GroupFace

Learning Latent Groups and Constructing Group-based Representations for Face Recognition

신승윤

Abstract

대규모 얼굴인식을 위한 Group-aware representation

self-distributed labels

+ 추가적인 annotation 없이 group 간의 sample 수를 uniform 하게 분포하게 함.

group-aware representation

+ search space를 줄이고 performance 에 향상을 가져옴

Introductions

DeepFace

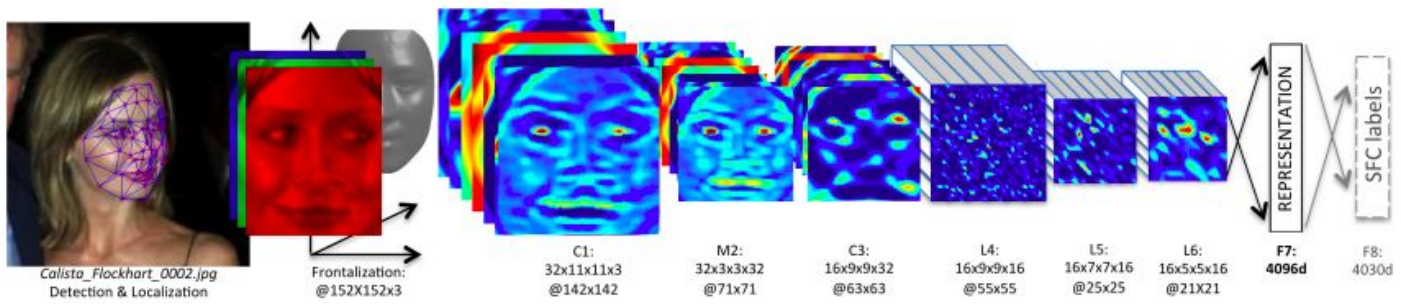
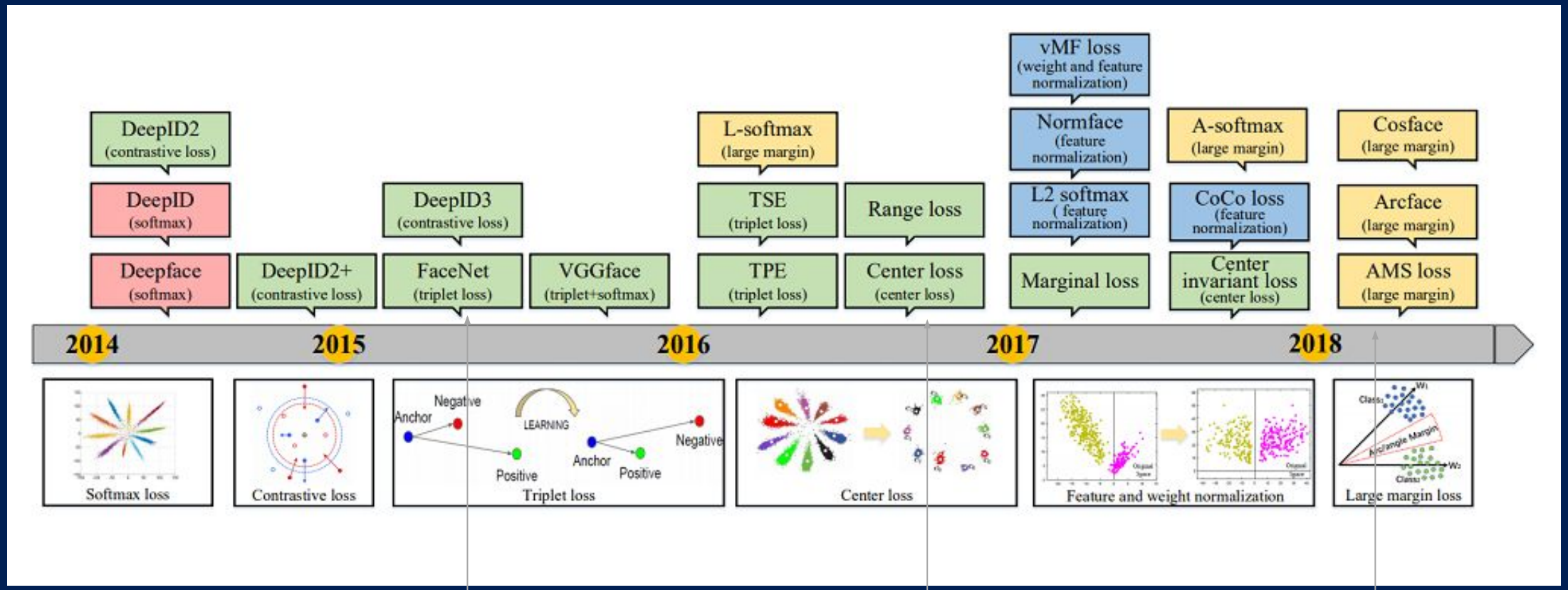


Figure 2. **Outline of the *DeepFace* architecture**. A front-end of a single convolution-pooling-convolution filtering on the rectified input, followed by three locally-connected layers and two fully-connected layers. Colors illustrate feature maps produced at each layer. The net includes more than 120 million parameters, where more than 95% come from the local and fully connected layers.

얼굴인식 ➡️ CNN을 이용한 Multi-classification 문제!

Introductions

Loss function



3개선택,
폭발적인증가

Seperable 보다는
Discriminative

hypersphere

Introductions

Grouping

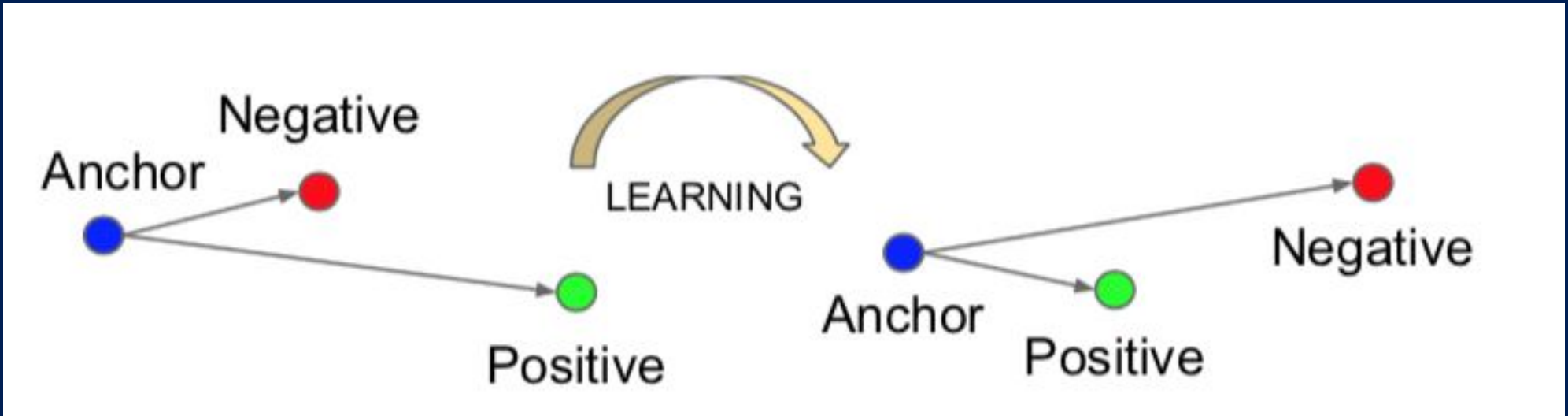
우리가 어떤 사람을 묘사할 때 그사람의 특징을 기억하고 말함.
이처럼 Grouping은 탐색공간을 줄이고 효과적인 embedding 을 도와줌

⚠️ 그렇지만 identities 에 대한 정보 이외에 그 사람의 특징에 대한 라벨링 또한 필요함.

