ScaleFace: Uncertainty-aware Deep Metric Learning

Roman Kail
Skoltech
Moscow, Russia
roma.vkail@gmail.com

Kirill Fedyanin
Technology Innovation Institute
Abu Dhabi, United Arab Emirates
kirill.fedyanin@gmail.com

Nikita Muravev
Lomonosov Moscow State University
Moscow, Russia
ne-ki-tos@yandex.ru

Alexey Zaytsev
Skoltech
Moscow, Russia
a.zaytsev@skoltech.ru

Maxim Panov
TII†, MBZUAI
Abu Dhabi, United Arab Emirates
panov.maxim@gmail.com

Abstract—The performance of modern deep learning-based systems dramatically depends on the quality of input objects. For example, face recognition quality is lower for blurry or corrupted inputs. Moreover, it is difficult to predict the influence of input quality on the resulting accuracy in more complex scenarios. We propose a deep metric learning framework that allows for direct estimation of the uncertainty with almost no additional computational cost. The developed ScaleFace algorithm uses trainable scale values that modify similarities in the space of embeddings. These input-dependent scale values represent a measure of confidence in the recognition result, thereby providing provably reasonable uncertainty estimation. We present results from comprehensive experiments on open-set classification tasks, including face recognition, which demonstrate the superior performance of ScaleFace compared to other uncertainty-aware face recognition approaches. We also extend our study to the task of text-to-image retrieval, showing that the proposed approach outperforms competitors by significant margins.

Index Terms—open set, metric learning, uncertainty estimation

I. INTRODUCTION

Deep metric learning [17], [18] is currently the leading approach for performing machine learning in such challenging scenarios as open-set classification [8], [23] and object retrieval [29]. Unlike the standard closed-set classification, the above-mentioned problems require the models to work with classes different from the ones used during training as the number of classes is huge, and new ones may emerge. Importantly, there is still limited progress in the direction of uncertainty estimation for open-set tasks aiming at mitigating risks and increasing the robustness of the resulting systems.

The standard approach in deep metric learning is to use a so-called backbone model that produces embeddings. One can then compare the obtained embeddings to decide whether a corresponding pair of objects belongs to one class. In more formal terms, one-nearest-neighbor classification for some distance between embeddings is performed.

In this work, we focus on uncertainty estimation for open-set recognition models. Uncertainty estimation methods aim to assess the confidence in the prediction for particular input objects. We argue that uncertainty estimates for such models are of great importance. For example, face recognition systems can report high similarity scores not only for images with the same identity, but also for low-quality (e.g., blurry or noisy) images of different identities thus producing false positives [32].

Importantly, the majority of existing uncertainty estimators are designed to work in more traditional closed-set classification scenarios. In such tasks, training and test sets of objects share the same set of classes. For closed-set uncertainty estimation, one can use output probabilities as a strong uncertainty estimation baseline [27]. This paper shows that existing open-set pipelines – where instead of probabilities of classes, we have distances between objects – make the problem of uncertainty estimation considerably more complicated. The existing approaches’ benefits in quality [4], [25], [32] are usually moderate, while computational complexity is often much higher than those of standard methods such as ArcFace [5] or CosFace [34].

In this work, we develop ScaleFace, a new method for deep metric learning that aims to provide computationally efficient uncertainty estimates while simultaneously improving downstream task quality. The idea is to make the scale value in ArcFace loss function [5] to be an object-dependent quantity rather than a hyperparameter. The approach requires only a small modification of existing metric learning pipelines, as scale value can be computed by a small separate head of a network. Thus, both training and inference time for ScaleFace is almost identical to those of the original ArcFace. The key contributions of this work are as follows.

1) We introduce ScaleFace, a new deep metric learning method that has natural uncertainty estimation capabilities, is computationally efficient and easy to apply. (Section II describes the method in detail).
2) We perform a careful experimental evaluation of ScaleFace on face recognition problems (in Section III) and show that it outperforms the competitors.
3) We extend the method to the problem of text-to-image retrieval and show its efficiency in this task (Section IV).

† Research was conducted while working at Technology Innovation Institute

Equal contribution
II. METHODS

A. ArcFace model

In a training dataset \( \{(x_i, y_i)\}_{i=1}^N \), \( x_i \) is a description of an object (e.g., an RGB image) and \( y_i \in \{1, \ldots, C\} \) is a class label. Here, \( C \) is the total number of classes in the training data. We consider models that transform \( x \) into an embedding \( e(x) \in \mathbb{R}^d \) via an encoder network. Here \( d \) is an embedding dimension (usually equal to 512). The standard classification loss function is a softmax loss, given by the following formula:

\[
L = - \frac{1}{N} \sum_{i=1}^{N} \log \frac{e^{\langle e(x_i), e_{y_i} \rangle}}{\sum_{j=1}^{C} e^{\langle e(x_i), e_j \rangle}}, \tag{1}
\]

where \( e_j \in \mathbb{R}^d \) is the centroid vector for the class \( j \), and \( e_i = e(x_i) \in \mathbb{R}^d \) is an embedding of the \( i \)-th object by an encoder. We absorb the bias term in vectors \( e_j \) to simplify the notation. Essentially, if we denote \( C = \{e_j\}_{j=1}^{C} \), then the logits of classes for object \( x_i \) are given by \( l_i = C e_i \), with \( C \in \mathbb{R}^{C \times d} \) being parameters of the last (fully connected) layer of a network.

