CLIP-Cluster: CLIP-Guided Attribute Hallucination for Face Clustering

Shuai Shen[1,2], Wanhua Li[1,2], Xiaobing Wang[3], Dafeng Zhang[3], Zhezhu Jin[3], Jie Zhou[1,2], Jiwen Lu[1,2,*]
[1]Department of Automation, Tsinghua University, China
[2]Beijing National Research Center for Information Science and Technology, China
[3]Samsung Research China-Beijing (SRC-B)
shens19@mails.tsinghua.edu.cn; wanhua016@gmail.com;
{x0106.wang, dfeng.zhang, zz777.jin}@samsung.com; {jzhou, lujiwen}@tsinghua.edu.cn;

Abstract

One of the most important yet rarely studied challenges for supervised face clustering is the large intra-class variance caused by different face attributes such as age, pose, and expression. Images of the same identity but with different face attributes usually tend to be clustered into different sub-clusters. For the first time, we proposed an attribute hallucination framework named CLIP-Cluster to address this issue, which first hallucinates multiple representations for different attributes with the powerful CLIP model and then pools them by learning neighbor-adaptive attention. Specifically, CLIP-Cluster first introduces a text-driven attribute hallucination module, which allows one to use natural language as the interface to hallucinate novel attributes for a given face image based on the well-aligned image-language CLIP space. Furthermore, we develop a neighbor-aware proxy generator that fuses the features describing various attributes into a proxy feature to build a bridge among different sub-clusters and reduce the intra-class variance. The proxy feature is generated by adaptively attending to the hallucinated visual features and the source one based on the local neighbor information. On this basis, a graph built with the proxy representations is used for subsequent clustering operations. Extensive experiments show our proposed approach outperforms state-of-the-art face clustering methods with high inference efficiency.

1. Introduction

Recent years have witnessed the remarkable success of face clustering technology [5, 21, 45, 47, 49, 52, 63], due to the progress in deep learning frameworks [17, 40, 41] and more available large-scale training data [4, 13, 24, 64]. However, among the existing research, the influences of different face attributes on clustering have been poorly investigated. Since facial appearance changes dramatically under different factors like age, pose, and expression [14, 31], how to minimize the effects of these facial variations for more compact intra-cluster face embedding is still an open challenge for face clustering.

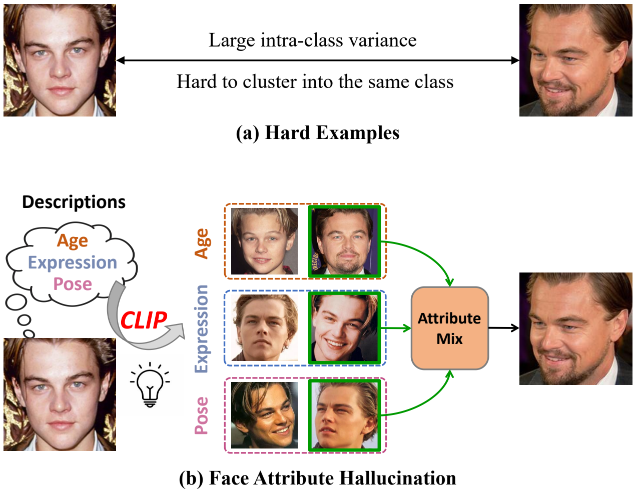
Large intra-class variance
Hard to cluster into the same class
(a) Hard Examples
Descriptions
Age
Expression
Pose
CLIP
Age
Expression
Pose
Attribute Mix
(b) Face Attribute Hallucination

Figure 1. (a) shows paired hard examples, where aging brings significant appearance changes, and it is difficult for the network to aggregate them into the same cluster. (b) visualizes the CLIP-guided text-driven face attribute hallucination, which is the core idea of the proposed CLIP-Cluster.

Face clustering has received thorough attention in the computer vision research area. More recently, the supervised face clustering methods are extensively studied [5, 27, 36, 47, 51, 57] and have achieved significant performance gain. While they put more attention on how to estimate vertice confidence [12, 51], construct neighbor linkage [45, 47], or mine graph structure [36, 53], few works systematically study the impact of intra-cluster variance on face clustering. Intra-class face variance is a common issue in both collected face datasets and real-world photos, and its impact on feature discrimination has actually been repeat-

[*] Corresponding author.

edly verified in the face recognition community [22, 44]. Even so, since facial appearance changes dramatically under different face attributes including age, pose, and expression, it is still challenging to effectively mitigate the impact of these variances. As the hard example shown in Figure 1 (a), for the same person, aging brings significant appearance changes, and it is difficult for the network to aggregate them into the same cluster.

In this work, we are committed to narrowing the intra-cluster attribute gap through a synthesis-like approach for easier face clustering. One possible solution is to synthesize face images through attribute manipulation, thus face samples are transferred to a uniform attribute space. However, training such networks with mainstream generative prototypes [11, 15] for controllable attribute editing has the following limitations: 1) high requirement for large-scale annotated paired data, 2) high training difficulty for generating realistic images in the pixel space, and 3) additional computational cost of extracting features for synthesized images for subsequent face clustering. For these reasons, we choose to generate faces for different attributes directly in the feature space, which is termed *attribute hallucination*.

In this work, we propose CLIP-Cluster, which leverages the recent powerful language-visual model CLIP [32] for text-driven face attribute hallucination, and opens a brand-new avenue for face clustering. The core idea of the CLIP-Cluster is shown in Figure 1 (b). With CLIP-guided text-driven face attribute hallucination, we can transfer faces toward various ages, poses, and expressions. Benefiting from the admirable zero-shot image classification capability of CLIP, we can get rid of the demand for a large amount of annotated paired data, and turn to more convenient text-based manipulation. Since CLIP maintains a well-aligned language-image embedding space, we directly perform attribute transfer in the latent space with identity-preserving supervisions, and use these features for subsequent face clustering. In this way, we naturally avoid the additional computational cost of synthesis in pixel space and extracting features for the generated images. Furthermore, while a face is transferred across various attributes, we design a neighbor-aware proxy generator to fuse them into a proxy feature by learning the neighbor-adaptive attention. This builds a bridge among different sub-clusters and reduces the intra-class variance. With these proxy representations to construct the affinity graph, the subsequent GCN-based edge predictor can perform face clustering in an easier way.