⚠️ 인간의 라벨링은 한정적임 (다양한 특징을 검출하지 못함)

Related Work

Triplet loss



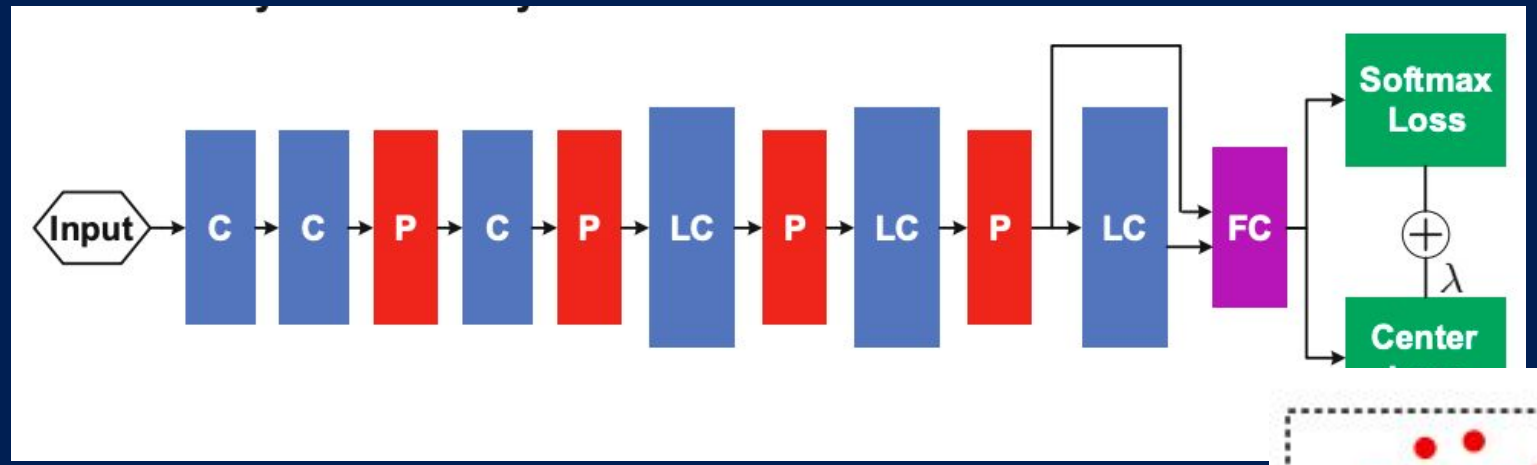
2개의 positive 와 1 개의 negative 로 구성

😓 Train pair 을 만드는 문제가 생각보다 어려움

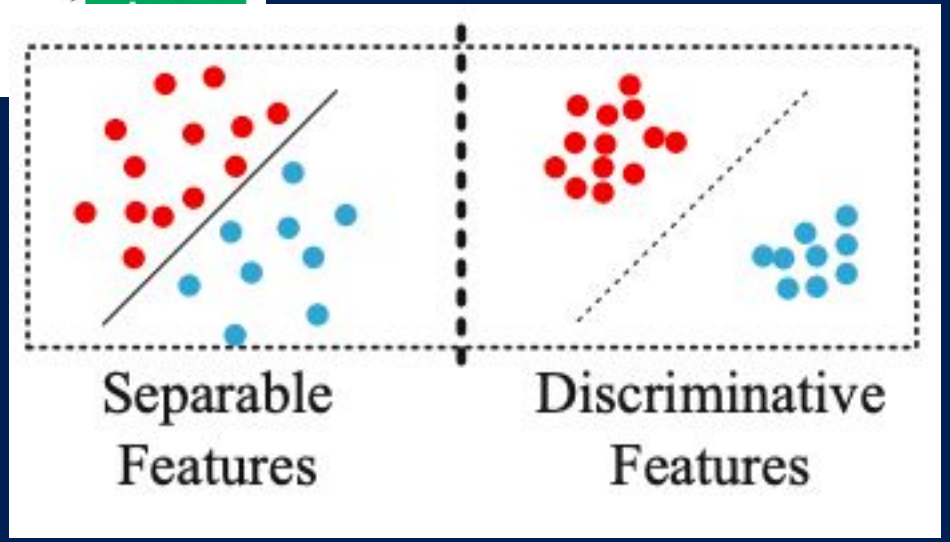
😓 pair 가 매우 많아져 훈련이 어려워짐.

Related Work

Center loss



Joint supervision 을 통해 discriminative power를 높임



Related Work

Losses

Marginal Loss	Margin 도입.
Range Loss	Data imbalance 문제를 해결하는 Range Loss 제안
Ring Loss	Feature maginitude 를 고정
NormFace	Feature normalization 으로 성능향상
SphereFace **CosFace**	Additive , angular margin 제안

