Article

Adaptive Facial Imagery Clustering via Spectral Clustering and Reinforcement Learning

Chengxiao Shen, Liping Qian * and Ningning Yu

Department of Information Engineering, Zhejiang University of Technology, Hangzhou 310023, China; shencx_zjut@163.com (C.S.); nnyu_zjut@163.com (N.Y.)
* Correspondence: lpqian@zjut.edu.cn

Abstract: In an era of big data, face images captured in social media and forensic investigations, etc., generally lack labels, while the number of identities (clusters) may range from a few dozen to thousands. Therefore, it is of practical importance to cluster a large number of unlabeled face images into an efficient range of identities or even the exact identities, which can avoid image labeling by hand. Here, we propose adaptive facial imagery clustering that involves face representations, spectral clustering, and reinforcement learning (Q-learning). First, we use a deep convolutional neural network (DCNN) to generate face representations, and we adopt a spectral clustering model to construct a similarity matrix and achieve clustering partition. Then, we use an internal evaluation measure (the Davies–Bouldin index) to evaluate the clustering quality. Finally, we adopt Q-learning as the feedback module to build a dynamic multiparameter debugging process. The experimental results on the ORL Face Database show the effectiveness of our method in terms of an optimal number of clusters of 39, which is almost the actual number of 40 clusters; our method can achieve 99.2% clustering accuracy. Subsequent studies should focus on reducing the computational complexity of dealing with more face images.

Keywords: adaptive clustering; face clustering; face feature extraction; reinforcement learning

1. Introduction

With the rapid development of computer vision and pattern recognition, face images, as one of the most common formats of visual information, provide a wide range of application prospects. Face images can be captured in many fields, including social media, forensic investigations, and surveillance monitoring, etc. [1–3]. However, these captured face images generally lack labels, which leads to difficulties in identifying them. The unknown identities (clusters) can range from a few dozen to thousands, and how to locate an efficient range or even an exact range raises an essential problem [4]. In addition, implementation of modern clustering algorithms [5,6] is required to ensure the number of clusters as well as some adjustable parameters in advance. Therefore, it is important to design an efficient facial imagery clustering algorithm for image labeling.

Recently, face image clustering has attracted a lot of interest from researchers [7–11]. L. Zhang et al. [7] presented a clustering method based on a spectral clustering and self-organizing feature mapping (SOM) neural network. C. Qi et al. [8] proposed a deep face clustering method based on a residual graph convolutional network (RGCN). M.S. Abdallah et al. [9] proposed a TV media mining system based on a DCNN to rapidly identify a specific individual in real-time processing video data. I. Ahn et al. [10] proposed a multiple segmentation technique combined with constrained spectral clustering to label facial images containing objects with complicated boundaries. C. Otto et al. [11] proposed an approximate rank-order clustering algorithm for large-scale face image clustering. Despite the tremendous efforts that have been devoted to image clustering, most previous studies have not adapted to the illuminance, pose, and complicated background of images.

In addition, existing studies have failed to estimate the number of clusters and the values of adjustable parameters.

In general, face clustering involves face representation and a clustering model. Face representation is generally a numeric vector descriptor of a face image. In the literature [12–15], most traditional hand-crafted face representations often lack robustness with respect to adverse factors within the imagery. The quality of the captured images is low due to the factors of illuminance, pose, and complicated background, etc. M. Versaci et al. [16] presented a fuzzy geometrical approach based on unit hypercubes for image contrast enhancement to solve this problem. L. Wang et al. [17] learned the discriminative target feature by aligning the feature domain globally, while distinguishing the target clusters locally. Their method reduced the impact of illuminance, pose, and image quality. We adopt the significant progress that has been achieved in unconstrained face representation [18–20], which can alleviate the influence of underlying factors by using a convolutional neural network-based face representation. Spectral clustering is an example of graph-based clustering algorithms. As compared with other clustering algorithms such as hierarchical clustering methods and K-means, spectral clustering has more robustness with respect to the uncertainty of data sparsity and distribution. Therefore, we use spectral clustering as the partition model. The main aim of this study is to propose an adaptive facial imagery clustering method, which mainly involves face representations, a spectral clustering model, and reinforcement learning (i.e., Q-learning used in this study) [21]. With Q-learning, the clustering model can adaptively search and find the optimal setting for the number of clusters and dimension reduction (i.e., the core adjustable parameter) in the spectral clustering model, which helps to dramatically improve the performance of clustering.

The remainder of this paper is organized as follows: In Section 2, we present the details of adaptive clustering; in Section 3, we briefly introduce the IMM Face Database and provide some simulations to show the effectiveness of our proposed method; finally, conclusions and future work are presented in Section 4.

2. Proposed Adaptive Clustering

In this section, we describe the proposed adaptive facial imagery clustering, which includes the following four components: face representations, spectral clustering, performance evaluation, and a feedback module (Q-learning), as shown in Figure 1. In the following subsections, we introduce each aspect in detail.