Extensive experiments show that the proposed CLIP-Cluster significantly boosts the face clustering performance on standard partial MS1M from 93.22 to 94.22 pairwise F-score, and the inference process can be completed efficiently within 280s. In summary, the main contributions of this work are as follows:

- We propose a CLIP-guided text-driven face attribute hallucination framework to bridge the large intra-class attribute gap. Therefore, face clustering can be performed in a more compact embedding space.
- Furthermore, we develop a neighbor-aware proxy generator that fuses the features describing various attributes into a proxy by learning the neighbor-adaptive attention to reduce the intra-class variance.
- The proposed CLIP-Cluster improves F_P on partial MS1M to 94.22 within 280s, which outperforms state-of-the-art methods with high inference efficiency.

2. Related Work

Face Clustering. Face clustering is a promising approach in a series of application scenarios including data annotation, file grouping, and photo organization. Traditional algorithms [10, 20, 23, 28, 38, 60, 62] usually rely on manually defined heuristic clustering strategies, which provide valuable theoretical foundations for follow-up works. Recently, more face clustering research turns to supervised-based learning [57], with *GCN* [47, 52, 59] and *Transformer* [12, 27] as the basic technology, and has achieved remarkable performance. These existing methods cope with the face clustering problem from different perspectives like vertice confidence estimation [12, 51], neighbor linkage prediction [45, 47], and graph structure mining [36, 37, 53]. Yang *et al.* [51] develop the global-based GCN learning paradigm for clustering. Wang *et al.* [47] utilize the local context information to predict the linkage relationship. Shen *et al.* [36] propose a structure-aware face clustering approach making it possible to perform training on ultra-large-scale graphs. However, few existing works systematically study the impact of intra-cluster variance on face clustering, which is a common problem in both collected face datasets and real-world photos. We therefore focus on this key issue, and propose the CLIP-Cluster to tackle this challenge through text-driven face hallucination for narrowing the intra-cluster attribute gap.

Vision-Language Models. Recent years have witnessed the great success of vision-language models [3, 9, 46, 61]. As a representative work, Radford *et al.* [32] develop the Contrastive Language-Image Pre-training (CLIP) model, which is trained on large-scale 400 million image-text pairs and can be employed for representation learning on multi-modal embedding space. CLIP maintains a well-aligned image-language space, and has been proven to have strong zero-shot classification capability which is transferable over 30 datasets. Motivated by the impressive performance of CLIP, many follow-ups [18, 54, 55] have been proposed. Patashnik *et al.* [30] propose StyleCLIP, which combines the generative powers of StyleGAN [15] with the visual concept encoding abilities of CLIP for image synthesis. Wang *et al.* [43] leverage the pre-trained CLIP model to

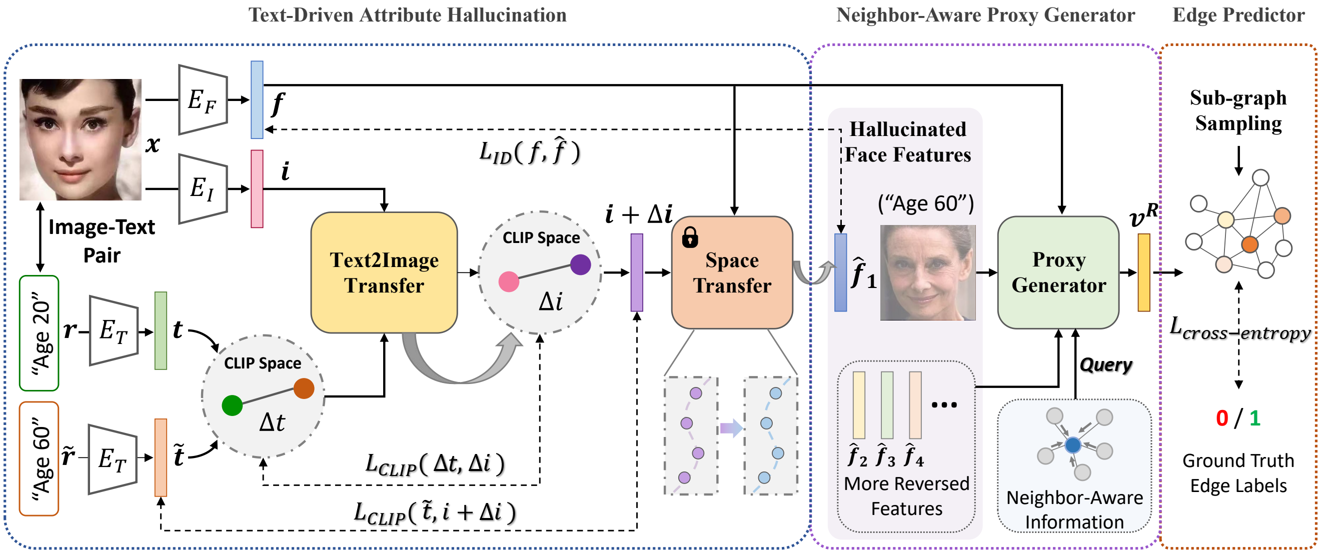

Figure 2. Overview of the proposed CLIP-Cluster to mitigate the effect of intra-cluster facial variations for better face clustering. Leveraging the powerful zero-shot image classification capability of CLIP, we develop a CLIP-guided text-driven attribute hallucination to bridge the attribute gap within clusters. With face features $\{i, f\}$ from CLIP and face recognition space $\mathcal{C}$, and the original and reversed text embedding $\{t, \tilde{t}\}$ as input, the text-driven attribute hallucination obtains transformed feature $i + \Delta i$ corresponding to $\tilde{t}$. It is further transferred to the $\mathcal{C}$ space with a trained space transfer to get hallucinated face features $\hat{f}$. Furthermore, to make reasonable use of this augmented information $\hat{f}$, a neighbor-aware proxy generator is designed for adaptive attribute fusion. Based on these designs, we can perform the subsequent GCN-based edge prediction in a more compact feature space, which significantly reduces the learning difficulty.

drive a NeRF [25] for image manipulation. Wav2CLIP [48] learns robust audio representations by distilling from CLIP embedding space. In this work, we leverage the joint language-image embedding space of the CLIP to constrain the consistency of the learned image features and the language conditions for text-driven face attribute transfer.

Graph Convolutional Network. Graph Convolutional Network (GCN) [7, 35, 42] opens an efficient path for processing non-Euclidean structured data compared with convolution operations. It shows impressive capability on various computer vision tasks including action recognition [6, 56], kinship reasoning [19], and semantic segmentation [33, 58]. Since GCN excels at modeling graph-structured data, it has also been widely adopted in the clustering research community as the basic workflow [45, 47, 51, 52, 53, 59]. Yang *et al.* [52] and Wang *et al.* [47] are an earlier group of researchers that introduce GCN into face clustering task. They reorganized face data into graphic patterns for interactive feature propagation and achieved admirable performance. In this work, we leverage the powerful feature learning ability of GCN to design the neighbor-aware information extractor and edge predictor.