Related Work

ArcFac

Feature와 center 간의 각도계산

각도가작아져야 softmax 확률또한 커짐

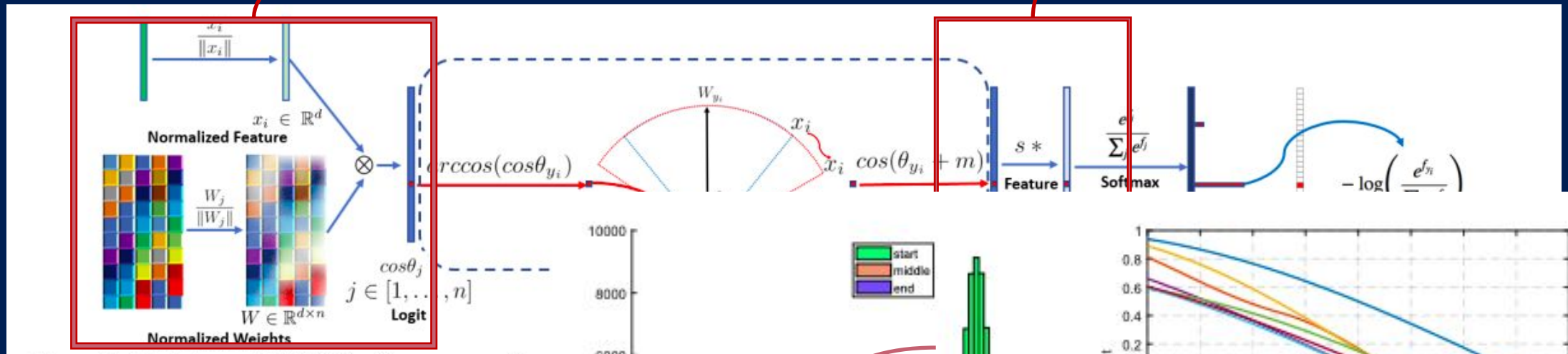
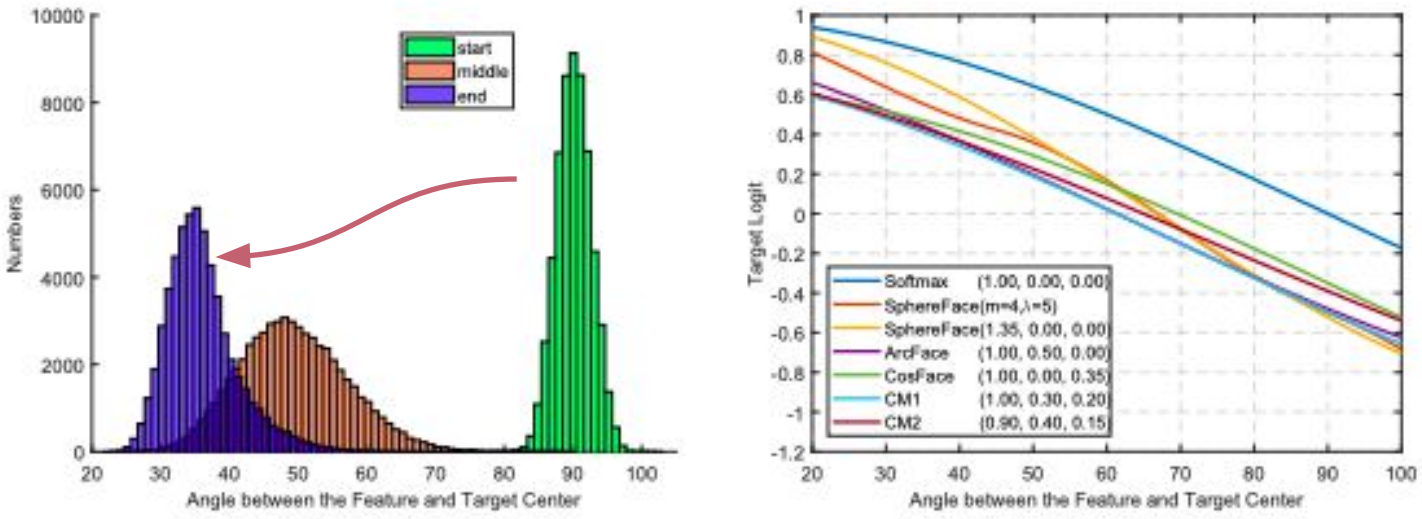


Figure 2. Training a DCNN for face recognition
get the $\cos\theta_j$ (logit) for each class as $W_j^T x_i$.
weight W_{y_i}. In fact, W_j provides a kind of cen
angle θ_{y_i}. After that, we calculate $\cos(\theta_{y_i} +$
function and contribute to the cross entropy loss



(a) θ_j Distributions (b) Target Logits Curves

Related Work

Focused on loss function

이러한 접근 방법
+ Loss 함수를 조금씩 바꾸고
+ FC 레이어나 일부 모델 구조를 변경

❗ 괄목할 만한 성능 향상을 가져오지는 못한다.

Related Work

Grouping

K-mean 과 같은 클러스터링 방법론
➡️ 딥러닝 기반 방법론
➡️ minibatch 에는 적용 불가능

💡 self – distributed
💡 expectation normalized (uniform)

