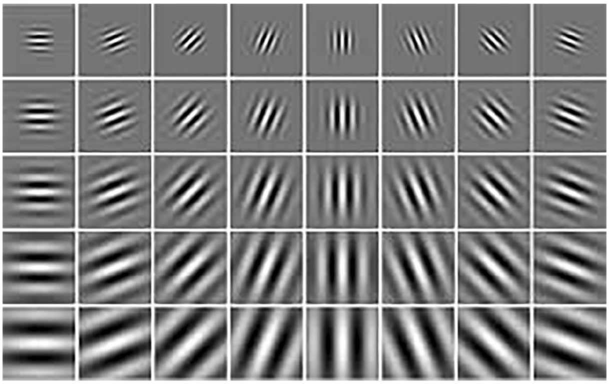

Fig. 1 Gabor filters for 8 orientations and 5 wavelengths

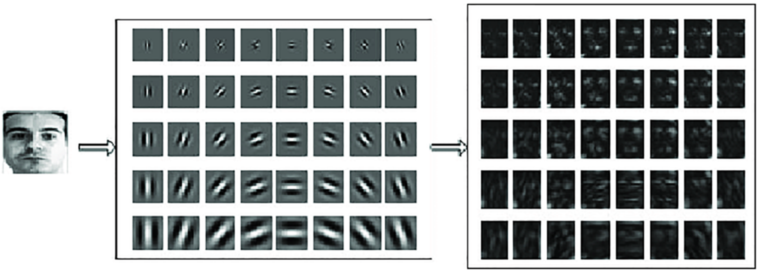

Fig. 2 Gabor outputs for 8 orientations and 5 wavelengths

$$F = \frac{2*precision*recall}{precision + recall} \tag{15}$$

3 Algorithm

Our algorithm can be summarized as follows:

1) For each face image face(x,y): Find the Gabor outputs for 8 orientations and 5 wavelengths. Every MxN face image will be represented by an augmented size M1xN1 image facegabor(x,y). Where M1 = 8 M and N1 = 5 N. This can be shown in Fig. 2.
2) Find the STRUCTURAL SIMILARITY INDEX (SSIM) among all the resulting images:
   s(i,k) = SSIM(facegabor(i), facegabor(k))
3) Apply the Affinity propagation algorithm on the augmented size images.
4) Get the face clusters and apply the F-measure.

4 Experiments

To show the effectiveness of our approach, we have compared our approach with:

Table 1 Comparison of our algorithm and other clustering algorithms on the LFW dataset

Algorithm	LFW database	# of clusters
k-means	0.098	5749
k-means	0.346	500
Spectral Clustering	0.033	5749
Spectral Clustering	0.257	75
Agglomerative	0.962	4500
Affinity Propagation	0.320	1203
ConPac [13]	0.965	6352
Our algorithm	0.967	6376

Table 2 Comparison of our algorithm and other clustering algorithms on the IJBB128

Algorithm	IJBB128 database	# of clusters
k-means	0.520	75
Spectral Clustering	0.377	100
Agglomerative	0.787	150
Affinity Propagation	0.230	351
ConPac [13]	0.895	738
Our algorithm	0.901	676

- K-means which requires the specification of the number of clusters.
- Spectral clustering also requires that the number of clusters to be specified a priori.
- Agglomerative clustering doesnot require the specification of the number of clusters.
- Affinity propagation using Euclidian distance as similarity measure.
- Conditional Pairwise Clustering (ConPac).
- And our approach which uses the SSIM as similarity measure.

We have used the following databases:

- Labeled Faces in the Wild (LFW) [7]. This database contains 13,233 images of 5749 subjects. We have used the entire database. The results are shown in Table 1.
- IARPA JANUS B (IJB-B) [20]: TheIJB-B dataset contains 11,754 images of 1845 subjects. We have used 2 subsets (IJBB128, IJBB1024) with a number of subjects respectively 128, 1024. The results are shown in Tables 2 and 3.
- The Extended Yale-B face database [10] contains 38 persons where images (frontal faces) were captured under 9 different illuminations. The results are shown in Table 4.

For K-means and spectral clustering, you need to specify a priori the number of clusters. For example In Table 1 corresponding to the LFW database, we have chosen 2 values of K. The first one is 5749 which the total number of Subjects (gave an F-measure = 0.098) and the other one is K = 500 and gave an F-measure of 0.346. We followed the same approach for the Spectral Clustering algorithm.

The other 4 techniques (Agglomerative, Affinity Propagation, ConPac, and our algorithm) will output automatically the number of clusters. We have also compared the usual Affinity Propagation which is applied directly to the images with our approach. Our approach gave the biggest value of the F-measure.

Table 3 Comparison of our algorithm and other clustering algorithms on the IJBB1024

Algorithm	IJBB1024 database	# of clusters
k-means	0.423	1000
Spectral Clustering	0.265	750
Agglomerative	0.544	1500
Affinity Propagation	0.241	2500
ConPac [13]	0.667	11,258
Our algorithm	0.669	9557

Table 4 Comparison of our algorithm and other clustering algorithms on the extended Yale database

Algorithm	Yale database	# of clusters
k-means	0.380	200
Spectral Clustering	0.301	160
Agglomerative	0.621	298
Affinity Propagation	0.256	250
Our algorithm	0.875	525

All of our simulations were done using Matlab R2017a on an Intel(R) core™ i7–4510 CPU 2Ghz. The additional time complexity of our algorithm compared to the regular Affinity propagation algorithm is the computation of the output of the Gabor filter which is 0.101 s per image.