3. Methodology

3.1. Overview

To mitigate the effect of intra-cluster facial variations on face clustering, we develop a CLIP-guided text-driven feature transfer strategy for face attribute hallucination to bridge the attribute gap within clusters, and term the proposed method as CLIP-Cluster. An overview of the proposed CLIP-Cluster is shown in Figure 2. Leveraging the powerful zero-shot image classification capability of CLIP, we achieve face attribute transfer in the feature space conditioned on text prompts. Furthermore, to make reasonable use of this augmented information, a neighbor-aware proxy generator is developed for adaptive attribute fusion. Based on these designs, we can perform the subsequent GCN-based edge prediction in a more compact feature space, which significantly reduces the learning difficulty. In the following, we will detail the text-driven face attribute hallucination in Section 3.2. In Section 3.3, the neighbor-aware proxy generator is introduced for reasonable attribution fusion. Section 3.4 describes the GCN-based edge prediction. And we systematically summarize the training and test process in Section 3.5.

3.2. Text-Driven Attribute Hallucination

In this section, we link the face image to its corresponding language concept, and employ text prompts as conditions to guide the attribute transfer with the help of the powerful vision-language pre-trained model CLIP. We interpret this process as face attribute hallucination. In this way, each face sample is extended into different variants, thus opening the avenue for bridging the attribute gap between intra-cluster neighbors.

Definition of Text Prompts. For a face image x, we take into account three representative attributes, *i.e. age, pose* and *expression*, since these factors are thought to be the main variables responsible for the intra-class differences. It is worth noting that the attribute category is expandable if more fine-grained variants are considered. Each attribute further contains multiple cases as: *Age* = $\{a_0, a_1, \cdots, a_{C-1}\}$, *Pose* = {frontal, profile} and *Expression* = {happy, sad, neutral,$\cdots$}, with attributes collection denoted as R = {*Age*, *Pose*, *Expression*}. We represent the attribute set of x as r, and the corresponding reversed set as $\tilde{r} = R - r$. On this basis, the text prompts templates are defined as "A photo of a male / female at the age of {age}", "A photo of a {expression} male / female", "A photo of a male / female in profile" and "A photo of a frontal male / female" respectively, with the target attributes in r and $\tilde{r}$ filled in.

Text-Driven Attribute Hallucination in CLIP Space. Since these attribute texts are easily accessible and contain rich prior knowledge, they are advantaged for attribute transfer guidance. With the image-attribute pair x, r and the target reversed attribute $\tilde{r}$, the image representation $i = E_I(x)$ and the text embedding $t = E_T(r)$ and $\tilde{t} = E_T(\tilde{r})$ are extracted, where E_I and E_T are the pre-trained image encoder and text encoder in CLIP as shown in Figure 2. To obtain the language embedding, we feed E_T with the constructed text prompt sentences of r and $\tilde{r}$. On this basis, the semantic offset $\Delta t = t - \tilde{t}$ represents the mapping direction of two different attributes in the latent language spaces. Since the image and text space is aligned in CLIP, Δt can be utilized to guide the corresponding manipulation in the image space. We therefore develop a text-to-image transfer module $\mathcal{M}$ to link these two spaces. The output image offset guided by Δt is formulated as,

$$\Delta i = \mathcal{M}_\theta(\Delta t, i). \tag{1}$$

Here we include the original image representation i as another input, since the correspondence relationship between Δt and Δi is not one-to-one mapping. Δt only guides the transfer direction, so i is needed to further provide a starting point for the transformation. This process is supervised with the following loss function,

$$L_{\text{CLIP}} = D_{\text{CLIP}}(\Delta t, \Delta i) + D_{\text{CLIP}}(\tilde{t}, i + \Delta i), \tag{2}$$

where D_{CLIP} is the cosine distance between the text and image embeddings in the CLIP space. We simultaneously constrain the D_{CLIP} of the global feature pair $(\tilde{t}, i+\Delta i)$ and the feature offset pair $(\Delta t, \Delta i)$ as shown in Figure 2 for better supervision. In this way, we achieve hallucination across various attributes with only text prompts as guidance. Benefiting from the zero-shot language-image classification capability of CLIP, we can get rid of the demand for large-scale annotated paired data and realize more free and extensible face attribute transfer.

Link CLIP and Clustering Space. CLIP is pre-trained with text-image pair under contrastive objective, leading to a wider feature space containing rich prior knowledge. Since the face clustering task is more focused on identity information, it is necessary to eliminate the identity-irrelevant message from CLIP embedding for purer face features. In representative face clustering frameworks, a CNN-based model E_F trained under face recognition supervision is usually used to extract face identity representation for clustering, and we denote this embedding as $\mathcal{C}$. Therefore, it is a natural solution to transfer CLIP embedding to $\mathcal{C}$ space to obtain identity-aware features which are more suitable for face clustering. More importantly, this also provides a way to constrain the identity consistency of faces under different attributes. For more accurate feature mapping, this MLP-based space transfer model $\mathcal{B}$ is pre-trained with paired face image embeddings $\{i, f\}$ from CLIP and $\mathcal{C}$ space, where $f = E_F(x)$. Instead of directly learning the feature mapping, we alternatively learn the transfer direction, which can significantly reduce the training difficulty. Specifically, a reference pair $\{i_r, f_r\}$ with the same identity class of $\{i, f\}$ is randomly selected, and the learning process of this feature transfer model $\mathcal{B}$ is,

$$\Delta \hat{f} = \mathcal{B}_\zeta(i - i_r, f_r), \tag{3}$$

with $\Delta\hat{f} + f_r$ as the final estimated feature in $\mathcal{C}$ space. In this way, we learn the manipulation direction mapping between the two spaces. And this training process is supervised by $L_{\text{Transfer}} = 1 - \langle \Delta\hat{f} + f_r, f\rangle$, where $\langle\cdot,\cdot\rangle$ computes the cosine similarity between its arguments. The trained transfer model $\mathcal{B}$ is then integrated into the whole framework as a frozen module for end-to-end training as shown in Figure 2, and its role is to map Δi in Equation 1 to the corresponding $\mathcal{C}$ space as $\Delta f = \mathcal{B}_\zeta(\Delta i, f)$. f is the representation of the original face extracted with $f = E_F(x)$, and here it acts as the reference. $\hat{f} = f + \Delta f$ is the estimated feature with transferred attribute. On this basis, the identity consistency constraint is defined as,

$$L_{\text{ID}} = 1 - \langle \hat{f}, f\rangle. \tag{4}$$

After this step, for each face sample, we can transfer it across various attributes with mere text guidance and obtain the corresponding embeddings $\hat{f} = [\hat{f_1}, \hat{f_2}, \cdots, \hat{f_M}]$ in the $\mathcal{C}$ space, where M is the number of total categories in the reversed attribute set $\tilde{r}$.

