Learning Feature Representations for Look-Alike Images

Ayça Takmaz, Thomas Probst, Danda Pani Paudel, Luc Van Gool
Computer Vision Lab, ETH Zürich
takmaza@student.ethz.ch, {probstt,paudel,vangool}@vision.ee.ethz.ch

Abstract

Human perception of visual similarity relies on information varying from low-level features such as texture and color, to high-level features such as objects and elements. While generic features learned for image or face recognition tasks somewhat correlate with the perceived visual similarity, they are found to be inadequate for matching look-alike images. In this paper, we learn the 'look-alike feature' embedding, capable of representing the perceived image similarity, by fusing low- and high-level features within a modified CNN encoder architecture. This encoder is trained using the triplet loss paradigm on look-alike image pairs. Our findings demonstrate that combining features from different layers across the network is beneficial for look-alike image matching, and clearly outperforms the standard pre-trained networks followed by fine-tuning. Furthermore, we show that the learned similarities are meaningful, and capture color, shape, facial or holistic appearance patterns, depending upon context and image modalities.

1. Introduction

Humans perceive visual similarity based on various types of internal image representations, which stem from low-level, mid-level, and high-level interpretations of images [2]. Perceived visual similarity depends on how human brain understands the given scene based on information from different-scales, along with a system which retrieves similar scenes from the memory. Mimicking such highly complex structure of visual perception to learn a visual similarity measure is quite a challenging problem for machines, while being useful for tasks such as image retrieval [6, 7]. The existing works in image retrieval deal with images from similar modalities (or portraying similar concepts), and therefore are not suitable for the perceived visual similarity measurement when images feature different subject types across diverse modalities. In such cases, we are interested to learn the feature embedding that represents the perceived visual similarities between images. Few examples of such image pairs are shown in Figure 1.

Figure 1. **Totally-Looks-Like Dataset [2].** Example image pairs.

Rosenfeld *et al*. [2] collected a dataset consisting of look-alike image pairs, paired by humans based on their perceived visual similarity of the images. The authors conduct experiments to imitate the way humans make these pairings, using features extracted from state-of-the-art convolutional neural networks (CNNs). While they believe that sufficiently generic visual features must reproduce these pairings without explicitly being trained for the task, the result obtained using feature-based pairings was found to be far from the ground truth pairings. Based on the findings from [2], we aim to examine the capability of CNNs to imitate the way humans perceive visual similarity of images across various modalities such as photos, cartoons and sketches.

2. Learning Perceived Similarity

Knowing that the available pre-trained networks do not provide satisfactory features for the look-alike image matching, we aim to directly learn an embedding that reflects the perceived similarities, using a training set of similar (positive) and dissimilar (negative) image pairs. Our approach focuses on training a CNN to map images onto a 'look-alike' feature space, where the Euclidean distances between embeddings represent the perceived visual similarity of images. We frame the problem of look-alike matching as an image retrieval task. Given a query image, we retrieve its closest look-alike from a database of images, using the feature embeddings as shown in Figure 2.

Base Architecture. We use the Inception-ResNet-v1 architecture as in FaceNet [4], due to its ability to achieve high

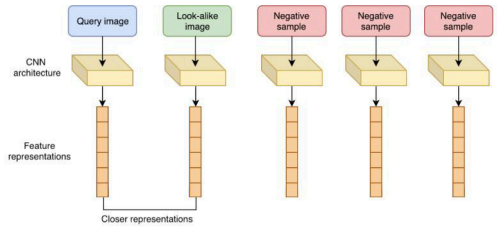

Figure 2. **Look-alike Feature Embedding.** The aim is to obtain a mapping to the look-alike feature space in which the distances between the representations of look-alike images are minimized.

performance with a small number of parameters. FaceNet directly learns how to represent face images in a compact Euclidean space in which the distances between the embeddings correspond to a measure of facial similarity. Mathematically, we represent the network with the embedding function $f : \mathcal{I} \rightarrow \mathbb{R}^d$, that maps a given RGB image $\mathsf{I} \in \mathcal{I} = \mathbb{R}^{w \times h \times 3}$ onto a d-dimensional space.