2.1. Face Representations

Since face images are generally captured under unconstrained situations, traditional face representations lack universal applicability. Considering the superior performance of deep convolutional neural networks (DCNNs) on feature extraction and representation, we leverage a DCNN for our face representation. The specific implementation of face representation includes facial landmark detection and deep face representation, as illustrated in Figure 2, and described in detail below.
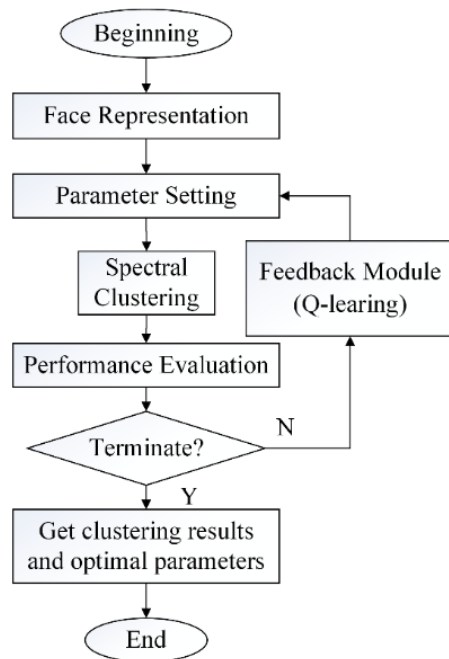
Figure 1. The diagram of our proposed adaptive facial imagery clustering. First, the face feature vectors are obtained. Second, the images are clustered according to the set parameters. Third, the parameters are adjusted by evaluating the clustering results. Finally, we repeat the second and third steps, and output the clustering results and parameters when the results meet our set requirements.
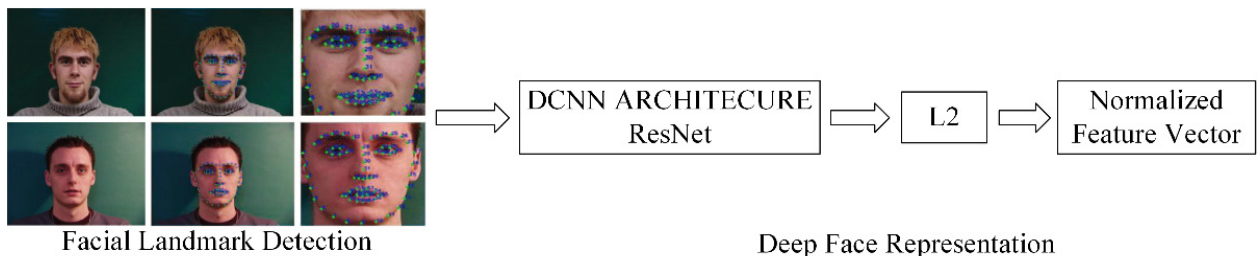
Figure 2. The diagram of the deep face representation. First, we determine the 68 landmark points of the face and crop them according to the landmark points. Then, we input the image and landmark points into the ResNet model, Finally, we use L2-normalization to obtain the normalized feature vector.

Facial landmark detection, also known as face alignment, aims to align faces into the mean face shapes based on facial keypoints (mouth, eye, canthus, etc.). In this way, the positions of these keypoints on all faces after alignment are almost the same. The face representations based on these keypoints extracted by a DCNN is more effective and representative, even if lower dimensions of the representations are adopted to cluster the faces. We adopt the success of the DLIB implementation of Kazemi and Sullivan's method [22], which uses 68 landmarks of the input image to detect a face and return the position of facial region.

After facial landmark detection, the aligned face is the input of the DCNN to obtain its deep face representation. Specifically, we employ deep architecture ResNet1 pretrained on DLIB, which finally outputs a 128-dimension feature vector. Then, we use L2-normalization [23] to normalize the vector, thus, each dimension of the feature vector is normalized into a specific interval. In addition, the use of L2 also improves the performance of the clustering model. After L2, the 128-dimension normalized vector is the final representation of the face.

2.2. Spectral Clustering

Spectral clustering is an algorithm that evolved from graph theory and has been widely used in different fields such as speech recognition, document analysis, and image segmentation. Implementation of the spectral clustering model includes construction of a similarity matrix and clustering partition. The detailed process is as follows:

- Give the face representations $(X_1, X_2, \ldots, X_n)$, and calculate the similarity matrix S based on the full connection method.
- Construct the adjacency matrix W based on S, and then calculate the degree matrix D.
- Calculate the Laplacian matrix L: $L = D - W$
- Construct the normalized Laplacian matrix L^*: $L^* = D^{-1/2}LD^{-1/2}$
- Create a set of eigenvectors by computing the eigenvectors of L^* and use dim eigenvectors corresponding to the largest dim eigenvalues. The dim is an adjustable parameter, which must be less than the dimension of the representation X.
- Create a new face representation matrix F, which uses these dim eigenvectors as columns; these eigenvectors are also normalized by L2.
- With the new representation matrix F, the final clustering result is generated by using K-means++.

In the above steps, we use the Gaussian Dadial Basis Function to construct S and W as:

$$s_{ij} = s_{ji} = \sum_{i=1,j=1}^{n} \exp\left(-\frac{\|X_i - X_j\|^2}{2\sigma^2}\right), (i, j \in 1, 2, \ldots, n) \tag{1}$$

where σ is the radial width parameter, w_{ij} is the adjoin metric and s_{ij} is the similarity metric of X_i and X_j, respectively.

2.3. Performance Evaluation

For most practical image clustering, face images generally lack labels or the labels are incomplete. Therefore, we use the Davies–Bouldin index (DBI) as the internal evaluation measure to evaluate the clustering performance. The DBI is defined as:

$$DBI = \frac{1}{K} \sum_{i=1}^{K} \max_{i \neq j} \left(\frac{avg(C_i) + avg(C_j)}{D_U(C_i, C_j)}\right) \tag{2}$$

where K is the number of clusters, $D_U(C_i, C_j)$ is the distance of centers of C_i and C_j, U is the center vector of cluster C, and $avg(C)$ is the average distance within the cluster C. Obviously, a smaller DBI value means better clustering performance with better intracluster compactness and intercluster separation.

2.4. Feedback Module

The feedback module adjusts the parameters based on the performance evaluation to optimize the clustering performance. For this, we adopt Q-learning to debug parameters adaptively. For this paper to be self-contained, first, we introduce the key idea of Q-learning before applying it to the parameter updating.