3.3. Neighbor-aware Proxy Generator

While a single face is transferred into various attributes as $\hat{f} = [\hat{f_1}, \hat{f_2}, \cdots, \hat{f_M}]$, another key issue is how to make reasonable and full use of this rich information. To this end, we further design a neighbor-aware proxy generator as shown in Figure 3, which is realized via the transformer decoder for meaningful feature fusion. Specifically,

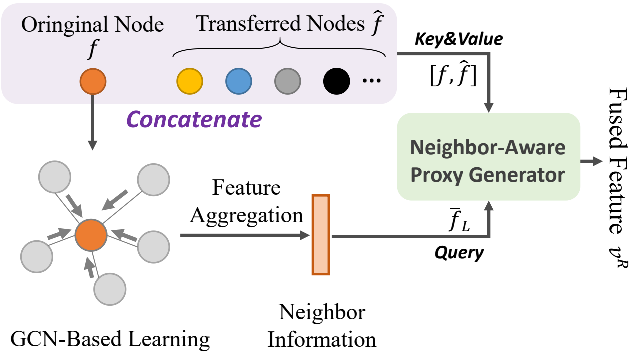

Figure 3. Visualization of the proposed neighbor-aware proxy generator to fuse the features describing various attributes into a proxy feature to build a bridge among different sub-clusters and reduce the intra-class variance. This module is realized via the transformer decoder with neighbor information as the *query* and the transferred attribute features as the *key* and *value*.

the neighbor information is extracted with a $\bar{L}$-layer GCN. $A \in \mathbb{R}^{N\times N}$ denotes the KNN-based adjacency matrix constructed with face features set $F = \{f\} \in \mathbb{R}^{N\times D}$, where N is the number of face images and D is the feature dimension. We zero the diagonal of A to get $\bar{A}$, and on this basis, the neighbor information is learned with,

$$\bar{F}_{l+1} = \sigma(\bar{A}\bar{F}_l\bar{W}_l), \quad l = 1 \ldots L \tag{5}$$

where $\bar{W}_l \in \mathbb{R}^{D\times D}$ is the learnable parameters for this GCN. σ is a nonlinear activation and we use ReLU [26, 50]. $\bar{F}_0 = F$ and $\bar{F}_L$ is the final learned neighbor information.

Since $\bar{f}_L$ aggregates rich messages from the neighborhood, it reflects the attribute characteristics of these neighboring face nodes. We therefore use $\bar{f}_L$ as the *query* and leverage the cross-attention mechanism in the transformer decoder for learning neighbor-aware attention. With $v = [f, \hat{f}] \in \mathbb{R}^{D\times(M+1)}$ as the *key* and *value*, the cross-attention learning process can be formulated as,

$$\begin{aligned} v^{r'} &= \mathrm{MCA}(\mathrm{LN}(\bar{f}_L, v^{r-1})), \\ v^{r} &= \mathrm{MLP}(\mathrm{LN}(v^{r'})), \quad r = 1 \ldots R \end{aligned} \tag{6}$$

where $v^0 = v$ and R is the depth of the decoder block. The transformer decoder architecture is a stack of alternating layers of Multi-head Cross-Attention (MCA) and MLP blocks, with the LayerNorm (LN) before every block. After this step, we can get the fused feature v^R for each face node. Since v^R is obtained under the guidance of neighbor information, it gives higher fusion weights to the attributes close to the neighbor nodes, thus building a bridge among different sub-clusters and reducing the intra-class variance. Based on these proxy representations, the subsequent network can perform face clustering in an easier way, and simultaneously provide feedback guidance for the previous network under the end-to-end training paradigm.

3.4. GCN-Based Edge Predictor

Based on the learned proxy representations $V = \{v^R\}$, we further introduce a GCN-based edge confidence predictor as a proxy task for face clustering. With the KNN-based affinity graph as the adjacency matrix A, which controls the feature propagation direction, this computational process is formulated as,

$$V_{l+1} = \sigma\left([V^T, (\widetilde{A}V_l)^T]^T W_l\right), \quad l = 1 \ldots L \tag{7}$$

where $\widetilde{A} = \widetilde{D}^{-1}(A + I)$ and $\widetilde{D} = \sum_j \widetilde{A}_{ij}$. $W_l \in \mathbb{R}^{D\times D_{\text{out}}}$ is a learnable matrix and σ is the ReLU activation function. V_l denotes the embeddings at l-th layer with $V_0 = V$ is the input face features. Furthermore, for edge prediction, we design an MLP-based binary classifier supervised by the cross-entropy loss L_{cross} between the predicted edge confidence and the ground-truth edge labels. Particularly, paired features corresponding to an edge are fed into the MLP for estimating the two-dimension edge confidence. The ground-truth label of an edge is 1 if the two nodes belong to the same class, otherwise it will be 0.

3.5. Training and Inference

In this section, we systematically summarize the training and testing process based on the above-mentioned designs.

Training For large-scale training, we leverage the structure-preserving sub-graph sampling strategy in [36]. In each training iteration, based on these sampled face nodes, we perform text-driven face hallucination and neighbor-aware proxy generator to get the fused face embeddings V which narrow the attribute gap within the cluster. These features are then utilized for GCN-based edge prediction learning. All modules are organized in an end-to-end training paradigm, and the overall loss function is formulated as,

$$L = L_{\text{CLIP}} + \lambda L_{\text{ID}} + \mu L_{\text{cross}} \tag{8}$$

Inference During inference, we use the whole graph as input for efficiency. The KNN-based affinity graph $\mathcal{G}$ is constructed as the initial cluster. This graph is then pruned to $\mathcal{G}'$ based on the learned edge scores with threshold τ. After this step, most wrong connections are removed. However, there still exists a minority of false positive edges, since the initial affinity graph is densely connected. We therefore introduce Infomap [34] as an off-the-shelf tool for further graph refinement based on $\mathcal{G}'$ and get the final clusters.