The ArcFace model [5] suggests normalizing both embedding vectors \( e_i \) and class centroids \( w_j = e_j / \|e_j\| \) to have unit l2-norm. As a result, scalar product \( l_{ij} = \langle e_i, w_j \rangle \) boils down to the cosine similarity between embedding and class vectors: \( l_{ij} = \cos \theta_{ij} \), where \( \theta_{ij} \) is the angle between vectors \( e_i \) and \( w_j \). The vector of logits is multiplied by scale constant \( s \). In the original article, the scale equals 64. The resulting vector is passed to the softmax function and then to the cross-entropy loss function:

\[
L = - \frac{1}{N} \sum_{i=1}^{N} \log \frac{e^{s \cos \theta_{iy_i} + m}}{e^{s \cos \theta_{iy_i}} + \sum_{j \neq y_i} e^{s \cos \theta_{ij}}} \tag{2}
\]

ArcFace also adds margin \( m \) to the terms in the loss related to the true class in order to achieve small intra-class distances. ArcFace model has shown impressive performance in applications. However, it was shown in [12] that the softmax function can be rewritten in the following form:

\[
\log \frac{e^{\langle e(x), e_i \rangle}}{\sum_{j=1}^{C} e^{\langle e(x), e_j \rangle}} = \log \frac{e^{\langle e(x), z_j \rangle}}{\sum_{j=1}^{C} e^{\langle e(x), z_j \rangle}}, \tag{3}
\]

where \( z_j = e_j + v \) for some vector \( v \) and \( \|z_j\| = t, j = 1, \ldots, C \), and \( t \) being some constant. It means that softmax confidence is high for points \( e(x) \), which are well aligned with the centroid direction \( z_j \) and have a high norm. However, the ArcFace model ignores the norm completely, thus losing an important degree of flexibility. Increasing the norm, one can make the distribution of probabilities over classes close to one-hot, meaning that the model is confident about its prediction. Otherwise, if the norm is low, then classes have almost equal probabilities, and the model is uncertain about its decision. We build on this intuition to propose a modification to the ArcFace model that takes the norm of the predictions into account in the next section.

Figure 1: Pipeline of the training scale for predicting uncertainty. We have a two-headed backbone that predicts the embedding vector \( e(x_i) \) and the scale coefficient \( s(x_i) \). These two values are then processed the same way as in the ArcFace article so as to get the loss function. The obtained scale head facilitates fast uncertainty estimation during inference.
B. Prediction of uncertainty using scale

The entropy of the probability distribution of the classes is a strong indicator of prediction uncertainty. In the ArcFace pipeline, the scale is the parameter responsible for the entropy of the resulting distribution. Thus, by adjusting scales for individual examples, we can account for uncertainty in a meaningful way. The presented scale value is learned to detect misclassified objects and thus corresponds to total uncertainty, which includes both aleatoric and epistemic ones. In particular, the scale may take large values both for the objects on the border between classes (high aleatoric uncertainty) and for the out-of-distribution objects (high epistemic uncertainty).

To compute object-dependent scale values, we suggest training an extra head of the network. In our implementation, this subnetwork takes activations from the penultimate layer of the backbone, transforms them via multilayer perceptron, and then predicts the scale coefficient for each image individually. The training pipeline for the network with the scale-predicting subnetwork is shown in Figure 1. The training procedure remains the same, except that the scale becomes a value predicted by a separate head of the network.

Now consider an input object \( x \) that we process with a two-headed backbone to get the l2-normalized embedding vector \( e(x) \) and scale coefficient \( s(x) \). We can then compute the corresponding modification of (2) that takes into account object-dependent scale coefficients:

\[
L = -\frac{1}{N} \sum_{i=1}^{N} \log \frac{e^s(x_i) \cos(\theta_{i,j} + m)}{\sum_{j \neq y_i} e^s(x_i) \cos(\theta_{i,j}) + \sum_{j \neq y_i} e^s(x_i) \cos(\theta_{i,j})}.
\]

Consider the object \( x_i \) for which the model has selected the right class. In order to further minimize the loss, it is beneficial for the model to predict the high value of the scale coefficient \( s(x_i) \). Otherwise, if the model misses the target class, predicting the low scale value \( s(x_i) \) is beneficial. We call the resulting model ScaleFace.

C. Applications of the learned scale values

1) Correspondence between scale and error probability: In the previous section, we have proposed a method that introduces learnable scale in the ArcFace model. This section shows that the learned scale value can be useful in uncertainty estimation. We prove that the error probability of a classifier is a decreasing function of the scale under common assumptions about the embedding space. Given this, the small scale values indicate a high probability of errors making ScaleFace exceptionally well-suited for the task of uncertainty estimation for the open-set classification.

Let us start with assumptions on data generation:

(A1) For each class \( j \), a unit vector \( w_j \) represents the center of the class in the embedding space.