Proposed Method

Overview

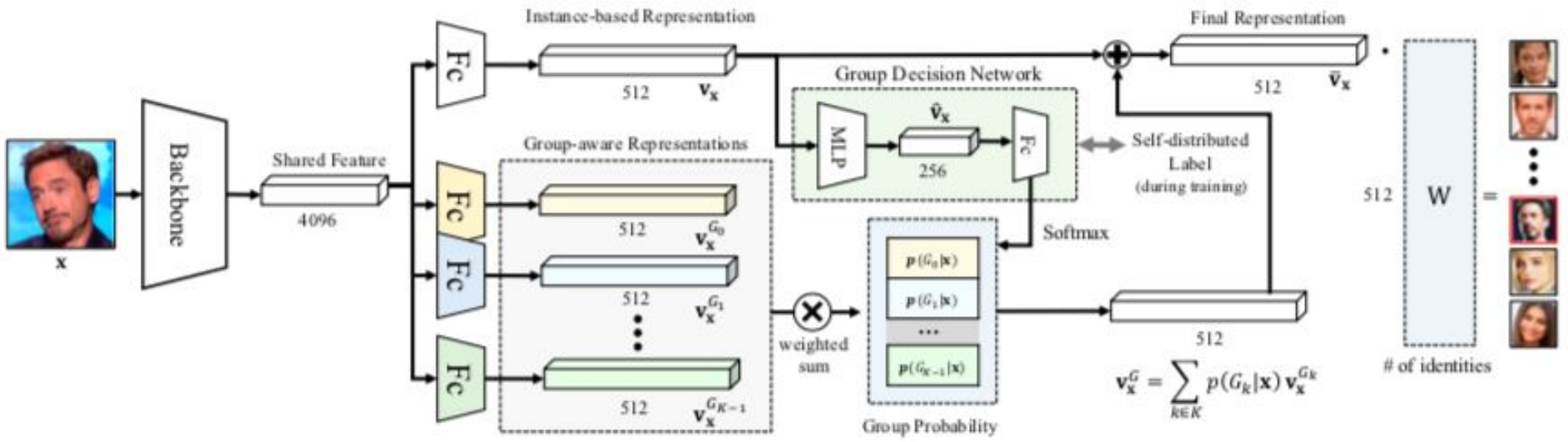
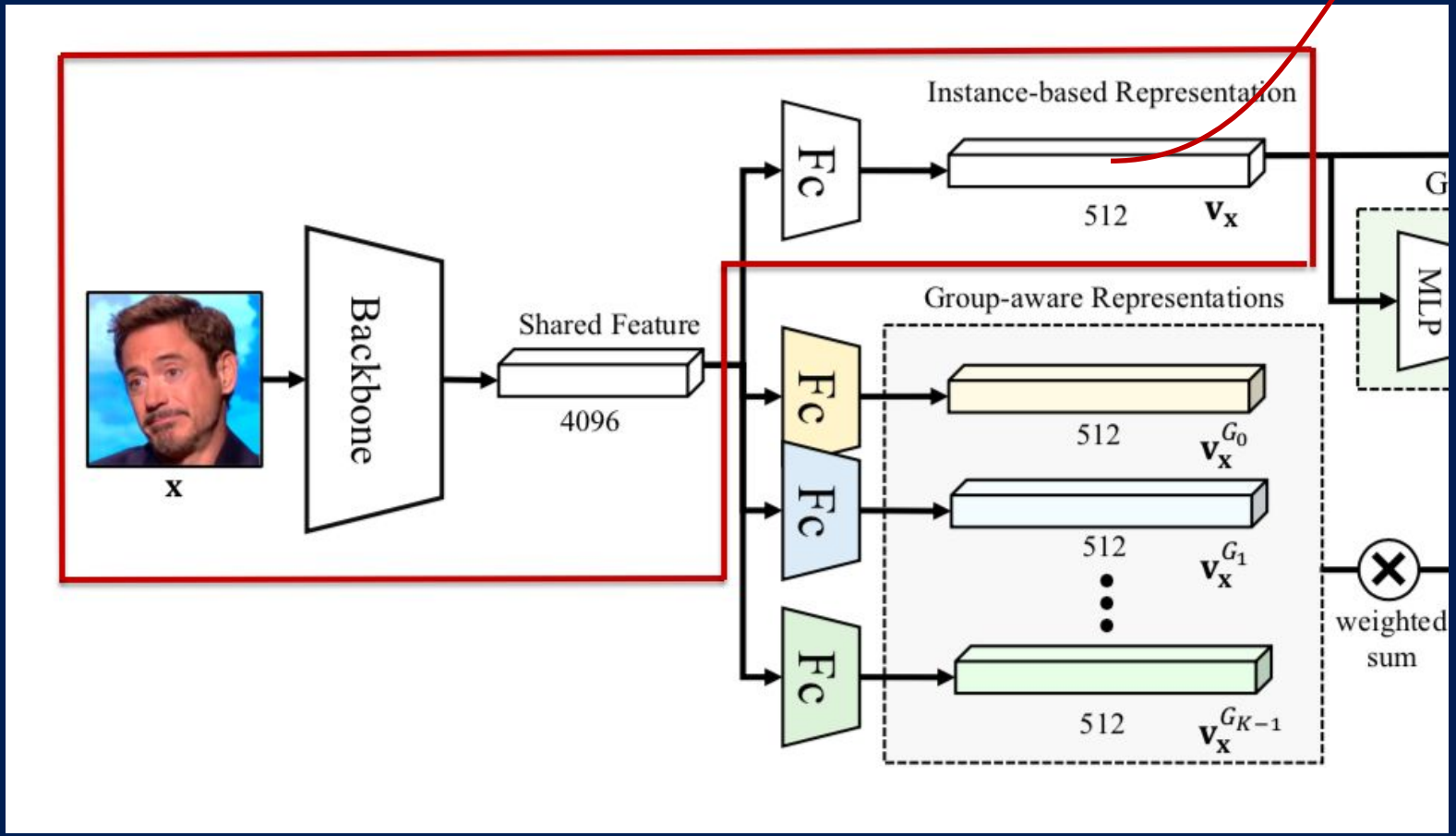


Figure 2. GroupFace generates a shared feature of 4096 dimension and deploys a fully-connected layer for an instance-based representation $\mathbf{v}_{\mathbf{x}}$ and K fully-connected layers for group-aware representations $\mathbf{v}_{\mathbf{x}}^{G}$ for a given sample $\mathbf{x}$. A group-decision-network, which is supervised by the self-distributed labeling, outputs a set of group probabilities $\{p(G_0|\mathbf{x}), p(G_1|\mathbf{x}), ..., p(G_{K-1}|\mathbf{x})\}$ from the instance-based representation. The final representation of 512 dimension is an aggregation of the instance-based representation and the weighted sum $\mathbf{v}_{\mathbf{x}}^{G}$ of the group-aware representations with the group probabilities. W is a weight of the function g.

Proposed Method

Instance based representation

$$p(\mathbf{y}_i|\mathbf{x}) = \mathrm{softmax}_k(g(\mathbf{v}_\mathbf{x})),$$



Proposed Method

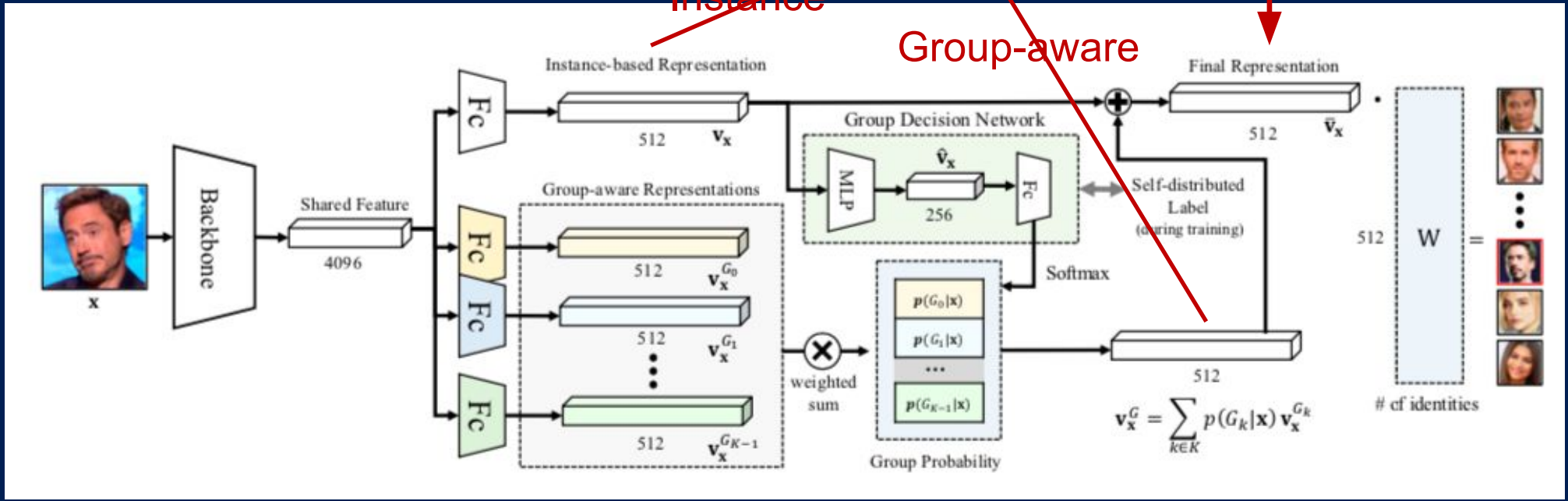
Group-aware representation

$$p(\mathbf{y}_i|\mathbf{x}) = \text{softmax}_k(g(\bar{\mathbf{v}}_\mathbf{x})),$$
$$= \text{softmax}_k(g(\mathbf{v}_\mathbf{x} + \mathbf{v}_\mathbf{x}^G)),$$

Instance

Group-aware

Final



Proposed Method

Group-aware representation

$$p(\mathbf{y}_i|\mathbf{x}) = \mathrm{softmax}_k(g(\bar{\mathbf{v}}_\mathbf{x})),$$
$$= \mathrm{softmax}_k(g(\mathbf{v}_\mathbf{x} + \boxed{\mathbf{v}_\mathbf{x}^G})),$$

V^G 는 어떻게 구할까 ?