4. Experiments

4.1. Experimental Settings

Datasets. We use MS-Celeb-1M (MS1M) [13] and Webface42M [64] for training and evaluation in the face clustering task. The refined MS1M contains about 5.82M face images from 85K identities. We then follow [53] to evenly split the MS1M [13] into 10 partitions, while each part consists of about 0.5M images from 8.6K identities. 1 part is used as labeled data ($Part0$) for training and the other 9 parts ($Part1$, $\cdots$, $Part9$) are regarded as unlabeled data for testing. The WebFace42M is a large face benchmark with about 42M images from 2M identities, which presents a new challenge for face clustering. The MegaFace [16] contains a probe set with 3,530 images and a gallery set with over 1M images. We use it to evaluate face recognition results when training models using pseudo-labeled images obtained through face clustering methods.

Metrics. We evaluate the face clustering performance on the mainstream metrics *Pairwise F-score* (F_P) and *BCubed F-score* (F_B) [2]. These two indicators measure the clustering performance from paired data and cluster-wise data respectively, with the harmonic mean of *Precision* and *Recall*. To further verify the quality of the pseudo-labels obtained from the clustering results, following the practice in [53], we use different proportions of pseudo-label data along with the labeled data in $Part0$ to train the face recognition model and then evaluate the rank-1 face identification accuracy on MegaFace challenge 1 with 1M distractors.

Implementation Details. In this work, we select *age, pose* and *expression* as three representative face attributes for research, which are considered to be the main factors for the intra-class variance. In further practice, attribute selection is extensible if other attributes become the main variance factors in the target dataset. The *age* and *expression* are estimated with FaceLib [1], and the head pose is predicted through Deepgaze [29]. We divide the age into four stages, *i.e.* childhood ($0-10$), youth ($10-30$), middle age ($30-50$), and old age ($50-100$), and use the median of each age group as the label. Therefore we have *Age* = $\{a_0, \cdots, a_3\}$ = $\{5, 20, 40, 75\}$. There are seven *Expressions* including "happy", "sad", and "neutral". *Pose* = {frontal, profile} depends on the head pose in yaw dimension, where yaw$\in [-20, 20]$ is considered as "frontal". We follow these partitions for the accuracy-efficiency trade-off, and finer divisions can also be made if necessary. For GCN training, the affinity graph is built by KNN algorithm [8] with $k = 80$, therefore there are more wrong edges to balance the number of positive and negative edge samples in training. During inference, the affinity graph is built with $k = 50$ to reduce the number of wrong edges for cleaner clustering results. In Equation 8, we experimentally choose suitable scale factors as $\lambda = \mu = 1$. The threshold τ is set as 0.5 for the graph pruning.

4.2. Ablation Study

Design of the Neighbor-Aware Proxy Generator. In this subsection, we explore different designs for face feature fusion and evaluate their face clustering performance with all models trained on the $Part0$ and tested on the $Part1$ of MS1M. After the face attribute hallucination operation, each face sample is augmented to various attributes across age, expression, and pose. Since our core intention is to narrow the attribute gap within the same class so as to compact the intra-cluster variance, it is a key issue to develop a fusion module for information fusion while making the most of these learned features. To this end, we exploit a cross-attention-based neighbor-aware proxy generator that employs neighbor messages as the query to fuse the features describing various attributes into a proxy feature to build a bridge among different sub-clusters and reduce the intra-class variance. Here we compare this design with two other ones, *i.e.* averaging operation, and self-attention-based approach. The performance comparison is shown in Table 1. It can be seen that the naive averaging operation is inferior to the other two learnable transformer-based strategies. The introduction of the attention mechanism helps to perform feature fusion adaptively, leading to significant performance improvement. Furthermore, in the cross-attention-based design, the GCN-based neighbor information is adopted as the query to reflect the attribute distribution of nearby face nodes, thus providing guidance for feature fusion with different attributes. In this way, the fused feature is encouraged to be consistent with the query information, thus achieving the purpose of reducing the attribute gap between adjacent samples. And it boosts the face clustering F_p from 94.04 to 94.22 compared with the self-attention-based one.

Method	Precision	Recall	F_P
Average	96.73	90.97	93.76
Self-Attention	95.67	**92.47**	94.04
Cross-Attention	**97.06**	91.54	**94.22**

Table 1. Comparison of different designs for face feature fusion. All are trained with the $Part0$ and tested on the $Part1$ in MS1M. We observe that the cross-attention-based neighbor-aware proxy generator achieves superior performance than others.

Method	Precision	Recall	F_P
Ours + Naive Pruning	94.26	84.05	88.86
Only Infomap	95.50	**92.51**	93.98
Ours + Infomap	**97.06**	91.54	**94.22**

Table 2. Investigation of the graph refinement designs. We compare the naive pruning and the Infomap-based refinement strategies, and show the results with mere Infomap. All are trained with $Part0$ and tested on $Part1$ in MS1M. It can be seen that the additional refinement operation brings significant performance gain.

Effect of the Additional Graph Refinement. In our method, we use edge prediction as a proxy task for face clustering. This proxy is simple with high efficiency, and edge pruning with the learned edge confidence is able to

Method	Precision	Recall	F_P	Time
K-Means [23]	52.52	70.45	60.18	11.5h
DBSCAN [10]	72.88	42.46	53.50	**110s**
HAC [39]	66.84	70.01	68.39	12.7h
ARO [28]	81.10	7.30	13.34	1650s
CDP [57]	80.19	70.47	75.01	140s
L-GCN [47]	74.38	83.51	78.68	5208s
GCN-D+S [53]	**98.24**	75.93	85.66	3700s
GCN-V+E [51]	92.56	83.74	87.93	690s
DA-Net [12]	95.88	85.87	90.60	329s
STAR-FC [36]	96.20	88.10	91.97	310s
STAR-FC++ [37]	96.74	89.93	93.21	312s
ADA-Nets [45]	-	-	92.79	1100s
MHC [5]	-	-	93.22	-
Ours	97.06	**91.54**	**94.22**	280s

Table 3. Methods comparison on face clustering performance and inference time. All models are trained with $Part0$ and tested with $Part1$ from MS1M. Our proposed method achieves state-of-the-art face clustering results with comparable inference efficiency with most of the recent high-performance methods.

eliminate a majority of negative edges. Whereas, since we perform inference based on the KNN-based densely-connected affinity graph, there may be some remaining false positive edges. We therefore add a further refinement operation for better clustering. In this subsection, we investigate the impact of different refinement operations on the final face clustering performance as shown in Table 2, with $Part0$ as the training data and testing on the $Part1$ of MS1M. In the naive pruning method, face clusters are obtained with dynamic edge pruning [57] based on the predicted edge scores. Compared with the naive pruning strategy, graph refinement based on Infomap [34] can improve the pairwise F-score from 88.86 to 94.22, which verifies the significance of the additional refinement operation to boost the face clustering results. We also show the face clustering results with mere Infomap technology for fairer comparisons. It can be seen that our method has great performance improvement on the basis of Infomap from 93.98 to 94.22, which proves the effectiveness of our proposed method in dealing with face clustering problems.