(A2) Observations \( e = e(x) \) in embedding space are given by the formula \( e = sw + \varepsilon \), where \( w = w_j \) for some class \( j \). \( s = s(x) \) is the scale and \( \varepsilon \sim N(0, \sigma^2 I) \).

(A3) For a large number of classes, a vector \( e \) from embedding space belongs to \( j \)-th class if \( t = \langle e, w_j \rangle / \|e\|_2 > a_j \), where \( a_j \) is a threshold for the \( j \)-th class.

The data generative model that follows from these assumptions is presented in Figure 2.

Assumptions (A1)-(A2) are typical for uncertainty estimation papers, where each object is associated with a multivariate Gaussian distribution in the embedding space [32]. However, we add a specific structure separating unit-norm centroid \( w \) and object-specific scale \( s \) that is natural for our setup. Assumption (A3) simplifies the decision rule for ArcFace to allow for analysis. The actual decision boundary for ArcFace models is more complicated and is represented by a Voronoi-type tessellation of the space with centroids \( z_j \), see (3) and [12].

Given these assumptions, we infer the error probability for such a classifier as a function of the scale \( s \):

\[
P(t < a \mid s) \approx 2 \left( 1 - \Phi \left( \frac{s}{\sigma}(1-a) \right) \right),
\]

where \( \Phi(\cdot) \) is the cumulative density function of the standard Gaussian distribution. It is easy to see that the error probability is monotonic in \( s \): as the scale \( s \) increases, the error probability decreases. More formally, the following statement holds.

**Theorem 1.** Let assumptions (A1)-(A3) hold. Then, for the error probability, it holds that if \( s_1 > s_2 \), then \( P(t < a \mid s_1) < P(t < a \mid s_2) \).

**Proof.** We can approximate the score \( t \) in the following way:

\[
t \approx \hat{t} = \sum_{i=1}^{d} w_i(s(w_i + \varepsilon_i)/s, \text{assuming that the perturbation by } \varepsilon \text{ is small compared to the vector } w. \text{ Then, } \sum_{i=1}^{d} w_i(s(w_i + \varepsilon_i)/s) = 1 + \frac{1}{s} \sum_{i=1}^{d} w_i \varepsilon_i. \text{ The noise values } \varepsilon_i \text{ are independent, and we can perform a Gaussian approximation for the distribution of } \hat{t}: \hat{t} \sim N(1, \frac{\sigma^2}{s^2}). \text{ as } \|w\|_2 = 1.
\]
While usage of the scale as an uncertainty measure is promising, suggested by the backbone, we were predicting logits, each equaling \( l_j(x) = s(x)(e(x), w_j) \), where \( e(x) \) is the L2-normalized vector, predicted by the backbone, \( w_j \) is the centroid vector corresponding to the \( j \)-th class, and \( s(x) \) is the predicted scale coefficient. This function represents the usual cosine similarity between vectors \( e(x) \) and \( w_j \), but adjusted by the scale coefficient \( s(x) \). The immediate idea is to use the modified similarity at the inference stage as well. Here we consider two possible scenarios:

1) We aim to compare two objects \( x_1 \) and \( x_2 \), taking into account uncertainties for both of them. Here, we assume that the similarity measure is used to solve a binary classification problem, distinguishing pairs belonging to one identity (positive class) or different identities (negative class). Then, we suggest considering the following similarity measure

\[
s(x_1, x_2) = \langle e(x_1), e(x_2) \rangle, \tag{6}
\]

where \( s(x_1, x_2) \) is some function computed based on the scales \( s(x_2) \) and \( s(x_2) \). For example, one may consider

\[
s(x_1, x_2) = \sqrt{s(x_1)^2 + s(x_2)^2}.
\]

2) We compare query object \( x \) with some template class object representation \( u \), for which we are sure of the quality of representation. In this case, we simply consider

\[
s(x) = \langle e(x), u \rangle. \tag{7}
\]

Such a similarity measure most closely resembles the one used during training.

Let us note that while modified similarities (6) and (7) represent the essential idea of pushing uncertain objects to have low similarity, in practice, the decision boundary between classes has a substantial positive value. That is why the direct application of these formulas may lead to many positive examples receiving similarities lower than the class-separation threshold. In order to overcome this issue, we propose introducing the shift parameter \( \mu > 0 \) and considering a modification of the similarity measure (6) with the following form:

\[
s(x_1, x_2) = \langle e(x_1), e(x_2) \rangle - \mu. \tag{8}
\]

By tuning parameter \( \mu \), we can try to push uncertain pairs closer to the class-separation border and confident pairs further from it. We can modify the template-based similarity measure (7) in the same way. The selection of parameter \( \mu \) can be done based on training or validation data (if available) in the following way:

1) We assume that a validation dataset \( D_{val} = \{ (x_{i1}, x_{i2}, y_i) \}_{i \in val} \) is available. Validation set, which should differ from the training set, will be used to find the class-separating threshold.