2.4.1. Q-Learning

Reinforcement learning is a goal-directed learning methodology, which constructs a control strategy to maximize the behavior performance of the agent. Specifically, the agent can require status information and reward signals from a complex environment. The agent is not advised which actions to take, but aims to find the best action that brings the best reward by trying all actions. In addition, the agent can use knowledge learned in the past to guide future actions. The sketch map of reinforcement learning is shown in Figure 3.
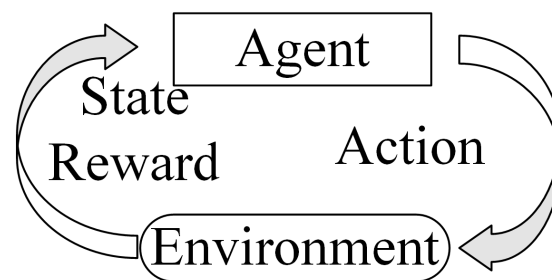
Figure 3. The sketch of reinforcement learning, in which the agent guides the choice of actions according to the state feedback information given by the environment, and the choice of actions affects the changes of the environment, thereby updating the parameters.

The environment model is not necessary because reinforcement learning also supports a model-free algorithm such as Q-learning that can gradually update the optimal choice of actions to reach the best state with reward incentives. The process is denoted as the Bellman Equation (3):

$$Q(s,a) \leftarrow (1 - \alpha)Q(s,a) + \alpha[r + \gamma \max_{a'} Q(s',a')] \qquad (3)$$

where $Q(\cdot, \cdot)$ is the state quality of all possible actions (Q-table), s is the state, a is the action, r is the reward signal, α is the learning rate, γ is the discount index, s' is the state of the next moment, and a' is the action of the next moment.

2.4.2. The Application of Q-Learning

In our proposed method, we can deem the different parameter settings as actions and the performance evaluation (*DBI*) as the reward. The state of the clustering system is constantly updating by trying the possible parameter settings whose clustering performance can be evaluated with the *DBI*. With the reward incentive of the optimization to make the *DBI* minimum, the dynamic debugging process can be implemented using a Q-learning algorithm.

In the clustering, there are two parameters for setting: the clustering number (*K*) and dimension reduction (dim). Each parameter is given three types of debugging, that is, increasing, decreasing, and remaining unchanged. Therefore, we obtain nine types of parameter debugging actions in each state of Q-learning. Each action contains two independent debugging actions of the two parameters. We set the step values λ and μ ($\lambda, \mu > 0$) for *K* and dim, respectively, to denote the parameter debugging steps in the debugging. Because the parameters may vary beyond their range, the corresponding debugging is set to remain unchanged if the parameter debugging is out of range at the next state.

The aim of the actions is to make the performance evaluation the *DBI* minimum. Therefore, the clustering system must give positive incentives when the action facilitates a reduction in the *DBI*. For an inapparent change in the *DBI*, we give tiny incentives, even no incentives. For a bad action which increases the *DBI*, we give a negative incentive. Due to the random initialization of K-means++, the observations of the *DBI* have small fluctuations for the same action. Therefore, we repeat the observations of the *DBI* for each action ten times and use the largest *DBI* value as the final *DBI*. The process is given as:

$$D(a) = \max D_i(a), (i = 1, 2, \ldots, 10) \qquad (4)$$

where $D(a)$ is the final evaluated *DBI* with the action a, and $D_i(a)$ is the i-th observation of the clustering process. We define the variable quantity Δ of the *DBI* as:

$$\Delta = D - D' \qquad (5)$$

where D is the *DBI* of the current status, and D' is the *DBI* of the next status. Therefore, the reward r can be expressed as:

$$r = \begin{cases} \sqrt{|\Delta|}, \Delta \in (\varepsilon, +\infty) \\ -\sqrt{|\Delta|}, \Delta \in (-\infty, -\varepsilon) \\ 0, \Delta \in [-\varepsilon, \varepsilon] \end{cases} \tag{6}$$

where r is the reward signal and ε ($\varepsilon > 0$) is a positive threshold.

To prevent the algorithm converging to the local minimum region, we use the β-greedy strategy to improve Q-learning. Specifically, we use $\Gamma(a|s)$ to denote the selection probability function for each action in Q-learning.

$$\Gamma(a|s) = \begin{cases} 1 - \beta + \frac{\beta}{N_a}, & a = \underset{a}{\mathrm{argmax}}Q(s, a) \\ \frac{\beta}{N_a} & a \neq \underset{a}{\mathrm{argmax}}Q(s, a) \end{cases} \tag{7}$$

where β ($0 < \beta < 1$) is a greedy coefficient and N_a is the number of actions. Different from the strategy in which Q-learning always chooses the best action in each state, the β-greedy strategy has β/N_a probability of choosing the remaining nonoptimal actions. In doing this, Q-learning enhances the exploration of unknown states, which may be potential global optimal states.

In addition, we terminate the Q-learning algorithm when we have $\Delta = 0$ for multiple consecutive states. In this study, we set the number of threshold termination states as10. The Algorithm 1 pseudo program diagram is as follows.

Algorithm 1: The Q-learning applied in our method.

Input: K, *dim*, λ, μ, β, ε
Output: K, *dim*, Δ
Initialize $Q(s, a)$
Repeat
Initialize counter $i = 0$
While $i < 10$ do
Choose a from s using the policy derived form Q with the β-greedy strategy;
take action a, and observe r and s' by
$Q(s, a) \leftarrow (1 - \alpha)Q(s, a) + \alpha[r + \gamma \max Q(s', a')]$;
Parameter updating: $K = K \pm \lambda$, dim $=$ dim $\pm \mu$;
Calculate Δ, and generate reward signals r;
State updating: $s' \leftarrow s$.
If $r = 0$ then
$i = i + 1$
Else
$i = 0$

---

3. Experiments

In this section, we use two face datasets to test our proposed algorithm. We scrambled all face images and placed them in the same folder, and then used the clustering algorithm to classify each image label, and finally compared the original label of the dataset to obtain the final accuracy. First, we introduce the face databases used, and then test our algorithm in four parts.