1. S-GroupFace aggregates multiple group-aware representations with corresponding probabilities of groups as weights, and is defined as:

$$\mathbf{v}_\mathbf{x}^G = \sum_{k \in K} p(G_k|\mathbf{x})\mathbf{v}_\mathbf{x}^{G_k}. \quad (3)$$

→ improvement

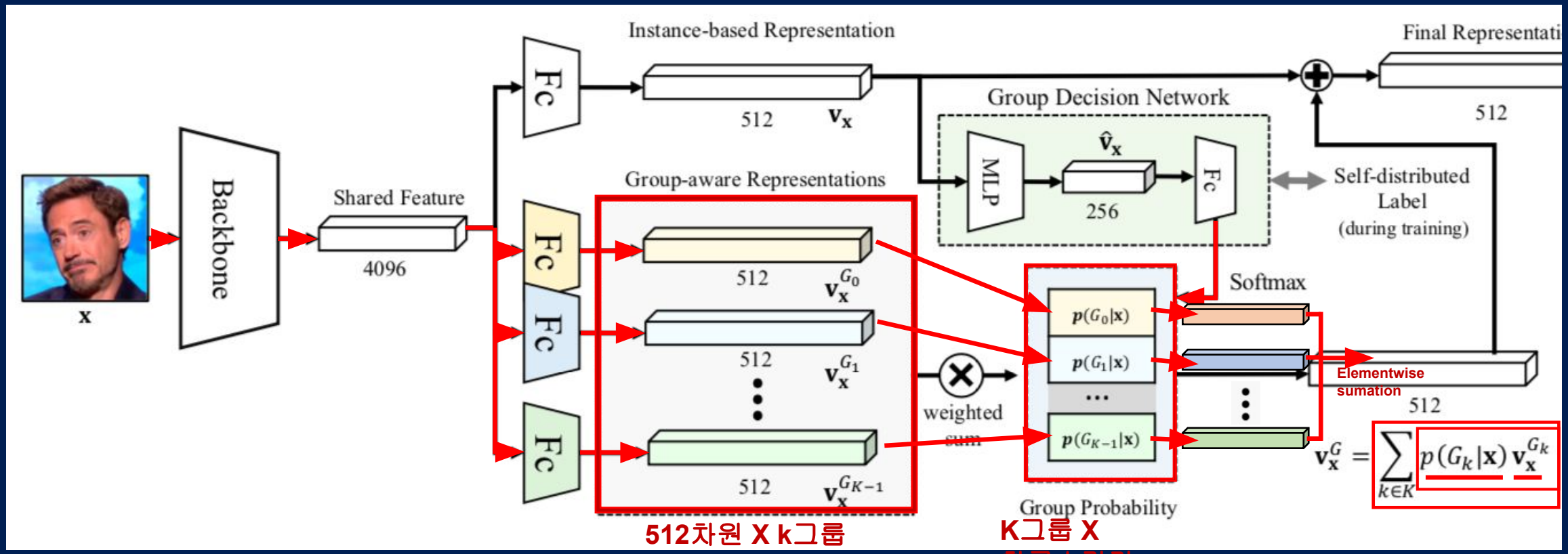
2. H-GroupFace selects one of the group-aware representations for which the corresponding group probability has the highest value, and is defined as:

$$\mathbf{v}_\mathbf{x}^G = \arg\max_{p(G_k|\mathbf{x})} \mathbf{v}_\mathbf{x}^{G_k}. \quad (4)$$

→ practical

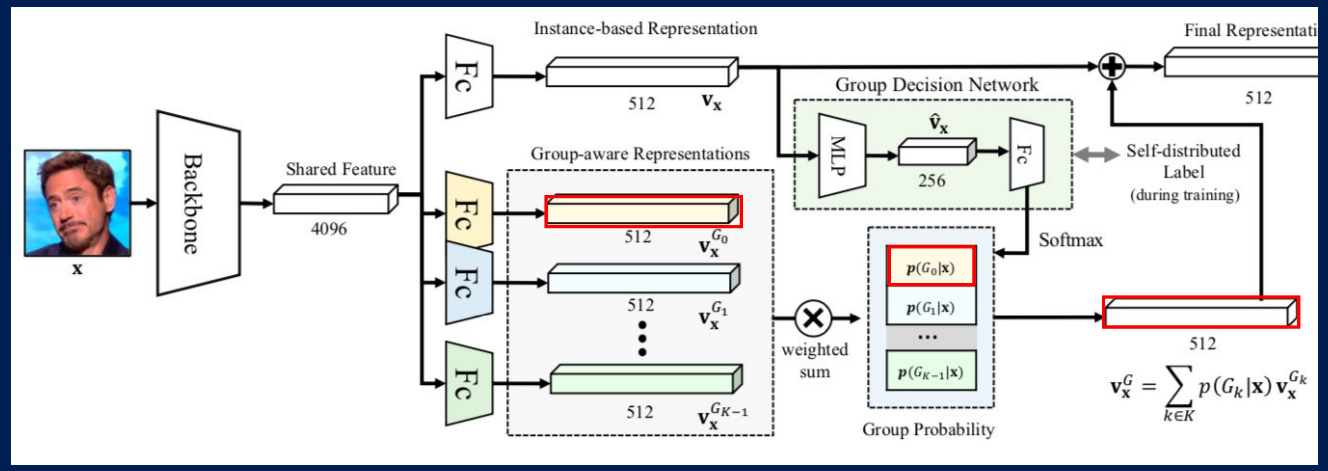
Proposed Method

S-GroupFace



Proposed Method

S-GroupFace




```
#group ensemble
group_mul_p_vk = list()
for k in range(self.groups):
    Pk = group_prob[:,k].unsqueeze(dim=-1).expand(B,512)
    group_mul_p_vk.append(torch.mul(v_G[k], Pk))
group_ensembled = torch.stack(group_mul_p_vk).sum(dim=0)
```