2) For each pair in the validation dataset, we calculate cosine similarity \( d_i = \langle e(x_{i1}), e(x_{i2}) \rangle \). Our validation sample consists of two subsamples: \( D^+_{val} \) and \( D^-_{val} \) containing positive and negative pairs, respectively. We then calculate mean similarity for \( \mu^+ = \frac{1}{|D^+_{val}|} \sum_{i \in D^+_{val}} d_i \) positive and \( \mu^- = \frac{1}{|D^-_{val}|} \sum_{i \in D^-_{val}} d_i \) negative classes for validation data. Then we average these

![Figure 3: Error probability for different decision thresholds \( a \) and different scales \( s \), as predicted by equation (5). We can see that a bigger scale leads to a lower error probability.](image)
two class centers to get a class-separating threshold \( \mu = \frac{1}{2}(\mu^+ + \mu^-) \).

3) We use the computed value of \( \mu \) to modify the similarity measure according to the equation in the main text. For example, for a pair of objects \((x_{j1}, x_{j2})\) we can compute \( \tilde{d}_j = s(x_{j1}, x_{j2})((e(x_{j1}), e(x_{j2})) - \mu) \) and use these new similarities \( \tilde{d}_j \) for open-set recognition.

In Section III-E, we will show how the proposed metrics improve the resulting quality of recognition.

### III. Open-Set Experiments

#### A. Overview

We begin experiments in Section III-C with a couple of basic experiments to show that images with higher scale values are more recognizable to the human eye. Additionally, we show that more complex datasets get lower scales (i.e., higher uncertainties), on average, from the ScaleFace model.

Another useful property of uncertainty estimates is that it allows a model to say, “I don’t know.” One way to quantify it is to drop part of the worst predictions; the faster the key metric grows, in this case, the better. As a key metric for face verification, we take the commonly used TAR@FAR (true acceptance rate at fixed false acceptance rate) and present the efficiency of the proposed approaches in Section III-D.

Finally, Section III-E shows that, even without any rejection, the \( \mu \)-ScaleFace method improves the key metrics.

#### B. Experimental setup

**Datasets.** For training of all models, we use MS1MV2 dataset [5] which is the revised version of MS-Celeb-1M dataset [10]. It contains the data of approximately 85K identities, with each identity having roughly 100 facial images.

For evaluation, we use the common IJB-C dataset [24] (3.5K identities and 148.8K images) and cross-pose LFW [41] (2.3K identities, 6K images). For preprocessing, we follow a pipeline similar to [25]. In particular, we use MTCNN face detector [38] to crop the images from the larger photo. Then we apply an affine transformation to move the 5 facial keypoints to the predefined positions.

**Uncertainty estimation approaches.** We consider the following uncertainty estimation approaches:

- Norm [37]: the inverse of the norm of the ArcFace embedding before normalization;
- PFE [32]: Probabilistic Face Embeddings;
- MagFace [25]: margin-based uncertainty estimate;
- ScaleFace (ours): an approach introduced in Section II-B;
- \( \mu \)-ScaleFace (ours): an approach with the threshold \( \mu \) selected using validation data, see Section II-C2.

All the methods except for MagFace [25] share the same trained ArcFace iResNet-50 backbone. MagFace trains the backbone and the uncertainty estimation module simultaneously, so its recognition quality differs from that of ArcFace and other methods for the same backbone architecture.

For all the methods, we use the hyperparameters and architectures suggested by their authors. For ScaleFace, we take multilayer perceptron \( s(x) \) with two hidden layers for the computation of scale values based on the embeddings from the penultimate level of the backbone.

#### C. Qualitative Experiments

1) **Face quality assessment:** In the first step, we want to show that the proposed ScaleFace method provides perceptually reasonable uncertainty estimates. We provide mean faces for each uncertainty bin, similarly to [25]. We predict confidence for each image from the IJB-C dataset. We then split images into eight bins, according to the confidence and averaged images in each bin pixel-wise.

Figure 4 presents resulting mean images. We see that the mean for the least confident images is blurry, whereas for the mean of the most confident images we see a readable face. We contend, that the most certain images are typically mug shots with clearly distinguishable facial features, while among images with low confidence there are many images with profile views, blurriness or other kinds of corruption or distortion.

Another way to make a sanity check it to divide images in the test sample by deciles of scale-based uncertainty \( u(x) \) and uniformly randomly selected images from these deciles.

Figure 5 presents five examples of images from the top decile and five examples of images from the bottom decile. As we can see, the computed scale values provide an adequate representation of the quality of the image. Images with high uncertainty are blurry or dark or only partly reveal the face. In contrast, images with low uncertainty allow for easy identification of the depicted persons.

2) **Comparison of uncertainties distributions for different datasets:** Figure 7 presents a comparison of histograms of ScaleFace’s confidence values for different considered datasets. For better presentation, we apply monotonic Box-Cox transformation [2] with \( \lambda = 3 \) to predicted scale values and linearly normalize results in a similar way for all datasets to constrain produced confidence values to the interval \([0, 1]\).

For LFW and MS1MV2, the histograms are quite close to each other, with an apparent shift towards more confident decisions. For MS1MV2, this is due to the fact that this dataset was used for training, while LFW dataset is known to be a relatively easy one for face recognition. In contrast, the IJB-C histogram is shifted to the left; we are less certain about decisions for images in this dataset as it is highly complicated real-world dataset. These results align with the existing research [25], [32] on the complexity of these datasets.