In the first part, we adopt the manual investigations of the parameters based on the variable-controlling approach, and the results prove that the *DBI* reaches a minimum value when the correct clustering parameters are set, therefore, the reinforcement learning method can be used to obtain this optimal value.

In the second part, we guide automatic parametric settings based on manual search results, and use the asynchronous length to perform the parameter settings test, which

proves the feasibility of our algorithm, and reduces the parameter setting time by two times the asynchronous long search.

In the third part, we compare our final clustering results with other algorithms, and the F-score is used as the evaluation index. The comparison result proves that our algorithm clustering can achieve high accuracy, which is better than other algorithms. In addition, our algorithm can automatically set important parameters such as the number of clusters, thereby greatly reducing the time spent on parameter modulation in the clustering process.

Finally, we test the robustness of the algorithm on larger wild face data.

3.1. Datasets

In this study, the ORL [24] and CFP [25] Face Databases are used to evaluate our proposed adaptive face clustering algorithm. The ORL Face Database contains 40 people of different ages, skin colors, and genders. Each person has 10 images with different lighting, postures, and expressions. There are glasses occluded in some of the images, and the pixels of each image are 92×110, such as the examples of face images shown in Figure 4.

Figure 4. Example face images from the IMM Face Database.

The CFP Face Database contains 500 unconstrained images of different people. The pixel size of each image is not the same, which makes it difficult to extract the feature vectors of the faces. As compared with the above face images collected in a laboratory environment, these images are obtained from the Internet. Therefore, the expression, lighting, and posture of the same person's image are quite different, and the face occlusion situation is more serious. We use 10 images that are close to the front of each person. Figure 5 is an example of some pictures in the CFP Face Database.
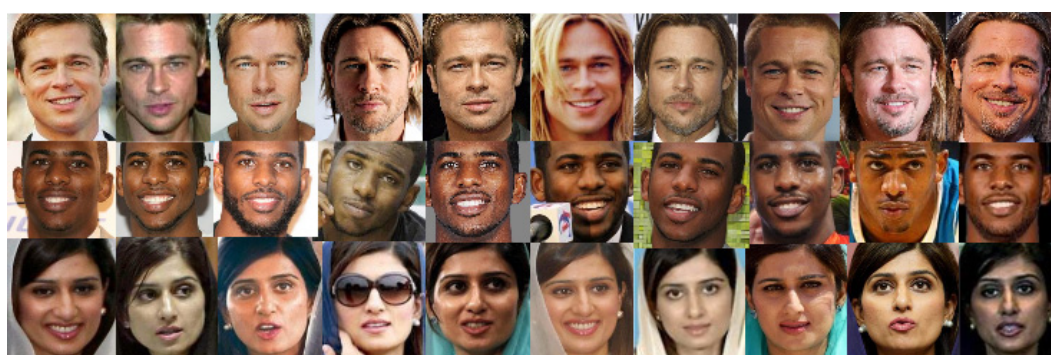

Figure 5. Example face images from the CFP Face Database.

3.2. Manual Investigation of Parameter Settings

We assume that the best number of clusters is unknown. However, for 400 facial images, we reasonably consider that they are divided into no more than 80 categories (i.e., the expected maximum number of clusters). Thus, the number of clusters *K* ranges from 0 to 80. In addition, the dimension reduction of spectral clustering should not exceed the original maximum dimension and retain the primary information of the face representations. Generally, it is reasonable to reserve no more than 50% of the original dimensions, that is to say, due to the original dimension 128, the dimension reduction (dim) can range from 0 to 60.

We observe several settings of *K* and dim using the variable-controlling approach. Specifically, when one parameter changes within its range, the other is an invariable parameter. Due to this investigation over a wide range of parameters, we make a rough global observation in which the interval of the changing parameters is five and the invariable parameter is 10. The observation results of the manual investigations are illustrated in Figures 6 and 7.
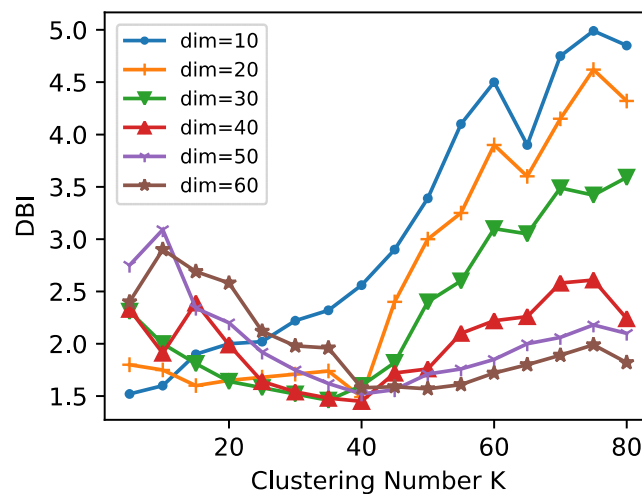

Figure 6. The obtained *DBI* under the different settings of cluster number K.

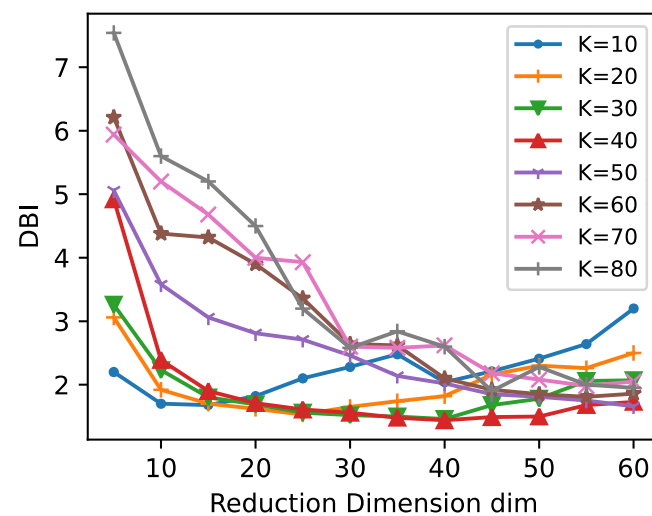**Figure 7.** The observation of dimension reduction dim. Dim is regarded as the changing parameter, while *K* is the invariable parameter. When $K \in [10, 40]$, first, the *DBI* value decreases, and then increases with an increase in the dim value. When $K \in [50, 80]$, the *DBI* value decreases with an increase in the dim value.