Triplet Loss and Triplet Selection. The embedding vector $f(\mathsf{I})$ obtained by the network can now be interpreted as the representation of the image I in a similarity-sensitive space. The desired property of this space is that the distance between two embedding vectors can be considered as a measure of the perceived similarity of the corresponding images. Given a pair of perceptually similar images $(\mathsf{L}_i, \mathsf{R}_i) \in \mathcal{I}^2$, we want to learn an embedding network $f(\mathsf{I})$ that minimizes the distance between the two embeddings, and that maximizes the distance between the embeddings of any two unrelated images. Triplet loss minimization [5] has proven to be a suitable strategy for such tasks.

A triplet of images $\tau_i = (\mathsf{I}_i^a, \mathsf{I}_i^p, \mathsf{I}_i^n) \in \mathcal{I}^3$, includes the anchor image, as well as a positive (the pair counterpart) and a negative image. The loss for τ_i is then computed as

$$\ell(\tau_i) = [\|f(\mathsf{I}_i^a) - f(\mathsf{I}_i^p)\|_2^2 - \|f(\mathsf{I}_i^a) - f(\mathsf{I}_i^n)\|_2^2 + \alpha]_+, \tag{1}$$

where $[\cdot]_+ = \max(\cdot, 0)$ is the hinge operation, and α is a margin parameter. For a batch of triplets $\mathcal{T}$, the triplet loss ℓ then is given by

$$\ell = \frac{1}{|\mathcal{T}|} \sum_{\tau_i \in \mathcal{T}} \ell(\tau_i). \tag{2}$$

During training, triplets are selected by hard-negative mining in order to have challenging examples for the model to learn. For each image pair in the dataset $(\mathsf{I}_i^a, \mathsf{I}_i^p) := (\mathsf{L}_i, \mathsf{R}_i)$, a negative image I_i^n is sampled randomly from the set of images which violate the margin α [4].

Hierarchical Feature Integration Architecture. With the aim of combining the outputs from low-level and mid-level layers with the final embedding, we extend the base architecture. To this end, we learn a new embedding that integrates features across multiple layers of the network hierarchy by a linear combination of the corresponding activations. Since the importance of the outputs from low-level and mid-level layers are not clear for our task, relative weights given to these layers are also being learned by the model as a set of scalar coefficients. These scalar coefficients are modeled as 1×1 Conv parameters in our network. Specifically, we extend the base architecture as follows.

- Each layer in the network stem and the reduction layers in the Inception ResNet-v1 are combined via a series of operations: output of 10 selected layers are connected to 10 separate 1×1 convolution layers with d channels for dimension reduction, followed by average-pooling operation to obtain vectors of size $1 \times d$. These vectors are only used to extract information from the layers belonging to the original architecture, and these vectors are not directly passed through to the final embedding.
- A stacking layer is used to combine the outputs from the extracted feature vectors and the FaceNet embedding vectors, yielding a matrix of size $11 \times d$.
- An 1×1 convolutional layer with 11 filters is used for dimension reduction, to obtain a final embedding of $1 \times d$. The output of this layer is the 'look-alike' feature.

The resulting architecture is depicted in Figure 3.

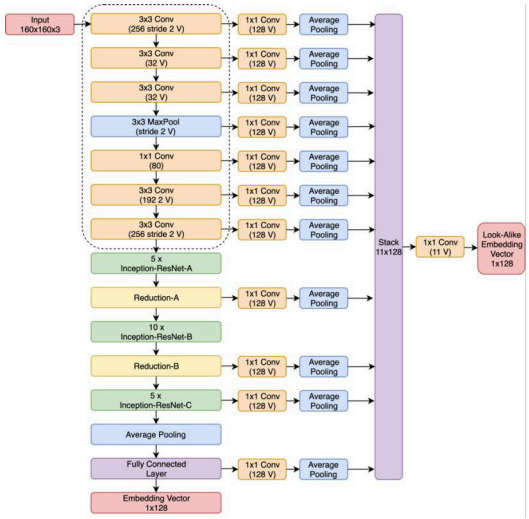

Figure 3. **Hierarchical Feature Integration.** Look-alike features are extracted and integrated across different levels of the feature hierarchy by systematically introducing skip-connections.