#### D. Reject verification

1) **Reject verification metric:** We consider the so-called reject verification evaluation procedure as a main tool for assessing the quality of uncertainty estimates in the context of open-set recognition. We describe it briefly below.

We consider a test dataset \( D_{test} = \{(x_{i1}, x_{i2}), y_i\}_{i=1}^N \) consisting of pairs of images \((x_{i1}, x_{i2})\) and labels indicating whether these images belong to one identity (\( y_i = 1 \)) or not (\( y_i = 0 \)). For each sample the backbone assigns a similarity score \( p_i = \langle e(x_{i1}), e(x_{i2}) \rangle \). Thus, we have predictions \( p_i \) and
target labels $y_i$ and can compare them via metrics for the binary classification problem. In this work, we use the true acceptance rate for a fixed false acceptance rate TAR@FAR, which is a common metric for the face verification task. Additionally, for each image $x$, the uncertainty estimator assigns a value $u(x)$ that represents the uncertainty of the backbone in the predicted embedding. Thus, for a pair of images $(x_{i1}, x_{i2})$, we get the uncertainty $u(x_{i1}, x_{i2})$ as the geometric mean of the uncertainties $u(x_{i1}), u(x_{i2})$.

We expect that pairs of images with high uncertainty have
greater chances of being verified incorrectly. We filter out a fixed share \( r \in [0, 0.5] \) of image pairs with the biggest value of uncertainty to get the subset \( D_{test}^{r} \) and calculate TAR@FAR metric on the remaining ones. We then plot the dependence of TAR@FAR on the share of rejected pairs \( r \) and calculate the area under this curve (AUC). Better uncertainty estimates lead to faster growth of the curve, as we reject more “bad” pairs, and AUC, therefore, is also bigger.

When \( r = 0 \), we use the whole sample \( D_{test} \) to get the metric, so the starting point of the curve is the same for all uncertainty estimates, if the backbone is the same. Ideally, we need to use the same backbone to calculate embeddings as both model accuracy and quality of uncertainty estimation influence the final result. To address this problem, we use the same ArcFace backbone and cosine similarity in the experiments of Sections III-D2 and III-D3. For the results with modified similarity metric, see Section III-E.

2) IJB-C reject verification: We perform the reject verification evaluation procedure described in Section III-D1. Figure 6 presents the comparison of the proposed ScaleFace method with the baselines sharing the same ArcFace backbone on IJB-C dataset. We note that in this experiment, all the approaches use the same cosine distance to compute the similarity. We see that scale-based uncertainty estimate is better for typical values of FAR: 0.0001, 0.001, 0.05, as the corresponding rejection curve is higher than that of other approaches.

One natural enhancement for face recognition is to use not a single photo, but a set of photos of the same person. The set of photos is called a “template”; in the test dataset IJB-C there is a test protocol for such case and we used it to benchmark the performance of our ScaleFace. The templates are divided into two parts – enrollment and verification. An enrollment template is used to get the precise “fingerprint” of a person to compare with later, during the verification phase. An enrollment template is usually made under optimal conditions, i.e. they have high quality and many photos for each identity. For the verification template, on the other hand, the condition and number of photos could be much worse. So in our experiment, we reject only by an estimated uncertainty for a verification photo. We also use a full template for enrollment and only a first photo of a verification template with a similar consideration in mind.

We call this approach N-to-1 test protocol, while the alternative is 1-to-1 image verification considered before. On both setups, ScaleFace-based uncertainties perform very well, see Table I. Interestingly, the approach based on the norm of the embeddings is competitive for small rejection rates, but its quality rapidly deteriorates for the larger ones, while PFE performs significantly better for an N-to-1 template setup.

3) Cross-pose LFW reject verification: One once-commonly used dataset for an open-set task is “Labeled faces in the wild” (LFW; [15]). It is rarely used nowadays, as all the considered models achieve almost perfect results for it. There are harder variations of it, i.e. cross-pose and cross-age LFW [41]. In these datasets, positive pairs have two semantically different photos, e.g. a person is much younger in one of the images or photos were made from completely different positions. We decided to run a test for reject verification, as this would test how well ScaleFace and other methods work with semantically hard photos.

Table I: AUC under rejection TAR@FAR curve on IJB-C test dataset. The first part is for 1-to-1 image pairs. The second part is for template N-to-1 face verification. We run experiments for different FARs for rejection portions from 0 to 0.5. Best values are in bold and second-best values are underscored. Results are normalized by optimal value.