As shown in Figure 6, different parameter settings result in the difference of the performance evaluation *DBI*. Specifically, the optimal value of K is about 40 because most of the dimension settings at this value make the clustering performance better (smaller *DBI*). In fact, this conclusion is of practical significance due to the fact that the ORL Face Database is indeed comprised of 40 different persons.

As shown in Figure 7, there are some valuable conclusions. First, the clustering performance deteriorates shapely when the reserved dimension of the face representations is less than 15. Second, more reserved dimensional information does not mean that the performance of clustering is better. On the contrary, it becomes worse, especially when dim is over 50 in most curves. Third, we can see that the red curve, which has the invariable parameter setting $K = 40$, in fact, corresponds to the optimal setting of K. The curve indicates that changing dim has less impact on the ideal K within the range of dim from 20 to 45. It implies that the closer K gets to the optimal value, the less dim changes. Therefore, it requires a smaller step size in the optimal field when we use searching methods for the parameter setting.

3.3. Adaptive Face Clustering

Our proposed adaptive face clustering achieves the optimal parameter setting by Q-learning. To reduce the runtime and computational complexity, we follow the principle that the clustering takes the smaller value within the range as the initial searching point.

3.3.1. Algorithm Complexity

The computational complexity of the clustering mainly depends on the spectral clustering and parameter adjustment. Let O_K and O_{dim}, respectively, denote the computational complexity of parameter searching K and parameter searching dim, satisfying:

$$O_K = \frac{K - K_0}{\lambda}\left[O_Q(\text{dim}^2 Km) + O_{DBI}(\text{dim}Km)\right] \tag{8}$$

$$O_{\text{dim}} = \frac{\text{dim} - \text{dim}_0}{\mu}\left[O_Q(\text{dim}^2 Km) + O_{DBI}(\text{dim}Km)\right] \tag{9}$$

Here, K_0 is the starting point clustering number, dim_0 is the starting point dimension reduction, λ and μ are step values, and m is the number of face images. O_Q is the computational complexity of spectral clustering, O_{DBI} in the computational complexity of the Davies–Bouldin index.

When K and dim increase, the time and space complexity of the system are shapely increasing, and the starting points have a greater impact on the algorithm. Therefore, it is necessary to run the searching algorithm with a smaller initial value within the searching range.

3.3.2. Parameter Searching

In this simulation, we want to compare the performance of rough searching and exact searching, and then perform exact searching on the basis of the rough searching results to speed up the search. For the rough searching, we search the optimal fields of K and dim synchronously with a large step. The system is trained 10 times from the same starting point to update the Q-table. The initial parameters of the Q-learning are shown in Table 1, second column.

Table 1. Initial parameters of Q-learning for searching.

| Parameters	Rough Observation	Exact Observation
Starting point K	10	30
Starting point dim	10	30
Step values of λ and μ	5	1
Given threshold ε	0.05	0.01
Learning rate α	0.5	0.2
Discount index γ	0.5	0.2
Greedy coefficient β	0.2	0.2
Maximum iteration	50	50
Training times	10	10

For the exact searching, we search the exact setting of *K* and dim synchronously with a small step in the optimal field. The initial parameters of the Q-learning are shown in Table 1, third column. The searching paths of *K* and dim are plotted in Figures 8 and 9. In order to show the convergence of the algorithm, the threshold interrupt algorithm is not used.
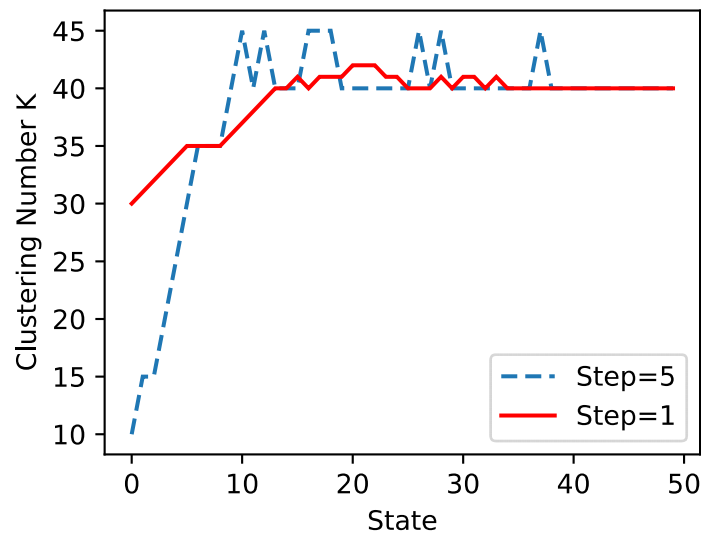
Figure 8. The path of searching for the optimal clustering number K.
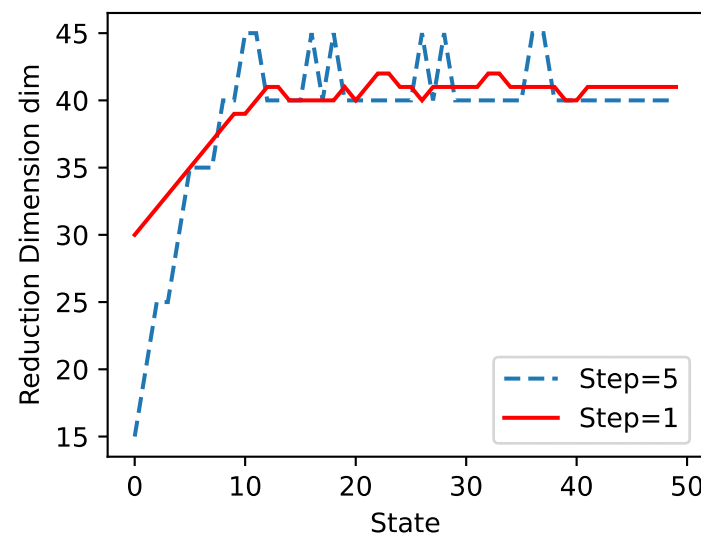
Figure 9. The path of searching for the optimal clustering number dim.