3. Experiments

Our experiments for the perceptual similarity measurement task can be divided into two parts. First, we obtain the baseline results using a state-of-the-art pretrained

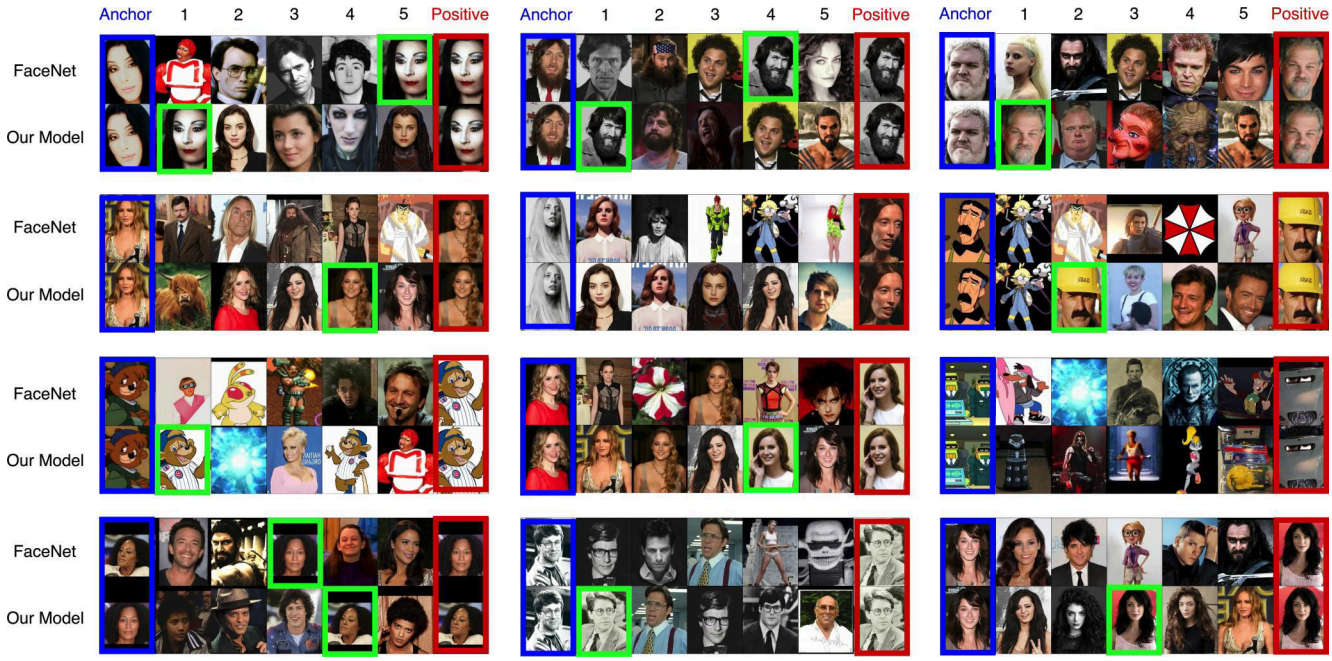

Figure 4. **Qualitative Look-Alike Retrieval Comparison.** We compare the FaceNet baseline model (top row) to our proposed feature integration technique (bottom row) on various examples from the TLL test set. Green markings indicate correct matches. Our approach retrieves more meaningful images in general, and is able to capture a wide variety of similarity types.

