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

Shuai Shen\textsuperscript{1,2}, Wanhua Li\textsuperscript{1,2}, Xiaobing Wang\textsuperscript{3}, Dafeng Zhang\textsuperscript{3}, Zhezhu Jin\textsuperscript{3}, Jie Zhou\textsuperscript{1,2}, Jiwen Lu\textsuperscript{1,2,*}

\textsuperscript{1}Department of Automation, Tsinghua University, China
\textsuperscript{2}Beijing National Research Center for Information Science and Technology, China
\textsuperscript{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\textsuperscript{5,21,45,47,49,52,63}, due to the progress in deep learning frameworks\textsuperscript{17,40,41} and more available large-scale training data\textsuperscript{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\textsuperscript{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.

Face clustering has received thorough attention in the computer vision research area. More recently, the supervised face clustering methods are extensively studied\textsuperscript{5,27,56,47,51,57} and have achieved significant performance gain. While they put more attention on how to estimate vertex confidence\textsuperscript{12,51}, construct neighbor link\textsuperscript{45,47}, or mine graph structure\textsuperscript{56,53}, 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-
dy 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 by learning the neighbor-adaptive attention.

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 $F_\beta$ 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 $G$CN [47] and Transformer [12] as the basic technology, and has achieved remarkable performance. These existing methods cope with the face clustering problem from different perspectives like vertex confidence estimation [12], neighbor linkage prediction [45, 57], 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 multimodal 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. [50] 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
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.

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.

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 \( R = \{\text{Age}, \text{Pose}, \text{Expression}\} \). We represent the attribute set of \( x \) as \( r \), and the corresponding reversed set as \( \hat{r} = R - r \). 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 \( \hat{r} \) 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 \( \hat{r} \), the image representation \( i = E_I(x) \) and the text embedding \( t = E_T(r) \) and \( \hat{t} = E_T(\hat{r}) \) are extracted, where \( E_I \) and \( E_T \) are the pre-trained image encoder and text encoder in CLIP as shown in Figure 3. To obtain the language embedding, we feed \( E_T \) with the constructed text prompt sentences of \( r \) and \( \hat{r} \). On this basis, the semantic offset \( \Delta t = t - \hat{t} \) represents the mapping direction of two different attributes in the latent language spaces. Since the image and text space is aligned in CLIP, \( \Delta t \) 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 \( \Delta t \) is formulated as,

\[
\Delta i = \mathcal{M}_\theta(\Delta t, i).
\]

Here we include the original image representation \( i \) as another input, since the correspondence relationship between \( \Delta t \) and \( \Delta i \) is not one-to-one mapping. \( \Delta t \) 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,

\[
L_{\text{CLIP}} = D_{\text{CLIP}}(\Delta t, \Delta i) + D_{\text{CLIP}}(\hat{t}, i + \Delta i),
\]

where \( D_{\text{CLIP}} \) is the cosine distance between the text and image embeddings in the CLIP space. We simultaneously constrain the \( D_{\text{CLIP}} \) of the global feature pair \( (t, i + \Delta i) \) and the feature offset pair \( (\Delta t, \Delta i) \) as shown in Figure 3 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 \( C \). Therefore, it is a natural solution to transfer CLIP embedding to \( C \) 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 \( B \) is pre-trained with paired face image embeddings \( \{i, f\} \) from CLIP and \( C \) 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 \( B \) is,

\[
\Delta \hat{f} = B_C(i - i_r, f_r),
\]

with \( \Delta \hat{f} + f_r \) as the final estimated feature in \( C \) 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 \( B \) is then integrated into the whole framework as a frozen module for end-to-end training as shown in Figure 3 and its role is to map \( \Delta i \) in Equation 1 to the corresponding \( C \) space as \( \Delta f = B_C(\Delta i, f) \). \( f \) is the representation of the original face extracted with \( f = E_F(x) \), and here it acts as the reference. \( \hat{f} = f + \Delta \hat{f} \) is the estimated feature with transferred attribute. On this basis, the identity consistency constraint is defined as,

\[
L_{\text{ID}} = 1 - \langle \hat{f}, f \rangle.
\]

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 \( C \) space, where \( M \) is the number of total categories in the reversed attribute set \( \hat{r} \).

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,
the neighbor information is extracted with a $\tilde{L}$-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 $\tilde{A}$, and on this basis, the neighbor information is learned with,

$$\tilde{F}_{l+1} = \sigma(\tilde{A}\tilde{F}_{l}W_{l}), \quad l = 1 \ldots L$$

where $W_{l} \in \mathbb{R}^{D \times D}$ is the learnable parameters for this GCN. $\sigma$ is a nonlinear activation and we use ReLU $\sigma$.