As shown in Figures 8 and 9, the searching path of the adaptive clustering system is convergent. In rough searching, the system converges to the field in which the K and dim are both 40 as the center. Therefore, we acquire the optimal field based on the center and the large searching step as $K, \dim \in [35, 45]$. In exact searching, the results indicate that the optimal setting of K and dim is 41. We analyze some valid reasons why we obtain the clustering number of 41 instead of 40 for the ORL. First, the difference in the *DBI* value within the optimal field is inapparent, which causes inefficiency of the rewards in the Q-learning. Second, the value of the *DBI* rests with the face representations. Therefore, the architectures of the DCNN lead to representations that are not unique for the same face. Third, the given threshold ε is too large, which is set, in advance, as small as possible but does not always meet the expected effect.

We use a combination of rough searching and exact searching to speed up the clustering. The final adaptive clustering takes 77.01 s, and each spectral clustering takes about 0.185 s. However, manual clustering parameter adjustment requires multiple comparisons of all image clustering results, and then multiple adjustments to the parameters. This process requires a lot of time and effort.

3.4. Comparison of Clustering Accuracy

On the ORL Face Database, we use an adaptive algorithm to obtain clustering results, and compare the results with other algorithms, including k-means, LGC [26], and AEWLP [27]. We use dlib to extract the facial features in the image, and then we use k-means clustering to add the clustering results to the comparison. Using F-score as a comprehensive index for evaluating the accuracy of clustering, it calculates the harmonic average of the precision rate P and the recall rate R, as shown in Formula (8). The accuracy rate P refers to the prediction result, which represents the proportion of the correct prediction being positive to the total prediction being positive. The recall rate R is for the original sample, which represents the proportion of the correct prediction that is positive to all that is actually positive. We use scikit-learn and MTALAB to reproduce the above algorithm and F-score, respectively:

$$F_1 = \frac{2PR}{P + R} = \frac{2TP}{2TP + FP + FN} \tag{10}$$

where TP is true positive, FP is true negative, and FN is false negative. Since random parameters are used in the clustering of the k-means algorithm, we evaluate the results of multiple clustering and take the average. The final results are shown in Table 2.

Table 2. Initial parameters of q-learning for searching.

	k-Means	LGC	AFWLP	Dlib-k-Means	Ours
1	0.743	0.840	0.960	0.976	0.986
2	0.856	0.950	0.960	0.996	0.986
3	0.776	0.940	0.960	0.923	0.993
4	0.783	0.950	1	0.942	1
5	0.696	0.980	1	0.915	0.993
Avg	0.771	0.932	0.976	0.950	0.992

It can be seen from the above table that our clustering effect can exceed other algorithms, and the F-score can reach 99.2%, indicating that only 2–3 face images in the ORL's 400 images will be classified into the wrong category. Comparing the result of directly using k-means for clustering and the result of using dlib for face feature extraction, and then using k-means clustering, the accuracy has been greatly improved, therefore, showing the effectiveness of the feature extraction technology we used.

3.5. CFP Face Database Clustering Test

We use 5000 images in the CFP Face Database for the clustering test. We mainly search for the number of clusters automatically, because as the number of identities in the data increases, the number of required features also continue to increase, as shown in Figure 7.

We randomly select 50, 100, 200, 300, 400 and all human images in the dataset for adaptive clustering, and compare the F-score value calculated by the clustering results with the k-means algorithm. As shown in Table 3, the clustering accuracy of our algorithm is still higher than the k-means algorithm, and it can automatically search for the appropriate number of clusters K. As the number of clusters increases, there is still some deviation between the number of searched clusters K and the actual number of people.

**Table 3.** Clustering test results for the CFP Face Database.

Actual Number	Clustering Number K	F-Score	
		Ours	**K-Means**
50	50	0.9912	0.7756
100	100	0.9775	0.6458
200	206	0.8357	0.5026
300	314	0.7750	0.4647
400	420	0.7563	0.4421
500	532	0.7483	0.4367

The images with incorrect clustering are presented separately, as shown in Figure 10. There is a huge age gap between different images of the same person, leading to morphological differences and making clustering errors. The algorithm judges young images into different categories to increase the number of clusters. However, the untagged images acquired through video surveillance, medical equipment, and social media over a period of time are basically the same age. The deviation in the number of clusters caused by age changes is not a big problem.

true   false   true   false   true   false

Figure 10. The image clustering performance at different ages.