| FAR       | 0.0001 | 0.001  | 0.01   | 0.05   |
|-----------|--------|--------|--------|--------|
| Verification | ArcFace backbone | Random | 0.8080 | 0.8640 | 0.9074 | 0.9346 |
|           |        | Norm   | 0.9064 | 0.9378 | 0.9608 | 0.9738 |
|           |        | PFE    | 0.9200 | 0.9418 | 0.9588 | 0.9698 |
|           |        | MagFace| 0.9068 | 0.9318 | 0.9520 | 0.9682 |
| ScaleFace (ours) | | 0.9366 | 0.9554 | 0.9706 | 0.9794 |
| Template verification | ArcFace backbone | Norm   | 0.8872 | 0.9206 | 0.9472 | 0.9642 |
|           |        | MagFace| 0.8934 | 0.9192 | 0.9422 | 0.9586 |
|           |        | PFE    | 0.9102 | 0.9332 | 0.9524 | 0.9642 |
| ScaleFace (ours) | | 0.9166 | 0.9404 | 0.9578 | 0.9688 |

Table II: AUC under rejection curve for different TAR@FAR values for rejection portions from 0 to 0.5 on Cross-pose LFW dataset. Best values are in bold and second best values are underscored. Results are normalized by optimal value.

| FAR | 0.005   | 0.01    | 0.1     |
|-----|---------|---------|---------|
| Random | 0.7850 | 0.8294 | 0.8862 |
| Norm   | 0.8855 | 0.9215 | 0.9334 |
| PFE    | 0.8967 | 0.9261 | 0.9529 |
| ScaleFace (ours) | 0.9106 | 0.9361 | 0.9613 |
Table IV shows that µ-ScaleFace improves upon the backbone and similarity score improved by uncertainty prediction: \( \sqrt{s(x_1)s(x_2)\langle e(x_1), e(x_2) \rangle - \mu} \), \( (9) \) where \( \mu \) is a threshold that separates negative and positive pairs computed similarly to the one in the recognition task (see Section II-C2). Our experiments show that the value of \( \mu \) can be calculated on the training data. Thus, no additional validation data is required.

### B. Experiments

1) **Experimental setup:** **Datasets.** There are two datasets used in our experiments: Conceptual Captions [31] and COCO [22]. Both provide image-caption pairs and are quite popular in retrieval benchmarks. We trained and tested our models on the same dataset. **Models.** We take a pretrained CLIP ViT-B-32 [29] as an encoder backbone and train MLP uncertainty predicting heads over its embeddings. Our experiments with heads’ architectures revealed a very small impact on the final result. We take a multilayer perceptron with four hidden layers as an uncertainty predicting head and use the same architecture for both text and image heads.

**Uncertainty estimation approaches.** Some of the methods that we used in recognition can be utilized in retrieval. We tested the following methods in our benchmarks: Norm [37], a modification of the cosine similarity metric with norms of the embeddings before normalization, as in \( (9) \); PFE [32] with two separate variance predicting heads both for text and image embeddings; \( \mu \)-Scale (ours), an extension of \( \mu \)-ScaleFace described above.

**Evaluation protocol.** First, we test all methods in typical retrieval when all of those sufficiently close to the query objects are retrieved. Setting different decision thresholds we get an approximation of the precision-recall curve. One can consider the area under this curve (Pr-Re AUC) as the target metric. Second, we test all the methods in a limited retrieval setting, where we allow to retrieve at most one object per query. This setting seems reasonable, as our datasets provide only one image per caption. Here too we can plot precision-recall curves and compute the areas under them as quality metrics (Pr-Re AUC@1).

2) **Experimental results:** Our experiments demonstrate the \( \mu \)-Scale algorithm’s superiority over the competitors (see Table V) as Norm and PFE methods fail to improve the baseline. It seems that norms of CLIP’s embeddings are not good measures of uncertainty and PFE cannot learn anything from pairs of embeddings.

## V. RELATED WORK

**Open-set and face recognition.** Open-set recognition quality significantly increased after the introduction of advanced loss functions in recent years. The survey [40] documents changes in the field since the introduction of deep learning models, while...
Table V: Comparison of pretrained CLIP with its uncertainty-aware margins during training. Finally, even a simple l2-norm paper \cite{11} proposes to boost ArcFace model by selecting class-margins correspond to high certainty of predictions. Also, the loss function as the measure of confidence: large predicted papers \cite{19}, \cite{25} consider the margin term in the ArcFace theoretically plausible probabilistic distribution on a sphere. SCF \cite{20} paper provides another view for this problem by considering more compared to non-probabilistic approaches. SCF \cite{20} and VMF-loss \cite{30}. As far as we know there are only a few studies related to uncertainty estimation open-set competitions \cite{36} and \cite{13}.

**Uncertainty estimation for open-set recognition.** However, there are only a few studies related to uncertainty estimation for open-set and face recognition; see recent surveys \cite{1}, \cite{7}.

Probably the most well-known and important work so far is the one on Probabilistic Face Embeddings model \cite{32}. It demonstrates the high quality of uncertainty estimates and improves the quality of face recognition. It was also shown to improve models in general open-set recognition setups \cite{16}, compared to non-probabilistic approaches. SCF \cite{20} paper provides another view for this problem by considering more theoretically plausible probabilistic distribution on a sphere.

Recently, several papers on face recognition \cite{11}, \cite{25}, \cite{37} advocated the idea of using not only the direction of the embedding vector, as in ArcFace \cite{5}, but also its norm. Another papers \cite{19}, \cite{25} consider the margin term in the ArcFace loss function as the measure of confidence: large predicted margins correspond to high certainty of predictions. Also, the paper \cite{11} proposes to boost ArcFace model by selecting class-aware margins during training. Finally, even a simple l2-norm of embedding from ArcFace \cite{5} was shown to be a pretty strong baseline for this task in \cite{37}.