4.3. Face Clustering on MS1M

For face clustering performance evaluation and method comparison, we conduct experiments on the popular large-scale face dataset MS1M [13]. All results in Table 3 are obtained on the MS1M dataset with $Part0$ as the training set and $Part1$ as the testing set. In addition to the pairwise F-score result, we also show the corresponding precision and recall for more comprehensive evaluations. It can be seen that the proposed method greatly surpasses the existing methods and achieves state-of-art face clustering results. For example, we improve the F_P from 93.22 to 94.22 compared with the recent high-performance work MHC. Moreover, the inference time of our method is comparable with most of the recent high-performance methods. Benefiting from the full graph operation and parallel matrix computing, the proposed CLIP-Cluster can perform inference with 280s, which demonstrates the efficiency of the proposed method. For a fair comparison, this reported inference time does not include the time of KNN graph construction, which takes about 15s with GPU acceleration. In Table 5 we further show the face clustering performance on larger-scale test sets, with 1.74M unlabeled data consisting of $Part0 \sim Part3$, 2.89M consisting of $Part0 \sim Part5$ and so on. Results in Table 5 show that although the inference becomes more challenging with larger-scale test data, the proposed CLIP-Cluster can keep superior face clustering performance, and our method consistently surpasses other clustering baselines under different scales of test sets. Particularly, compared with the recent representative clustering approach ADA-Nets [45], our method boosts the F_P significantly from 83.99 to 87.99 and improves the F_B from 83.28 to 85.85 on 5.21M unlabeled data.

Method	Precision	Recall	F_P	F_B
K-Means [23]	95.99	50.05	65.80	78.29
HAC [39]	98.25	59.76	74.31	85.46
DBSCAN [10]	94.77	44.12	60.21	77.87
ARO [28]	99.34	62.83	76.98	88.83
GCN-D [53]	98.05	52.54	68.42	71.47
STAR-FC [36]	96.77	94.00	95.36	94.93
STAR-FC++ [37]	99.14	97.43	98.27	97.50
Ours	**99.81**	**98.52**	**99.16**	**99.42**

Table 4. Method comparison on face clustering performance on the WebFace42M dataset. All methods are trained with 4M face images and evaluated with the 4M test data from WebFace42M.

4.4. Face Clustering on WebFace42M

In this subsection, we evaluate the face clustering performance on the million-scale face benchmark WebFace42M [64] for more sufficient method comparisons. Following the practice in STAR-FC [36], 4M images from the WebFace42M dataset are used for training and the inference is performed on 4M test data. There is no identities overlap between the training set and the testing set. The face clustering performance is shown in Table 4. It can be seen that under such large-scale training and large-scale testing experiment setting, the proposed CLIP-Cluster once again outperforms other approaches on both F_P and F_B.

4.5. Face Recognition with Pseudo labels

Since an important application of face clustering is data annotation, to verify the effectiveness of the proposed

#unlabeled	1.74M		2.89M		4.05M		5.21M	
Method / Metrics	F_P	F_B	F_P	F_B	F_P	F_B	F_P	F_B
K-Means [23]	73.04	75.20	69.83	72.34	67.90	70.57	66.47	69.42
HAC [39]	54.40	69.53	11.08	68.62	1.40	67.69	0.37	66.96
DBSCAN [10]	63.41	66.53	52.50	66.26	45.24	44.87	44.94	44.74
ARO [28]	8.78	12.42	7.30	10.96	6.86	10.50	6.35	10.01
CDP [57]	70.75	75.82	69.51	74.58	68.62	73.62	68.06	72.92
L-GCN [47]	75.83	81.61	74.29	80.11	73.70	79.33	72.99	78.60
GCN-D [53]	83.76	83.99	81.62	82.00	80.33	80.72	79.21	79.71
GCN-V+E [51]	84.04	82.84	82.10	81.24	80.45	80.09	79.30	79.25
Clusformer [27]	84.60	84.05	82.79	82.30	81.03	80.51	79.91	79.95
STAR-FC [36]	88.28	86.26	86.17	84.13	84.70	82.63	83.46	81.47
ADA-Nets [45]	89.33	87.98	87.50	86.03	85.40	84.48	83.99	83.28
STAR-FC++ [37]	90.00	88.15	88.32	86.12	87.04	84.83	85.89	84.39
Ours	**91.44**	**89.44**	**89.95**	**87.75**	**88.93**	**86.78**	**87.99**	**85.85**

Table 5. Method comparison on face clustering when training with 0.5M face images ($Part0$) and testing with different numbers of unlabeled face images (the 1.74M unlabeled data consists of $Part1 \sim Part3$ and so on) from MS1M. Although the inference becomes more challenging with larger-scale test data, the proposed CLIP-Cluster can keep superior face clustering performance, and our method consistently surpasses other clustering baselines under different scales of test sets.

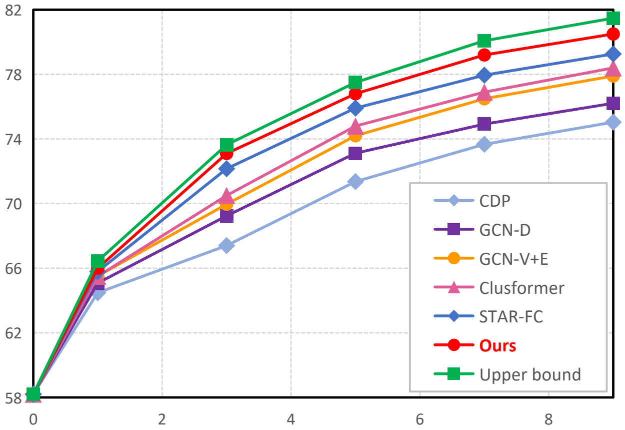

Figure 4. Rank-1 face identification accuracy on MegaFace with 1M distractors, with the horizontal axis indicating the ratio of unlabeled to labeled data. The point where the ratio is 0 indicates that only a split of labeled data is used for training.

