Explainable Model-Agnostic Similarity and Confidence in Face Verification

Martin Knoche Torben Teepe Stefan Hörmann Gerhard Rigoll
Technical University of Munich
Arcisstrasse 23, 80333 München, Germany
Martin.Knoche@tum.de

Abstract

Recently, face recognition systems have demonstrated remarkable performances and thus gained a vital role in our daily life. They already surpass human face verification accountability in many scenarios. However, they lack explanations for their predictions. Compared to human operators, typical face recognition network system generate only binary decisions without further explanation and insights into those decisions. This work focuses on explanations for face recognition systems, vital for developers and operators. First, we introduce a confidence score for those systems based on facial feature distances between two input images and the distribution of distances across a dataset. Secondly, we establish a novel visualization approach to obtain more meaningful predictions from a face recognition system, which maps the distance deviation based on a systematic occlusion of images. The result is blended with the original images and highlights similar and dissimilar facial regions. Lastly, we calculate confidence scores and explanation maps for several state-of-the-art face verification datasets and release the results on a web platform. We optimize the platform for a user-friendly interaction and hope to further improve the understanding of machine learning decisions. The source code is available on GitHub[1] , and the web platform is publicly available at `http://explainable-face-verification.ey.r.appspot.com`.

1. Introduction

Machine learning has recently demonstrated remarkable performances in multiple tasks, from image processing to natural language processing, especially with the advent of deep learning. Along with research progress, it has influenced many fields and disciplines. For example, in the medical sector or security systems, a high level of accountability and thus greater transparency and interpretability is required. However, these systems are often considered black boxes, and it is not known what happens internally. They lack an explanation. According to Phillips *et al*. [23], an accompanying explanation needs to be interpretable and accurate, and models must operate within their known limits.

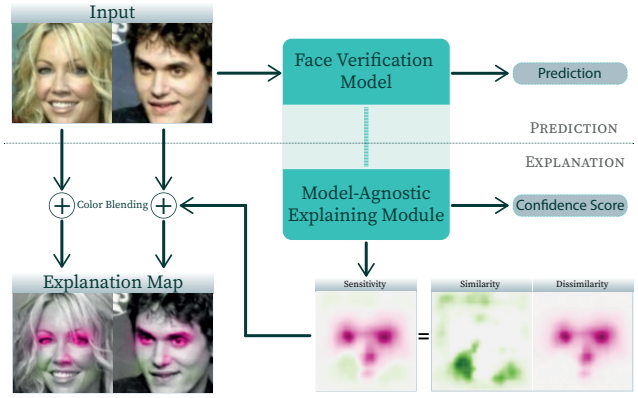

Figure 1. The proposed approach generates a similarity map and blends it with the input images into an explanation map. Besides the binary prediction of the network, we introduce a confidence score to explain the decision further.

Explainable artificial intelligence (XAI) arose from the need to understand models in various areas [8, 27]. The benefits of explanations are apparent and in recent years, more and more approaches have been introduced. With ever more explainable face recognition systems, humans are getting more involved in the decision process, which is vital for many fields of applications. But explainability is not only important for the final user, but also for developers, which can benefit from a better understanding of datasets and models. The distribution and accessibility of those explanations is necessary.

There exist model-agnostic approaches [21, 22] and the famous model-specific Gradient-weighted Class Activation Mapping approaches (GradCAM) [7,26,29], which propose saliency maps, highlighting decisive facial regions. They often require access to the layers of deep learning archi-

[1] https://github.com/martlgap/x-face-verification

tectures used by facial matchers, which is not always feasible in commercial systems. Instead, we follow the idea of Mery [22] and consider deep learning models as input-output functions, which cannot be accessed. Our model-agnostic approach focuses on the face verification problem, and we provide explanations with similarity maps, which indicate similar and dissimilar regions of the face. Instead of simply interpreting the cosine distance for the certainty of the decision, we establish a more precise confidence score calculation (see Figure 1).
Our main contributions are summarized as follows:

- We introduce a confidence score for face recognition networks.
- We provide three different explanation methods for face recognition.
- We build and release a user-friendly, interactive, and modern web platform containing the proposed confidence scores and explanation maps for several state-of-the-art datasets and models.