Figure 11 shows the path for searching on the CFP Face Database of 5000 images. The number of clusters is large, so the first step is 20 for rough searching, but it is difficult to converge in the final result. Because a large step size will cause a large change in the number of clusters, and then a large change in the *DBI* value. However, the optimal domain range of the number of clusters 450, 580 can be obtained through a rough search. Then, the final optimal number of clusters is 532 through an exact search with a step size of two. Although there are some deviations, it proves that the clustering parameter adaptive algorithm we designed can still obtain a suitable number of clusters. The F-score calculated for the final clustering result is 0.7483, which is better than 0.4367 obtained by the k-means algorithm.
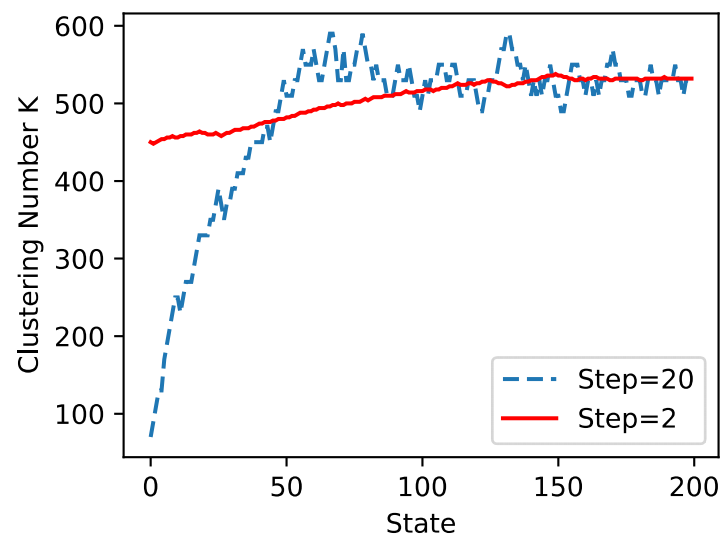
Figure 11. The path of searching for the optimal clustering number K on the CFP Face Database.

We also tested the method on the LFW face dataset [28], which contained 13,233 images of 5749 people. First, we deleted the images that contained multiple faces in the same image. The 12,323 images of the remaining 5484 people are clustered, and the final cluster number K is 5622, and the F-score is 0.6492. Because the data samples provided in the LFW dataset are not balanced (the number of images for each person is different), the clustering accuracy is not high.

We conducted a test on the YouTube Faces Dataset [29]. First, the video was intercepted into images at 20-frame intervals, and images containing multiple faces and no faces were removed through face detection. While obtaining enough face images, we also ensured that there would be a certain amount of deflection angle, illumination, and other changes between the images of the same person. In the end, we constructed a dataset of 27,650 images containing 1595 people. Using the proposed approach to cluster the dataset, the final cluster number K is 1684, and the F-score is 0.7724. Although the number of images increases, accuracy is better than the results achieved on the CFP dataset, because the changes between images of the same person captured in a video are small.

From the above test, it can be found that when the number of identities and the number of images increase, the clustering accuracy decreases. Therefore, in practical applications, a large number of face images can be split into multiple folders, and then clustered. Automatic searching of clustering parameters can save a lot of parameter adjustment time in the clustering process, because for a large number of unlabeled face images, manual parameter adjustment needs to compare and confirm the clustering results one by one, and then adjust the clustering parameters. This step needs to be performed many times, and it is difficult to adjust manually when the number of images reaches thousands. Therefore, the adaptive clustering algorithm we proposed still has application value.

4. Conclusions

In this study, we proposed an adaptive face clustering method via a spectral clustering model and reinforcement learning. The method involved face representations, spectral clustering, and Q-learning. First, we used a DCNN to obtain the deep representations of face images. Then, we used the spectral clustering algorithm to achieve the partition of these face representations. To cluster these representations efficiently, we used internal clustering measures to evaluate the clustering quality. To seek the optimal parameters for the unknown identity datasets, we employed Q-learning as the feedback module to build the dynamic multparameter debugging process. The simulation results on the ORL and CFP Face Databases demonstrated that our proposed method effectively searched the optimal parameters, which helped to improve the clustering quality and accuracy. Due to

the use of reinforcement learning to realize the parameter search, it can effectively cluster unknown face datasets and can reduce the time of manual parameter modulation.

In future studies, we plan to focus on the inefficiency of the exact searching within the optimal field. In addition, we plan to pay more attention to reducing the running complexity of our method and modifying the clustering method to improve the accuracy of image clustering for large amounts of data.