5 Conclusions and future directions

In this paper, we proposed a novel clustering algorithm which we called GABOR FACE CLUSTERING USING AFFINITY PROPAGATION AND STRUCTURAL SIMILARITY INDEX. It clusters face images without knowing a priori the number of persons. It is based on: a representation using a set of Gabor filters Affinity propagation clustering and structural similarity index, which is a very powerful method for measuring the similarity between two images. To show the effectiveness of our approach, we compared it to other clustering algorithms on challenging face databases. In future directions, we will extend our ideas and skills to other multimedia applications, such as feature selection [8], image segmentation [21], and haze removal [24, 25].

References

1. Bhattarai B, Sharma G, Jurie F, P'erez P (2014) "Some faces are more equal than others: Hierarchical organization for accurate and efficient large-scale identity-based face retrieval," in ECCV Workshops, pp. 160–172. 2
2. Cao Z, Yin Q, Tang X, Sun J (2010) "Face recognition with learning based descriptor," in Proc. CVPR. IEEE, pp. 2707–2714. 2
3. Cui J, Wen F, Xiao R, Tian Y, Tang X (2007) "Easyalbum: an interactive photo annotation system based on face clustering and re-ranking," in Proc. of the SIGCHI conference on Human factors in computing systems. ACM, pp. 367–376. 2
4. Dueck D (2009) 'Affinity propagation: clustering data by passing messages', PhD, University of Toronto
5. Geng Du, Gong Liming Fei Su (2009). "An effective Gabor-feature selection method for face recognition," 2009 IEEE International Conference on Network Infrastructure and Digital Content, Beijing, pp. 722–725
6. Ho J, Yang MH, Lim J, Lee KC, Kriegman D (2003) "Clustering appearances of objects under varying illumination conditions," in Proc. CVPR IEEE. 2, 3
7. Huang GB, Mattar M, Berg T, Learned-Miller E (2008) Labeled faces in the wild: a database for studying face recognition in unconstrained environments. In Workshop on Faces in Real-Life Images: Detection, Alignment, and Recognition. 5
8. Li H, He F, Liang Y, Quan Q (2020) A dividing-based many-objective evolutionary algorithm for large-scale feature selection. Soft Comput 24:6851–6870
9. Liu T, Rosenberg C, Rowley HA (2007) "Clustering billions of images with large scale nearest neighbor search," in Proc. WACV. IEEE, pp. 28–28. 3
10. Mart'ınez A, Benavente R (1998) "The extended AR Face Database,"Technical Report 24, Computer Vision Center, Bellatera,Barcelona, Spain

11. Sarwas G, Skoneczny S (2018) “FSIFT based feature points for face hierarchical clustering,” 2018 signal processing: algorithms, architectures, arrangements, and applications (SPA), Poznan, pp. 55-58. L
12. Schroff F, Kalenichenko D, Philbin J (2015) “Facenet: A unified embedding for face recognition and clustering,” in Proc. CVPR. 3
13. Shi Y, Otto C, Jain AK (2018) Face clustering: representation and pairwise constraints. in IEEE Transactions on Information Forensics and Security 13(7):1626–1640
14. Tapaswi M, Parkhi OM, Rahtu E, Sommerlade E, Stiefelhagen R, Zisserman A (2014) “Total cluster: A person agnostic clustering method for broadcast videos,” in Proc. of the Indian Conference on Computer Vision Graphics and Image Processing. ACM, p. 7. 2
15. Tian Y, Liu W, Xiao R, Wen F, Tang X (2007) “A face annotation framework with partial clustering and interactive labeling,” in Proc. CVPR IEEE. 2
16. Vidal R, Favaro P (2014) Low rank subspace clustering (lrsc). Pattern Recogn Lett 43(2):47–61
17. Wang Z, Bovik AC, Sheikh HR, Simoncelli EP (2004) Image quality assessment: from error visibility to structural similarity. in IEEE Transactions on Image Processing 13(4):600–612
18. Wang J, Wang J, Zeng G, Tu Z, Gan R, Li S (2012) “Scalable k-NN graph construction for visual descriptors,” in Proc. CVPR. IEEE, pp. 1106–1113. 2
19. Wei P, Zhou Z, Li L, Jiang J (2018) Research on face feature extraction based on K-mean algorithm. J Image Video Proc 2018:83
20. Whitelam C, Taborsky E, Blanton A, Maze B, Adams J,Miller T, Kalka N, Jain AK, Duncan JA, Allen K, Cheney J, Grother P (2017) Iarpa janus benchmark-b face dataset.In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, July. 6
21. Yu H, He F, Pan Y (2020) A scalable region-based level set method using adaptive bilateral filter for noisy image segmentation. Multimed Tools Appl 79:5743–5765
22. Zerabi S, Meshoul S (2017) “External clustering validation in big data context,” 2017 3rd International Conference of Cloud Computing Technologies and Applications (CloudTech), Rabat, pp. 1–6
23. Zha M, Teo Y, Liu S, Chua T, Jain R (2006) “Automatic person annotation of family photo album,” in Image and Video Retrieval. Springer, pp. 163–172. 2, 3
24. Zhang S, He F (2019) DRCDN: learning deep residual convolutional dehazing networks. Vis Comput
25. Zhang J, He F, Chen Y (2020) A new haze removal approach for sky/river alike scenes based on external and internal clues. Multimed Tools Appl 79:2085–2107
26. Zhang C Li, H Wang, Shi M (2018), “Research On Face Image Clustering Based On Integrating Som And Spectral Clustering Algorithm,” 2018 International conference on machine learning and cybernetics (ICMLC), Chengdu, pp. 337–342
27. Zhu C, Wen F, Sun J (2011). “A rank-order distance based clustering algorithm for face tagging,” in Proc. CVPR. IEEE, pp. 481–488. 1, 2, 3, 4