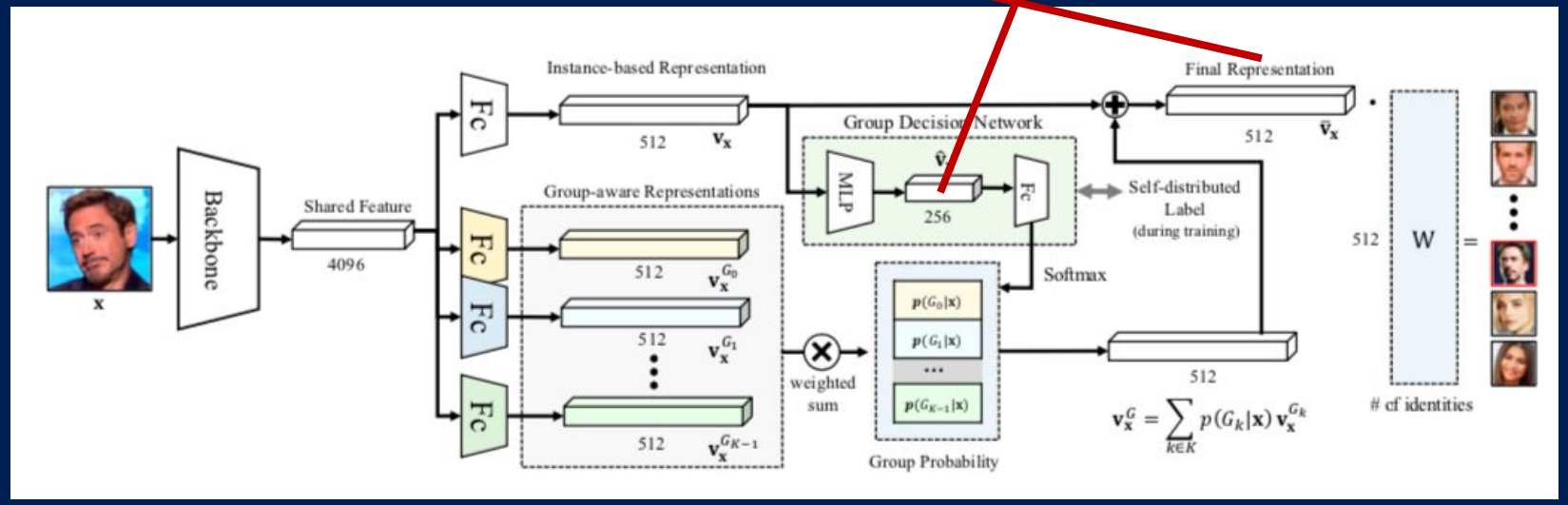
Proposed Method

Group-aware similarity

💡 새로운 similarity metric 제안

similarity = Instance similarity + group similarity

$$S^*(\mathbf{x}_i, \mathbf{x}_j) = S(\bar{\mathbf{v}}_{\mathbf{x}_i}, \bar{\mathbf{v}}_{\mathbf{x}_j}) - \beta D(\hat{\mathbf{v}}_{\mathbf{x}_i}, \hat{\mathbf{v}}_{\mathbf{x}_j})^{\gamma}, \quad (5)$$



Proposed Method

Self-distributed Grouping

Group : 같은 visual, non-visual 특징을 가진 것들의 집합

GDN
+ self-grouping
+ latent groups

💡 latent 로부터 explicit 한 정보없이 self-training

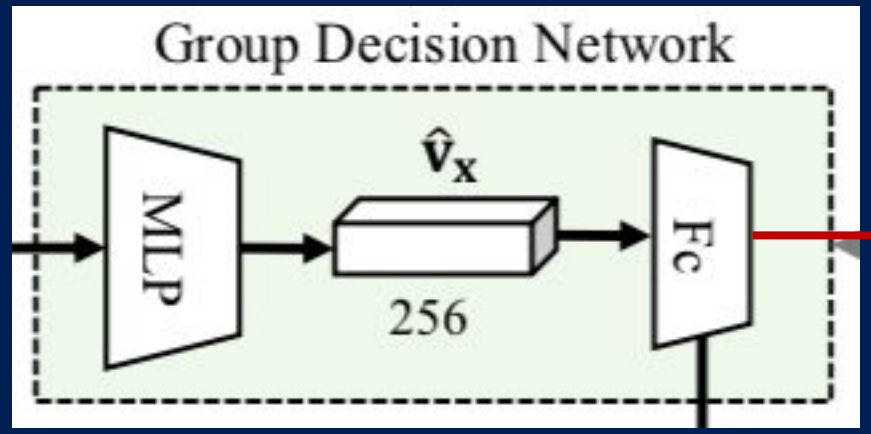
Proposed Method

Naïve Labeling

❗ 특정 group 에 분포가 치우쳐지게됨.

$$p(G_k|\mathbf{x}) = \mathrm{softmax}_k(\boxed{f(\mathbf{x})}),$$

$$G^*(\mathbf{x}) = \arg\max_k p(G_k|\mathbf{x}),$$



확률 값이 가장 큰
그룹 k
를 고른다

Proposed Method

Self-distributed Labeling

⛳ group을 Uniformly-distributed 하게 라벨링을 하자!
uniform distributed label $\tilde{p}$

🤔 Expectation normalized probability

$$\tilde{p}(G_k|\mathbf{x}) = \frac{1}{K}\left\{p(G_k|\mathbf{x}) - E_{\mathbf{x}\sim\text{data}}\left[p(G_k|\mathbf{x})\right]\right\} + \frac{1}{K},$$

Proposed Method

Self-distributed Labeling

$$\tilde{p}(G_k|\mathbf{x}) = \frac{1}{K}\{p(G_k|\mathbf{x}) - E_{\mathbf{x}\sim\text{data}}[p(G_k|\mathbf{x})]\} + \frac{1}{K},$$

⛳ $\tilde{p}$ 에서 그룹 k 에 속한 sample 이 일정하게 분포되도록

➡ $\tilde{p}$ 의 기댓값이 $\frac{1}{k}$ 이면 된다!

$$\begin{aligned} E_{x\sim data}[\tilde{p}(G_k|x)] &= E_{x\sim data}\left[\boxed{\frac{1}{K}\{p(G_k|x) - E_{x\sim data}[p(G_k|x)]\} + \frac{1}{K}}\right] \\ &= \frac{1}{K}\{\cancel{E_{x\sim data}[p(G_k|x)] - E_{x\sim data}[p(G_k|x)]}\} + \frac{1}{K} \\ &= \frac{1}{K} \end{aligned}$$

Proposed Method

Self-distributed Labeling

$$\tilde{p}(G_k|\mathbf{x}) = \frac{1}{K}\{p(G_k|\mathbf{x}) - \boxed{E_{\mathbf{x}\sim\text{data}}[p(G_k|\mathbf{x})]}\} + \frac{1}{K}$$

Probability $p(G_i|\mathbf{x})$

	G_0	G_1	G_2	G_3
$\mathbf{x}_0$	0.10	0.20	0.40	0.30
$\mathbf{x}_1$	0.20	0.25	0.20	0.35
$\mathbf{x}_2$	0.20	0.20	0.10	0.50
$\mathbf{x}_3$	0.20	0.30	0.10	0.40
$\mathbf{x}_4$	0.10	0.50	0.30	0.10
$E[p(G_i\|\mathbf{x})]$	0.16	0.29	0.22	0.33