**Scale selection in softmax.** The scale selection approach developed in this paper can be seen as setting a specific temperature for the loss function. Selection of a single temperature for the whole dataset seems to be an important factor for deep learning model calibration in general \cite{9}, \cite{26}, as well as for out-of-distribution (OOD) detection \cite{21}.

Scale selection turns to be important for open-set recognition as well \cite{39}, with optimal value depending on the resulting embedding space and training epoch. Local temperature parameters specific to each image have been used to improve calibration for segmentation problem \cite{6}. Note, that all these approaches target calibration quality improvement and require additional validation samples for temperature parameters estimation, thus complicating the overall model pipeline.

More recent works on OOD detection focus on the scale values prediction via a separate simple neural network \cite{14}, \cite{33}. One study \cite{33} suggests learning scale for image classification. However, they consider scale only on the training stage and do not use it on evaluation stage. Another approach \cite{14} goes a step further by using the learned scale values as a measure of uncertainty useful for OOD detection. However, they do not consider the open-set problem statement, where we achieve important benefits from the modified similarity metric.

**Uncertainty estimation for retrieval.** The success of probabilistic approaches in face recognition inspired a number of similar approaches in retrieval. Usually these methods are tested in class-disjoint image-to-image retrieval benchmarks \cite{16}. Among the most prominent approaches are HIB \cite{28}, PFE \cite{32}, DUL \cite{3}, SCF \cite{20} and VMF-loss \cite{30}. As far as we know none of these methods has been implemented to text-to-image retrieval, especially on CLIP-sized models and large datasets.

VI. CONCLUSIONS

In this work, we introduce ScaleFace, a new uncertainty estimation method, designed for the open-set recognition and
retrieval problems. The idea of the approach is to estimate the scale value in the ArcFace [5] loss function for every input object and to use it as a confidence measure. Additionally, we introduce the modification to cosine distance based on the computed scale values.

Our experiments examine the proposed measure and state-of-the-art uncertainty estimates from various angles and provide detailed comparisons of the methods considered. The versions of ScaleFace demonstrate significant improvements over PFE [32] and other baselines without any computationally-demanding modifications, instead using a separate head of the neural network for scale estimation. All the code to reproduce the experiments is available at https://github.com/stat-ml/scale-face.

Based on the conducted experiments, we recommend using ScaleFace method for lightweight uncertainty estimates in open-set recognition and retrieval problems. It is quick to train and provides superior uncertainty quality metrics.

ACKNOWLEDGMENTS

The research was supported by the Russian Science Foundation (project 21-11-00373).

REFERENCES

[1] Moloud Abdar, Farhad Pourpanah, Sadiq Hussain, Dana Rezaazadegan, Li Liu, Mohammad Ghavamzadeh, Paul Fieguth, XiaoChun Cao, Abbas Khosravi, U Rajendra Acharya, et al. A review of uncertainty quantification in deep learning: Techniques, applications and challenges. Information Fusion, 76:243–297, 2021. 9

[2] George EP Box and David R Cox. An analysis of transformations. Journal of the Royal Statistical Society: Series B (Methodological), 26(2):211–243, 1964. 5

[3] Jie Chang, Zhonghao Lan, Changmiao Cheng, and Yichen Wei. Data uncertainty learning in face recognition. In CVPR, June 2020. 9

[4] Kai Chen, Taihe Yi, and Qi Lv. Fast and reliable probabilistic face embeddings based on constrained data uncertainty estimation. Image and Vision Computing, page 104429, 2022. 1

[5] Jiankang Deng, Jia Guo, Nianan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In CVPR, pages 4690–4699, 2019. 1, 2, 5, 9, 10

[6] Zhipeng Ding, Xu Han, Peirong Liu, and Marc Niethammer. Local temperature scaling for probability calibration. In ICCV, pages 6889–6899, 2021. 9

[7] Biying Fu, Cong Chen, Olaf Henniger, and Naser Dumer. A deep insight into measuring face image utility with general and face-specific image quality metrics. In WACV, pages 905–914, 2022. 9

[8] Manuel Gunther, Steve Cruz, Ethan M Rudd, and Terrance E Boult. Toward open-set face recognition. In CVPR Workshops, pages 71–80, 2017. 1

[9] Chuan Guo, Geoff Pleiss, et al. On Calibration of Modern Neural Networks. In ICML, pages 1321–1330, PMLR, 2017. 9

[10] 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, pages 87–102. Springer, 2016. 5

[11] Qishen Ha, Bo Liu, Fuxi Liu, and Peiyuan Liao. Google landmark recognition 2020 competition third place solution. arXiv:2010.05350, 2020. 9

[12] Sibyle Hess, Wouter Duveestijn, and Decibel Mocanu. Softmax-based classification is k-means clustering: Formal proof, consequences for adversarial attacks, and improvement through centroid based tailoring. arXiv preprint arXiv:2001.01987, 2020. 2, 3