Author Contributions: Conceptualization, C.S. and N.Y.; methodology, C.S. and N.Y.; software, C.S.; validation, C.S. and N.Y.; investigation, C.S. and N.Y.; writing—original draft preparation, C.S.; writing—review and editing, N.Y. and L.Q.; visualization, L.Q.; supervision, L.Q.; project administration, C.S.; funding acquisition, L.Q. All authors have read and agreed to the published version of the manuscript.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from http://www.cfpw.io/ and https://www.cst.cam.ac.uk/, accessed on 11 August 2021.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Akcay, S.; Kundegorski, M.E.; Willcocks, C.G.; Breckon, T.P. Using Deep Convolutional Neural Network Architectures for Object Classification and Detection within X-ray Baggage Security Imagery. *IEEE Trans. Inf. Forensics Secur.* **2018**, *13*, 2203–2215. [CrossRef]
2. Nie, D.; Wang, L.; Adeli, E.; Lao, C.; Lin, W.; Shen, D. 3-D Fully Convolutional Networks for Multimodal Isointense Infant Brain Image Segmentation. *IEEE Trans. Cybern.* **2019**, *49*, 1123–1136. [CrossRef] [PubMed]
3. Do, T.; Cheung, N. Embedding Based on Function Approximation for Large Scale Image Search. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 626–638. [CrossRef] [PubMed]
4. Zhou, S.; Xu, Z.; Liu, F. Method for Determining the Optimal Number of Clusters Based on Agglomerative Hierarchical Clustering. *IEEE Trans. Neural Netw. Learn. Syst.* **2017**, *28*, 3007–3017. [CrossRef]
5. Ayed, A.B.; Halima, M.B.; Alimi, A.M. Adaptive fuzzy exponent cluster ensemble system based feature selection and spectral clustering. In Proceedings of the 2017 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), Naples, Italy, 9–12 July 2017; pp. 1–6.
6. Havens, T.C.; Bezdek, J.C.; Leckie, C.; Hall, L.O.; Palaniswami, M. Fuzzy c-Means Algorithms for Very Large Data. *IEEE Trans. Fuzzy Syst.* **2012**, *20*, 1130–1146. [CrossRef]
7. Zhang, L.F.; Li, C.F.; Wang, H.R.; Shi, M.Y. Research On Face Image Clustering Based On Integrating Som And Spectral Clustering Algorithm. In Proceedings of the 2018 International Conference on Machine Learning and Cybernetics (ICMLC), Chengdu, China, 15–18 July 2018.
8. Qi, C.; Zhang, J.; Jia, H.; Mao, Q.; Song, H.J.K.-B.S. Deep face clustering using residual graph convolutional network. *Knowl. Based Syst.* **2021**, *211*, 106561. [CrossRef]
9. Abdallah, M.S.; Kim, H.W.; Ragab, M.E.; Hemayed, E.E.J.E. Zero-Shot Deep Learning for Media Mining: Person Spotting and Face Clustering in Video Big Data. *Electronics* **2019**, *8*, 1394. [CrossRef]
10. Ahn, I.; Kim, C. Face and Hair Region Labeling Using Semi-Supervised Spectral Clustering-Based Multiple Segmentations. *IEEE Trans. Multimed.* **2016**, *18*, 1414–1421. [CrossRef]
11. Otto, C.; Wang, D.; Jain, A.K. Clustering Millions of Faces by Identity. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 289–303. [CrossRef]
12. Ameur, B.; Masmoudi, S.; Derbel, A.G.; Hamida, A.B. Fusing Gabor and LBP feature sets for KNN and SRC-based face recognition. In Proceedings of the 2016 2nd International Conference on Advanced Technologies for Signal and Image Processing (ATSIP), Monastir, Tunisia, 21–23 March 2016; pp. 453–458.
13. Lu, J.; Liong, V.E.; Zhou, X.; Zhou, J. Learning Compact Binary Face Descriptor for Face Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 2041–2056. [CrossRef]
14. Lee, S.H.; Choi, J.Y.; Ro, Y.M.; Plataniotis, K.N. Local Color Vector Binary Patterns From Multichannel Face Images for Face Recognition. *IEEE Trans. Image Process.* **2012**, *21*, 2347–2353. [CrossRef] [PubMed]
15. Arashloo, S.R.; Kittler, J.; Christmas, W. Face Spoofing Detection Based on Multiple Descriptor Fusion Using Multiscale Dynamic Binarized Statistical Image Features. *IEEE Trans. Inf. Forensics Secur.* **2015**, *10*, 2396–2407. [CrossRef]

16. Versaci, M.; Calcagno, S.; Morabito, F.C. Fuzzy geometrical approach based on unit hyper-cubes for image contrast enhancement. In Proceedings of the 2015 IEEE International Conference on Signal and Image Processing Applications (ICSIPA), Kuala Lumpur, Malaysia, 19–21 October 2015; pp. 488–493.

17. Wang, M.; Deng, W.J.N. Deep face recognition with clustering based domain adaptation. *Knowl.-Based Syst.* **2020**, *393*, 1–14. [CrossRef]

18. Lu, J.; Wang, G.; Deng, W.; Jia, K. Reconstruction-Based Metric Learning for Unconstrained Face Verification. *IEEE Trans. Inf. Forensics Secur.* **2015**, *10*, 79–89. [CrossRef]

19. Liao, S.; Jain, A.K.; Li, S.Z. A Fast and Accurate Unconstrained Face Detector. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *38*, 211–223. [CrossRef] [PubMed]

20. Wang, D.; Otto, C.; Jain, A.K. Face Search at Scale. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1122–1136. [CrossRef] [PubMed]

21. Pandey, D.; Pandey, P. Approximate Q-Learning: An Introduction. In Proceedings of the 2010 Second International Conference on Machine Learning and Computing, Bangalore, India, 9–11 February 2010; pp. 317–320.

22. Kazemi, V.; Sullivan, J. One millisecond face alignment with an ensemble of regression trees. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 1867–1874.

23. Dai, Z.; Chen, W.; Huang, X.; Li, B.; Zhu, L.; He, L.; Guan, Y.; Zhang, H. CNN Descriptor Improvement Based on L2-Normalization and Feature Pooling for Patch Classification. In Proceedings of the 2018 IEEE International Conference on Robotics and Biomimetics (ROBIO), Kuala Lumpur, Malaysia, 12–15 December 2018; pp. 144–149.

24. Samaria, F.S.; Harter, A.C. Parameterisation of a stochastic model for human face identification. In Proceedings of the 1994 IEEE Workshop on Applications of Computer Vision, Sarasota, FL, USA, 5–7 December 1994; pp. 138–142.

25. Sengupta, S.; Chen, J.; Castillo, C.; Patel, V.M.; Chellappa, R.; Jacobs, D.W. Frontal to profile face verification in the wild. In Proceedings of the 2016 IEEE Winter Conference on Applications of Computer Vision (WACV), Lake Placid, NY, USA, 7–10 March 2016; pp. 1–9.

26. Zhou, D.; Bousquet, O.; Lal, T.N.; Weston, J.; Schölkopf, B. Learning with Local and Global Consistency. In *Advances in Neural Information Processing Systems*; Max Planck Institute for Biological Cybernetics: Tuebingen, Germany, 2004; Volume 16.

27. Karasuyama, M.; Mamitsuka, H. Manifold-based Similarity Adaptation for Label Propagation. *Adv. Neural Inf. Process. Syst.* **2013**, *26*, 1547–1555.

28. Huang, G.B.; Mattar, M.; Berg, T.; Learned-Miller, E.J.M. *Labeled Faces in the Wild: A Database forStudying Face Recognition in Unconstrained Environments*; Technical Report 07-49; University of Massachusetts: Amherst, MA, USA, 2007.

29. Wolf, L.; Hassner, T.; Maoz, I. Face recognition in unconstrained videos with matched background similarity. In Proceedings of the CVPR 2011, Colorado Springs, CO, USA, 20–25 June 2011; pp. 529–534.