More Uniformly Distributed Label

Expectation-Normalized Probability $\tilde{p}(G_i|\mathbf{x})$

	G_0	G_1	G_2	G_3
$\mathbf{x}_0$	0.19	0.16	0.43	0.22
$\mathbf{x}_1$	0.29	0.21	0.23	0.27
$\mathbf{x}_2$	0.29	0.16	0.13	0.42
$\mathbf{x}_3$	0.29	0.26	0.13	0.32
$\mathbf{x}_4$	0.19	0.46	0.33	0.02
$E[\tilde{p}(G_i\|\mathbf{x})]$	0.25	0.25	0.25	0.25

```python
#self distributed labeling
P = group_prob.data
E = group_label_p.mean(dim=0)
P_u = (P - E.unsqueeze(dim=-1).expand(self.groups,B).T)/self.groups +(1/self.groups)
G = torch.argmax(P_u, dim=1).data
```

Proposed Method

Loss function

**Fina
l**

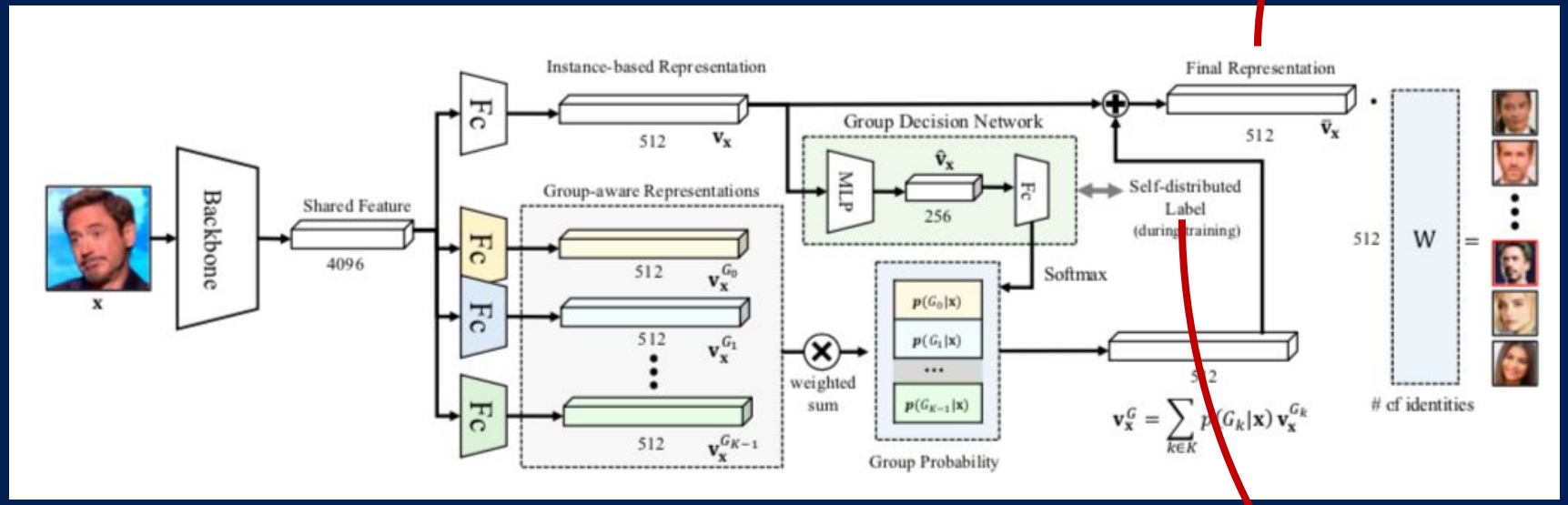
$$L_1 = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{e^{s(\cos(\theta_{\mathbf{y}_i}+m))}}{e^{s(\cos(\theta_{\mathbf{y}_i}+m))}+\sum_{j=1,j\neq\mathbf{y}_i}^{n}e^{s(\cos(\theta_j))}}, \tag{11}$$

**Self
Grouping**

$$L_2 = -\frac{1}{N}\sum_{i=1}^{N}\mathrm{CrossEntropy}(\mathrm{softmax}(f(\mathbf{x}_i)), G^*(\mathbf{x}_i)). \tag{12}$$

Proposed Method

Overview



$$L_1 = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{e^{s(\cos(\theta_{\mathbf{y}_i}+m))}}{e^{s(\cos(\theta_{\mathbf{y}_i}+m))}+\sum_{j=1,j\neq\mathbf{y}_i}^{n}e^{s(\cos(\theta_j))}}, \tag{11}$$

$$L_2 = -\frac{1}{N}\sum_{i=1}^{N}\mathrm{CrossEntropy}(\mathrm{softmax}(f(\mathbf{x}_i)), G^*(\mathbf{x}_i)). \tag{12}$$

Proposed Method

Overview

$$L = L_1 + \underset{0.1}{\lambda} L_2, \tag{13}$$

두 개 태스크에 대한 weight

Experiments

Learning

+ 8 GPUS

+ 128 images per GPU ➡️ 1024 mini-batch

+ pretrained with Softmax (group probability의 안정적인 수렴을 위해)

+ learning rate (0.005)

+ weight decay (0.0005)

+ momentum (0.9)

+ SGD

+ 최근 B-Batches 에 대해 Expection 을 구함(32와 128이 선험적으로 비슷)

Experiments

Ablation Studies

of Group

	TAR	
	FAR=1e-6	FAR=1e-5
Baseline [7]	0.3828	0.8933
- 4 Groups	0.4395	0.8962
- 16 Groups	0.4435	0.8993
- 32 Groups	0.4678	0.9115

(a) Number of Groups

Learning of GDN

	TAR@FAR=1e-6
Baseline [7]	0.3828
- without Loss	0.4468
- Naive Labeling	0.4535
- Self-distributed Labeling	0.4678

(b) Learning for GDN

Ensemble

	TAR@FAR=1e-6	GFLOPS
Baseline [7]	0.3828	24.2G
- H-GroupFace	0.4439	24.4G
- S-GroupFace	0.4678	24.5G

(c) Hard *vs.* Soft Ensemble

Aggregation vs Concat

	TAR	
	FAR=1e-6	FAR=1e-5
Baseline [7]	0.3828	0.8933
- Concat	0.4745	0.8999
- Aggregation	0.4678	0.9115

(d) Aggregation vs. Concatenataion

Group similarity

	TAR@FAR=1e-6
Baseline [7]	0.3828
- GroupFace	0.4678
- GroupFace†	0.5212

(e) Group-aware Similarity

Experiments

Evalutaion

Method	Protocol	Ident	Verif
RegularFace [44]	Large	75.61	91.13
UniformFace [9]	Large	79.98	95.36
CosFace [34]	Large	80.56	96.56
ArcFace [7]	Large	81.03	96.98
GroupFace	Large	81.31	97.35
SphereFace [19]	Large / R	97.91	97.91
AdaptiveFace [18]	Large / R	95.02	95.61
CosFace [34]	Large / R	97.91	97.91
ArcFace [7]	Large / R	98.35	98.49
GroupFace	Large / R	98.74	98.79

Table 4. Identification and verification evaluation on MegaFace. Ident means rank-1 identification accuracy (%) and Verif means TAR@FAR=1e-6 (%). R denotes the evaluation procedure on the refined version [7] of MegaFace dataset.