method from this dimension, we further test the performance of the cluster labels on the face recognition task. Specifically, pseudo-labels are assigned to unlabeled data based on the face clustering results. Then we utilize these pseudo-labeled data to train face recognition models and investigate the performance gain brought by these extra annotated training data. Following the setting in [51, 53], we use labeled data from $Part0$ and the remaining unlabeled data with pseudo-label to train the face recognition models and then evaluate the trained models on MegaFace [16]. Figure 4 plots the rank-1 face identification accuracy on MegaFace with 1M distractors, where the horizontal axis represents the ratio of the amount of pseudo-labeled and the labeled data used for training. The upper bound is trained using data with ground-truth labels. From this curve graph, it can be seen that with the increase of pseudo-label training data, the performance of face recognition has been continuously improved. And the proposed CLIP-Cluster achieves superior recognition results owing to the more accurate cluster labels compared with other face clustering baselines. Our method boosts the face recognition performance on MegaFace from 58.2% to 80.5% eventually depending on the extra 5.21M pseudo-labeled data.

5. Conclusion

In this work, we have proposed an attribute hallucination framework named CLIP-Cluster to narrow the intra-class variance caused by different face attributes for better face clustering. The CLIP-Cluster first hallucinates multiple representations for different attributes based on the well-aligned image-language CLIP space. Furthermore, a cross-attention-based neighbor-aware proxy generator is developed to fuse the features describing various attributes into a proxy feature to build a bridge among different sub-clusters and reduce the intra-class variance. Extensive experiments show that our proposed approach outperforms state-of-the-art face clustering methods with high inference efficiency.

Acknowledgment. This work was supported in part by the National Key Research and Development Program of China under Grant 2022ZD0160102 and in part by the National Natural Science Foundation of China under Grant 62125603.

References

[1] Facelib. https://github.com/sajjjadayobi/FaceLib.

[2] Enrique Amigó, Julio Gonzalo, Javier Artiles, and Felisa Verdejo. A comparison of extrinsic clustering evaluation metrics based on formal constraints. *IR*, 2009.

[3] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *ICCV*, pages 2425–2433, 2015.

[4] Qiong Cao, Li Shen, Weidi Xie, Omkar M Parkhi, and Andrew Zisserman. VGGFace2: A dataset for recognising faces across pose and age. In *FG*, 2018.

[5] Yingjie Chen, Huasong Zhong, Chong Chen, Chen Shen, Jianqiang Huang, Tao Wang, Yun Liang, and Qianru Sun. On mitigating hard clusters for face clustering. In *ECCV*, 2022.

[6] Ke Cheng, Yifan Zhang, Congqi Cao, Lei Shi, Jian Cheng, and Hanqing Lu. Decoupling gcn with dropgraph module for skeleton-based action recognition. In *ECCV*, 2020.

[7] Wei-Lin Chiang, Xuanqing Liu, Si Si, Yang Li, Samy Bengio, and Cho-Jui Hsieh. Cluster-GCN: An efficient algorithm for training deep and large graph convolutional networks. In *KDDM*, 2019.

[8] Thomas Cover and Peter Hart. Nearest neighbor pattern classification. *TIT*, 1967.

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, 2019.

[10] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *SIGKDD*, 1996.

[11] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 2020.

[12] Senhui Guo, Jing Xu, Dapeng Chen, Chao Zhang, Xiaogang Wang, and Rui Zhao. Density-aware feature embedding for face clustering. In *CVPR*, 2020.

[13] Yandong Guo, Lei Zhang, Yuxiao Hu, Xiaodong He, and Jianfeng Gao. MS-Celeb-1M: A dataset and benchmark for large-scale face recognition. In *ECCV*, 2016.

[14] Zhizhong Huang, Junping Zhang, and Hongming Shan. When age-invariant face recognition meets face age synthesis: A multi-task learning framework. In *CVPR*, 2021.

[15] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, 2019.

[16] Ira Kemelmacher-Shlizerman, Steven M Seitz, Daniel Miller, and Evan Brossard. The MegaFace Benchmark: 1 million faces for recognition at scale. In *CVPR*, 2016.

[17] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet classification with deep convolutional neural networks. In *NeruIPS*, 2012.

[18] Wanhua Li, Xiaoke Huang, Zheng Zhu, Yansong Tang, Xiu Li, Jie Zhou, and Jiwen Lu. Ordinalclip: Learning rank prompts for language-guided ordinal regression. In *NeurIPS*, pages 35313–35325, 2022.

[19] Wanhua Li, Jiwen Lu, Abudukelimu Wuerkaixi, Jianjiang Feng, and Jie Zhou. Reasoning graph networks for kinship verification: from star-shaped to hierarchical. *TIP*, 30:4947–4961, 2021.

[20] Wei-An Lin, Jun-Cheng Chen, Carlos D Castillo, and Rama Chellappa. Deep density clustering of unconstrained faces. In *CVPR*, 2018.

[21] Junfu Liu, Di Qiu, Pengfei Yan, and Xiaolin Wei. Learn to cluster faces via pairwise classification. In *ICCV*, 2021.

[22] Weiyang Liu, Yandong Wen, Zhiding Yu, Ming Li, Bhiksha Raj, and Le Song. Sphereface: Deep hypersphere embedding for face recognition. In *CVPR*, 2017.

[23] Stuart Lloyd. Least squares quantization in pcm. *TIP*, 1982.

[24] Brianna Maze, Jocelyn Adams, James A Duncan, Nathan Kalka, Tim Miller, Charles Otto, Anil K Jain, W Tyler Niggel, Janet Anderson, and Jordan Cheney. IARPA Janus Benchmark C: Face dataset and protocol. In *ICB*, 2018.

[25] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020.

[26] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *ICML*, 2010.

[27] Xuan-Bac Nguyen, Duc Toan Bui, Chi Nhan Duong, Tien D Bui, and Khoa Luu. Clusformer: A transformer based clustering approach to unsupervised large-scale face and visual landmark recognition. In *CVPR*, 2021.

[28] Charles Otto, Dayong Wang, and Anil K Jain. Clustering millions of faces by identity. *TPAMI*, 2017.

[29] Massimiliano Patacchiola and Angelo Cangelosi. Head pose estimation in the wild using convolutional neural networks and adaptive gradient methods. *PR*, 2017.

[30] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. Styleclip: Text-driven manipulation of stylegan imagery. In *ICCV*, 2021.