Since $f_{L}$ aggregates rich messages from the neighborhood, it reflects the attribute characteristics of these neighboring face nodes. We therefore use $f_{L}$ 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,

$$v^{r'} = \text{MCA} (\text{LN}(f_{L}, v^{r-1})), \quad v^{r} = \text{MLP} (\text{LN}(v^{r'})), \quad r = 1 \ldots R$$

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 $v^{0}$ 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,

$$V_{l+1} = \sigma \left( [V^{T}, (\tilde{A}V_{l})^{T}]W_{l} \right), \quad l = 1 \ldots L$$

where $\tilde{A} = D^{-1} (A + I)$ and $D = \sum_{i} A_{ii}$. $W_{l} \in \mathbb{R}^{D \times D_{out}}$ is a learnable matrix and $\sigma$ 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 [50]. 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 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,

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

**Inference** During inference, we use the whole graph as input for efficiency. The KNN-based affinity graph $G$ is constructed as the initial cluster. This graph is then pruned to $G'$ based on the learned edge scores with threshold $\tau$. 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 $G'$ 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, · · · , 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 (FP) and BCubed
F-score (FB) [8]. These two indicators measure the clustering
performance from paired data and cluster-wise data re-
spectively, 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 expres-
sion 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 = {a0, · · · , a3} = {5, 20, 40, 75}. There are seven Ex-
pressions including “happy”, “sad”, and “neutral”. Pose =
{frontal, profile} depends on the head pose in yaw dimen-
sion, where yaw ∈ [−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 bal-
ance 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 λ = µ = 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

<table>
<thead>
<tr>
<th>Method</th>
<th>Precision</th>
<th>Recall</th>
<th>FP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average</td>
<td>96.73</td>
<td>90.97</td>
<td>93.76</td>
</tr>
<tr>
<td>Self-Attention</td>
<td>95.67</td>
<td>92.47</td>
<td>94.04</td>
</tr>
<tr>
<td>Cross-Attention</td>
<td>97.06</td>
<td>91.54</td>
<td>94.22</td>
</tr>
</tbody>
</table>

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.

<table>
<thead>
<tr>
<th>Method</th>
<th>Precision</th>
<th>Recall</th>
<th>FP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ours + Naive Pruning</td>
<td>94.26</td>
<td>84.05</td>
<td>88.86</td>
</tr>
<tr>
<td>Only Infomap</td>
<td>95.50</td>
<td>92.51</td>
<td>93.98</td>
</tr>
<tr>
<td>Ours + Infomap</td>
<td>97.06</td>
<td>91.54</td>
<td>94.22</td>
</tr>
</tbody>
</table>

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

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 com-
 pact 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 ap-
proach. 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 de-
sign, 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 dif-
ferent attributes. In this way, the fused feature is encour-
ged to be consistent with the query information, thus achiev-
ing the purpose of reducing the attribute gap between adjacent
samples. And it boosts the face clustering FP from 94.04 to
94.22 compared with the self-attention-based one.

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
dge pruning with the learned edge confidence is able to
eliminate a majority of negative edges. Whereas, since we perform inference based on the KNN-based density-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 ∼ Part3, 2.89M consisting of Part0 ∼ 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.

#### 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
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 ∼ 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.

<table>
<thead>
<tr>
<th>Method / Metrics</th>
<th>1.74M</th>
<th>2.89M</th>
<th>4.05M</th>
<th>5.21M</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$F_P$</td>
<td>$F_B$</td>
<td>$F_P$</td>
<td>$F_B$</td>
</tr>
<tr>
<td>K-Means [23]</td>
<td>73.04</td>
<td>75.20</td>
<td>69.83</td>
<td>72.34</td>
</tr>
<tr>
<td>HAC [39]</td>
<td>54.40</td>
<td>69.53</td>
<td>11.08</td>
<td>68.62</td>
</tr>
<tr>
<td>DBSCAN [10]</td>
<td>63.41</td>
<td>66.53</td>
<td>52.50</td>
<td>66.26</td>
</tr>
<tr>
<td>ARO [28]</td>
<td>8.78</td>
<td>12.42</td>
<td>7.30</td>
<td>10.96</td>
</tr>
<tr>
<td>CDP [57]</td>
<td>70.75</td>
<td>75.82</td>
<td>69.51</td>
<td>74.58</td>
</tr>
<tr>
<td>L-GCN [47]</td>
<td>75.83</td>
<td>81.61</td>
<td>74.29</td>
<td>80.11</td>
</tr>
<tr>
<td>GCN-D [53]</td>
<td>83.76</td>
<td>83.99</td>
<td>81.62</td>
<td>82.00</td>
</tr>
<tr>
<td>GCN-V+E [51]</td>
<td>84.04</td>
<td>82.84</td>
<td>82.10</td>
<td>81.24</td>
</tr>
<tr>
<td>Clusformer [27]</td>
<td>84.60</td>
<td>84.05</td>
<td>82.79</td>
<td>82.30</td>
</tr>
<tr>
<td>STAR-FC [56]</td>
<td>88.28</td>
<td>86.26</td>
<td>86.17</td>
<td>84.13</td>
</tr>
<tr>
<td>ADA-Nets [45]</td>
<td>89.33</td>
<td>87.98</td>
<td>87.50</td>
<td>86.03</td>
</tr>
<tr>
<td>STAR-FC++ [37]</td>
<td>90.00</td>
<td>88.15</td>
<td>88.32</td>
<td>86.12</td>
</tr>
<tr>
<td>Ours</td>
<td>91.44</td>
<td>89.44</td>
<td>89.95</td>
<td>87.75</td>
</tr>
</tbody>
</table>

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

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