Method	$m = 10$			$m = 20$			$m = 60$		
	top-1	top-2	top-5	top-1	top-2	top-5	top-1	top-2	top-5
Random chance	10%	20%	50%	5%	10%	25%	2%	3%	8%
TLL Baseline [2]	38%	-	-	27%	-	-	-	-	
FaceNet	45%	58%	83%	32%	45%	65%	24%	30%	37%
FaceNet + fine-tuned on TLL	61%	72%	90%	45%	58%	75%	31%	43%	59%
Ours + fine-tuned on TLL	**64%**	**76%**	**94%**	**53%**	**65%**	**83%**	**39%**	**51%**	**69%**

Table 1. **Look-Alike Retrieval Accuracy.** We report the top-1, top-2, top-5 accuracies for different methods, including random choice, using generic+facial features from [2], FaceNet features, FaceNet features fine-tuned on the TLL dataset, and our proposed feature integration technique fine-tuned on the TLL dataset. We evaluated over test reference sets of sizes $m = \{10, 20, 60\}$.

CNN model, followed by fine-tuning using the triplet loss paradigm. Second, we integrate features across the network hierarchy, to represent images in the perceived visual similarity space, and compare against the baseline.

We use the FaceNet model with Inception-ResNet-v1 architecture; a choice made under the assumption that a network trained for face recognition can also extract features across different hierarchy levels [4]. We initialize it with weights pre-trained on the VGG-Face 2 dataset [1]. Following FaceNet, we set the embedding dimension to $d = 128$, and used the same hyperparameters, with a reduced learning rate of $\eta = 10^{-5}$ for fine-tuning. We employed a publicly available implementation in Tensorflow [3].

Dataset and Evaluation Metric. We perform training and evaluation of our models on the previously mentioned Totally-Looks-Like (TLL) dataset [2]. The TLL dataset consists of 6016 similar image pairs from the wild, which were collected from an entertainment website where users were able to vote for the visual similarity of two given images. The images within the dataset represent various modalities such as photos, cartoons, and sketches with examples including (but not limited to) human faces, cartoon characters, animals, and objects (see Figure 1). The diversity of these image pairs gives insight into the features used by humans when making similarity judgments, as the image pairs manifest several possible ways of similarities. Some examples include general facial similarity, facial feature attributions to animals and objects, similarity in terms of color, texture, and shape [2]. Hence, similarity may not be explained by only low- or high-level representations.

We split the data into 240 test and 5776 training image pairs, and employ data augmentation via vertical flips for training. The model accuracy is then evaluated by averaging over batches of m images, where $m \in \{10, 20, 60\}$, as in

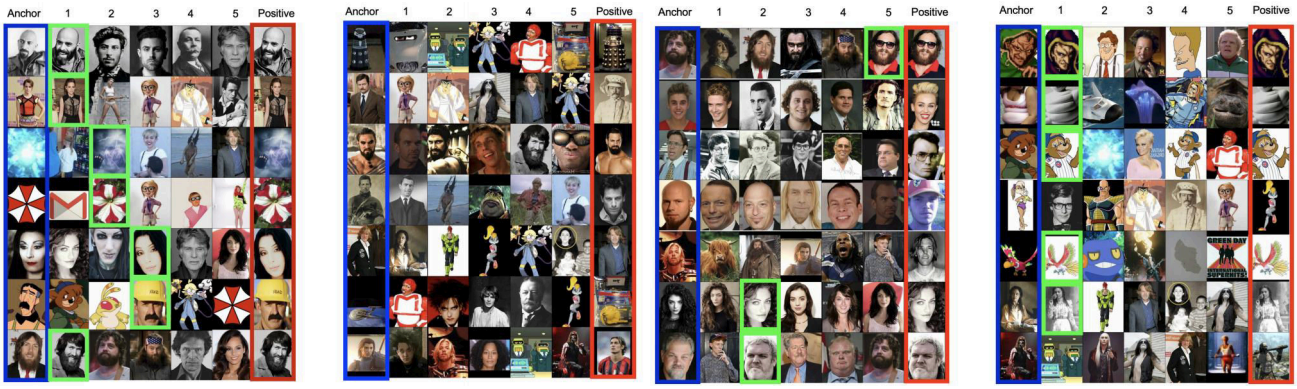

Figure 5. **Qualitative Look-Alike Retrieval Results.** Each row of the mosaics shows the anchor image, the top-5 matches, and the ground truth match (columns from left to right) of a sample using our proposed method. Green markings indicate correct matches. From left to right we show: samples with successful top-5 retrieval, failed retrievals, random faces, and random samples.