[13] Addison Howard. Product matching competition: recap. https://www.kaggle.com/competitions/shopee-product-matching/discussion/240667, 2021. 9

[14] Yen-Chang Hsu, Yilin Shen, Hongxia Jin, and Zsolt Kira. Generalized ODIN: Detecting out-of-distribution image without learning from out-of-distribution data. In CVPR, pages 10951–10960, 2020. 9

[15] Gary B Huang, Manu Ramesh, Tamara Berg, and Erik Learned-Miller. Labeled faces in the wild: a database for studying face recognition in unconstrained environments. university of massachusetts. Technical report, Amherst, Technical Report 07-49 (October 2007), 2020. 7

[16] Ivan Karpukhin, Stanislav Dera, and Sergey Kolesnikov. Probabilistic embeddings revisited. arXiv:2002.06768, 2022. 9

[17] Mahmut Kaya and Hasan Şakir Bilge. Deep metric learning: A survey. Symmetry, 11(9):1066, 2019. 1

[18] Chan Kha Vu. Deep metric learning: A (long) survey, 2021. 1, 9

[19] Minchul Kim, Anil K Jain, and Xiaoming Liu. AdaFace: Quality adaptive margin for face recognition. In CVPR, pages 18750–18759, 2022. 9

[20] Shen Li, Jianqing Xu, Xiaqing Xu, Pengcheng Shen, Shaoxin Li, and Bryan Hooi. Spherical confidence learning for face recognition. In CVPR, pages 15629–15637, June 2021. 9

[21] Shiyu Liang, Yixuan Li, and R Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. In ICLR, 2018. 9

[22] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In ECCV, pages 740–755, 2014. 8

[23] Carlo Masone and Barbara Caputo. A survey on deep visual place recognition. IEEE Access, 9:19516–19547, 2021. 1

[24] Brianna Maze, Jocelyn Adamis, James A Duncan, Nathan Kalka, Tim Miller, Charles Otto, Anil K Jain, W Tyler Niggel, Janet Anderson, Jordan Cheney, et al. Iarpa janus benchmark-c: Face dataset and protocol. In International Conference on Biometrics (ICB), pages 158–186, 2015. 5

[25] Qiang Meng, Shichao Zhao, Zhida Huang, and Feng Zhou. Magface: A universal representation for face recognition and quality assessment. In CVPR, pages 14225–14234, 2021. 1, 5, 9

[26] Matthias Minderer, Josip Djolonga, and Andrew C. Gallagher. Modeling uncertainty with hedged instance embedding. In ICLR, 2019. 9

[27] 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, pages 8748–8763. PMLR, 2021. 1, 8

[28] Tyler R. Scott, Andrew C. Gallagher, and Michael C. Mozer. Von-mises-fisher loss: An exploration of embedding geometries for supervised learning. In ICCV, pages 10612–10622, October 2021. 9

[29] Piyush Sharma, Nan Ding, Sebastian Goodmann, and Radu Sorici. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In ACL, pages 2556–2565, 2018. 8

[30] Yichun Shi, Anil K. Jain, and Nathan D. Kalka. Probabilistic face embeddings. In ICCV, pages 6901–6910, 2019. 1, 3, 4, 5, 8, 9, 10

[31] Engkarat Techapanurak, Masanori Suganuma, and Takayuki Okatani. Hyperparameter-free out-of-distribution detection using cosine similarity. In ACCV, 2020. 9

[32] Minjun Wang, Yitong Wang, Zheng Zhou, Xing Ji, Dihong Gung, Jingchao Zhou, Zhifeng Li, and Wei Liu. Cosface: Large margin cosine loss for deep face recognition. In CVPR, pages 5265–5274, 2018. 1

[33] Mei Wang and Weihong Deng. Deep face recognition: A survey. Neurocomputing, 429:215–244, 2021. 9

[34] Tobias Weyand, Andre Araujo, Bingyi Cao, and Jack Sim. Google landmarks dataset v2-a large-scale benchmark for instance-level recognition and retrieval. In CVPR, pages 2575–2584, 2020. 9

[35] Chang Xu, Xiangyu Zhu, Zhen Lei, and Stan Z Li. Out-of-distribution detection for reliable face recognition. IEEE Signal Processing Letters, 27:710–714, 2020. 5, 8, 9

[36] Kaipeng Zhang, Zhenpeng Zhang, Zhifeng Li, and Yu Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. IEEE signal processing letters, 23(10):1499–1503, 2016. 5

[37] Xiao Zhan, Rui Zhao, Yu Qiao, Xiaogang Wang, and Hongsheng Li. Adacos: Adaptively scaling cosine logits for effectively learning deep face representations. In CVPR, pages 10823–10832, 2019. 9

[38] Liang Zheng, Yi Yang, and Alexander G Hauptmann. Person re-identification: Past, present and future. arXiv:1610.02984, 2016. 8

[39] T. Zheng and W. Deng. Cross-pose fllw: A database for studying cross-pose face recognition in unconstrained environments. Technical Report 18-01, Beijing University of Posts and Telecommunications, 2018. 5, 7