Experiments

Representation

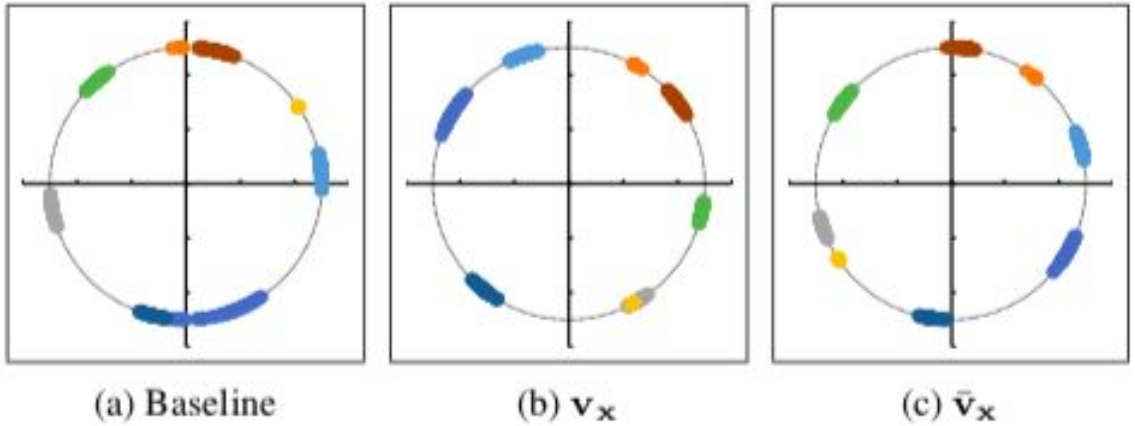


Figure 4. A quantitative comparison of the baseline network (ArcFace [7]), $\mathbf{v_x}$ that denotes an instance-based representation of GroupFace and $\bar{\mathbf{v}}_\mathbf{x}$ that denotes a final representation of GroupFace for the first eight identities in the refined MSCeleb-1M [10] dataset on a 2D space. The eight colored-circles represents the eight identities.

👋Feature 가 더욱 distinctive

👋Instance reprentation 도 향상

Experiments

Activation distribution of Groups

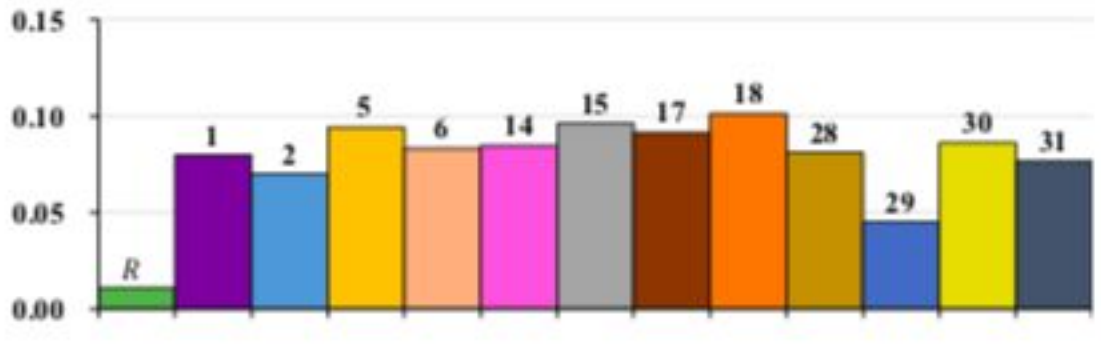
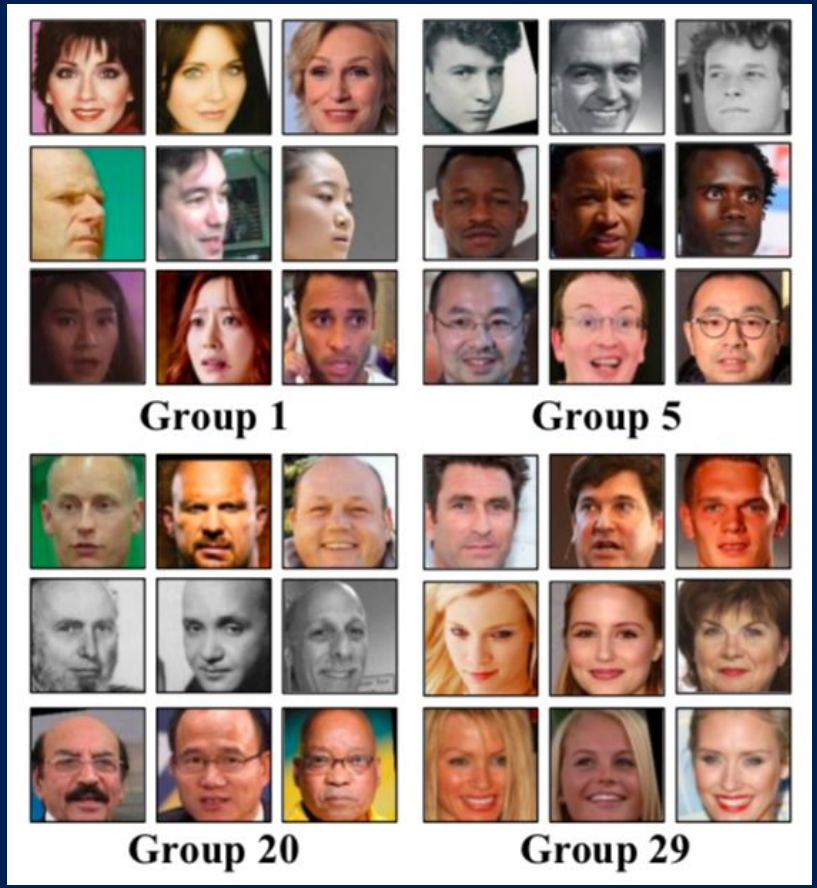


Figure 5. Average probabilities of 32 groups on the refined MSCeleb-1M [10] dataset. 12 groups can mainly activate and the rest groups, R, softly affect.

Self Grouping 은 모든 Group에 대해 Uniform
한 확률을 내놓으려고 함.
Softmax로 인해 어느정도의 불균형이 생김
모든 Group에 대해서는 Uniform 하지 않음
그렇지만 high activation 중에서도
Dominant 한 것은 없음.

Experiments

Interpretation of Groups