[31] Haibo Qiu, Baosheng Yu, Dihong Gong, Zhifeng Li, Wei Liu, and Dacheng Tao. Synface: Face recognition with synthetic data. In *ICCV*, 2021.

[32] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

[33] Ryan Razani, Ran Cheng, Enxu Li, Ehsan Taghavi, Yuan Ren, and Liu Bingbing. Gp-s3net: Graph-based panoptic sparse semantic segmentation network. In *ICCV*, 2021.

[34] Martin Rosvall, Daniel Axelsson, and Carl T Bergstrom. The map equation. *EPJST*, 2009.

[35] Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne Van Den Berg, Ivan Titov, and Max Welling. Modeling relational data with graph convolutional networks. In *ESWC*, 2018.

[36] Shuai Shen, Wanhua Li, Zheng Zhu, Guan Huang, Dalong Du, Jiwen Lu, and Jie Zhou. Structure-aware face clustering on a large-scale graph with 107 nodes. In *CVPR*, 2021.

[37] Shuai Shen, Wanhua Li, Zheng Zhu, Jie Zhou, and Jiwen Lu. Star-fc: Structure-aware face clustering on ultra-large-scale graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[38] Yichun Shi, Charles Otto, and Anil K Jain. Face clustering: representation and pairwise constraints. *TIFS*, 2018.

[39] Robin Sibson. Slink: an optimally efficient algorithm for the

single-link cluster method. *The Computer Journal*, 1973.

[40] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv*, 2014.

[41] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *CVPR*, 2015.

[42] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. In *ICLR*, 2018.

[43] Can Wang, Menglei Chai, Mingming He, Dongdong Chen, and Jing Liao. Clip-nerf: Text-and-image driven manipulation of neural radiance fields. In *CVPR*, 2022.

[44] Hao Wang, Yitong Wang, Zheng Zhou, Xing Ji, Dihong Gong, Jingchao Zhou, Zhifeng Li, and Wei Liu. Cosface: Large margin cosine loss for deep face recognition. In *CVPR*, 2018.

[45] Yaohua Wang, Yaobin Zhang, Fangyi Zhang, Senzhang Wang, Ming Lin, YuQi Zhang, and Xiuyu Sun. Ada-nets: Face clustering via adaptive neighbour discovery in the structure space. *ICLR*, 2022.

[46] Zihao Wang, Xihui Liu, Hongsheng Li, Lu Sheng, Junjie Yan, Xiaogang Wang, and Jing Shao. CAMP: cross-modal adaptive message passing for text-image retrieval. In *ICCV*, 2019.

[47] Zhongdao Wang, Liang Zheng, Yali Li, and Shengjin Wang. Linkage based face clustering via graph convolution network. In *CVPR*, 2019.

[48] Ho-Hsiang Wu, Prem Seetharaman, Kundan Kumar, and Juan Pablo Bello. Wav2clip: Learning robust audio representations from clip. In *ICASSP*, 2022.

[49] Yifan Xing, Tong He, Tianjun Xiao, Yongxin Wang, Yuanjun Xiong, Wei Xia, David Wipf, Zheng Zhang, and Stefano Soatto. Learning hierarchical graph neural networks for image clustering. In *ICCV*, 2021.

[50] Bing Xu, Naiyan Wang, Tianqi Chen, and Mu Li. Empirical evaluation of rectified activations in convolutional network. *arXiv*, 2015.

[51] Lei Yang, Dapeng Chen, Xiaohang Zhan, Rui Zhao, Chen Change Loy, and Dahua Lin. Learning to cluster faces via confidence and connectivity estimation. In *CVPR*, 2020.

[52] Lei Yang, Qingqiu Huang, Huaiyi Huang, Linning Xu, and Dahua Lin. Learn to propagate reliably on noisy affinity graphs. In *ECCV*, 2020.

[53] Lei Yang, Xiaohang Zhan, Dapeng Chen, Junjie Yan, Chen Change Loy, and Dahua Lin. Learning to cluster faces on an affinity graph. In *CVPR*, 2019.

[54] Zhao Yang, Yansong Tang, Luca Bertinetto, Hengshuang Zhao, and Philip H.S. Torr. Hierarchical interaction network for video object segmentation from referring expressions. In *BMVC*, 2021.

[55] Zhao Yang, Jiaqi Wang, Yansong Tang, Kai Chen, Hengshuang Zhao, and Philip H.S. Torr. Lavt: Language-aware vision transformer for referring image segmentation. In *CVPR*, 2022.

[56] Fanfan Ye, Shiliang Pu, Qiaoyong Zhong, Chao Li, Di Xie, and Huiming Tang. Dynamic gcn: Context-enriched topology learning for skeleton-based action recognition. In *ACM MM*, 2020.

[57] Xiaohang Zhan, Ziwei Liu, Junjie Yan, Dahua Lin, and Chen Change Loy. Consensus-driven propagation in massive unlabeled data for face recognition. In *ECCV*, 2018.

[58] Bingfeng Zhang, Jimin Xiao, Jianbo Jiao, Yunchao Wei, and Yao Zhao. Affinity attention graph neural network for weakly supervised semantic segmentation. *TPAMI*, 2021.

[59] Yaobin Zhang, Weihong Deng, Mei Wang, Jiani Hu, Xian Li, Dongyue Zhao, and Dongchao Wen. Global-Local GCN: Large-scale label noise cleansing for face recognition. In *CVPR*, 2020.

[60] Ming Zhao, Yong Wei Teo, Siliang Liu, Tat-Seng Chua, and Ramesh Jain. Automatic person annotation of family photo album. In *ICIVR*, 2006.

[61] Zexuan Zhong, Dan Friedman, and Danqi Chen. Factual probing is [MASK]: learning vs. learning to recall. In Kristina Toutanova, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tür, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao Zhou, editors, *NAACL*, 2021.

[62] Chunhui Zhu, Fang Wen, and Jian Sun. A rank-order distance based clustering algorithm for face tagging. In *CVPR*, 2011.

[63] Wenbin Zhu, Chien-Yi Wang, Kuan-Lun Tseng, Shang-Hong Lai, and Baoyuan Wang. Local-adaptive face recognition via graph-based meta-clustering and regularized adaptation. In *CVPR*, 2022.

[64] Zheng Zhu, Guan Huang, Jiankang Deng, Yun Ye, Junjie Huang, Xinze Chen, Jiagang Zhu, Tian Yang, Jiwen Lu, Dalong Du, and Zhou Jie. Webface260M: A benchmark unveiling the power of million-scale deep face recognition. In *CVPR*, 2021.