the image retrieval task. For each L_i, we randomly sample $m-1$ images from the other pairs within the test set, where these images are represented by I_k with $k \in \{1,..,n-1\}$. Then, the image R_i is also included in this image batch as I_n; hence, a reference set of images $\mathcal{S}$ of size m is obtained. After assembling the batch, the embedding vectors corresponding to these m candidates are calculated, and the pairwise distances of the embeddings (distance between $f(\mathsf{L}_i)$ and the candidate images I_k for $k \in \{1,..,n\}$ are calculated. The images I_k are ranked in terms of these distances, and the average top-1, top-2 and top-5 accuracies are computed over all test batches, which are summarized in Table 1.

FaceNet-based baseline models. We extract FaceNet features from the input images using the original FaceNet model, and calculate the similarity between images using the Euclidean distance in the feature embedding. Since our dataset includes non-face images as well as face images across various modalities, such as cartoons or sketches, we remove the face detection step from the input pipeline. The network is then trained with the TLL dataset via triplet loss learning, starting from the pre-trained weights. Doing so, yields better performance, with the ability of capturing low- or high-level similarities separately. However, we observed that the model was still inadequate to capture the mixture of low- and high-level similarities.

Hierarchical Feature Integration. For training the extended architecture, two alternatives were considered. First one was to train the whole model, together with 1×1 Conv layers, as shown in Figure 3. Second one was to freeze the weights of the layers from the initial architecture and train only the additionally introduced (1×1 Conv) layers. We found that training only the additional layers results in the best feature representation in terms of image retrieval performance. In fact, this implies that a rich feature hierarchy has already been learned, which can be exploited for 'look-alike' similarity task by designing a feature integration path in the network. Figure 4 gives a qualitative impression of the performance gain as measured in terms of accuracy. In Figure 5, we show qualitative results of our proposed method for different subsets of the test set.

4. Conclusion and Future Work

In this work, we investigated learning schemes for look-alike image matching. Our experiments show that perceived similarities appear in different levels of the feature hierarchy. This can be exploited for improved matching performance, by integrating features from multiple layers.

For future work, further insights may be obtained by localizing regions important for visual similarity. Other recognition tasks may also benefit from the proposed technique, due to the explicit learning of visual similarity.

References

[1] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman. Vggface2: A dataset for recognising faces across pose and age. In *International Conference on Automatic Face and Gesture Recognition*, 2018. 3

[2] A. Rosenfeld, M. D. Solbach, and J. K. Tsotsos. Totally looks like - how humans compare, compared to machines. *CoRR*, abs/1803.01485, 2018. 1, 3

[3] D. Sandberg. Face recognition using tensorflow. https://github.com/davidsandberg/facenet, 2017. 3

[4] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. *CoRR*, abs/1503.03832, 2015. 1, 2, 3

[5] Y. Sun, X. Wang, and X. Tang. Deep learning face representation by joint identification-verification. *CoRR*, abs/1406.4773, 2014. 2

[6] J. Wang, Y. Song, T. Leung, C. Rosenberg, J. Wang, J. Philbin, B. Chen, and Y. Wu. Learning fine-grained image similarity with deep ranking. *CoRR*, abs/1404.4661, 2014. 1

[7] W. Zhou, H. Li, and Q. Tian. Recent advance in content-based image retrieval: A literature survey. *CoRR*, abs/1706.06064, 2017. 1