2. Related Work

2.1. Explanation Maps

One of the earliest approaches for explainable artificial intelligence (XAI) is the local interpretable model-agnostic explanations (LIME) technique introduced by Ribeiro *et al*. [25]. In their work, they proposed a method for faithfully explaining any classifier's predictions by learning an interpretable model locally around the prediction.

The most relevant XAI methods similar to our approach are model-agnostic algorithms:

Firstly, Mery and Morris [22] introduced six different saliency maps that can be used to explain any face verification algorithm without manipulating the model. The key idea of their method is to define a matching score of two facial images, which changes when one image is perturbed. In addition, they experimented with XAI saliency maps based on contours.

Secondly, in [21], Mery introduced an XAI method based on how the probability of recognition of a given image changes when it is perturbed. His algorithm removes and aggregates different parts of the image and then measures the contributions of those parts individually and in-collaboration as well. The generated saliency maps highlight the most relevant areas for the recognition process.

Third, the work from Lin *et al*. [18] provided a learnable module that can be integrated into most face verification models. This module generates meaningful explanations with the help of a patched cosine map and an attention map. These maps represent similarities instead of saliency.

Other model-specific XAI techniques require knowledge of the structure to observe or manipulate the outputs of hidden model layers: The most popular approach is the Gradient-weighted Class Activation Mapping (Grad-CAM) [26] algorithm that utilizes the gradient of the class signal with respect to the input image; Recently, many other XAI techniques based on GradCAM, like GradCAM++ [3], HiResCAM [7], AblationCAM [24], ScoreCAM [29], or XGradCAM [9], have been introduced; Cao *et al*. [2] modified a network with a feedback loop to infer the activations of hidden layers according to the corresponding targets; In [16] and [4], the authors trained separate models to predict saliency explanation maps; Pruning a neural network for a given single input to keep only neurons that highly contribute to the prediction was introduced in the work of Khakzar *et al*. [13].

2.2. Confidence Scores

In [12] Huber *et al*. exploited the approximation of model uncertainty through dropout and proposed an uncertainty score for the comparison of two images. Based on that, they additionally calculated a decision confidence to make the decisions for face verification more transparent without any training effort.

In contrast, Li *et al*. [17] propose a novel framework for face confidence learning in a spherical space. They extended the Mises Fisher density to it´s r-radius counterpart.

3. Method

3.1. Confidence Score

Nowadays, face verification systems [1, 5, 14, 19, 20, 32] make predictions based on the distance between two feature vectors. Those feature vectors are typically derived from a convolutional neural network $\mathscr{N}(\cdot)$, which extracts facial features $\mathscr{N}(\boldsymbol{I}) = \boldsymbol{f}$ from an aligned facial image $\boldsymbol{I} \in \mathbb{R}^{112\times112\times3}$. Most approaches utilize the cosine distance metric d for calculating the distance between two facial feature vectors $\boldsymbol{f}_1, \boldsymbol{f}_2$ which is defined as:

$$d(\boldsymbol{f}_1, \boldsymbol{f}_2) = 1 - \frac{\boldsymbol{f}_1 \cdot \boldsymbol{f}_2}{\|\boldsymbol{f}_1\|_2 \, \|\boldsymbol{f}_2\|_2}. \tag{1}$$

From this follows that $d \in [0, 2]$, whereas d is 0 for identical features, 1 for orthogonal vectors, and 2 for opposite vectors. To classify a pair of images as genuine ($d \leq t$) or imposter ($d > t$) one can then define a particular threshold t. For common face verification benchmark datasets (*e.g*., LFW [11], CALFW [31], CPLFW [30], SLLFW [6], XQLFW [15]), the threshold t is derived by applying 10-fold cross-validation on the test set. A certain threshold is found for each fold by maximizing the verification accuracy on the remaining folds. A prediction from a face verification with a distance d close to the threshold t can be interpreted

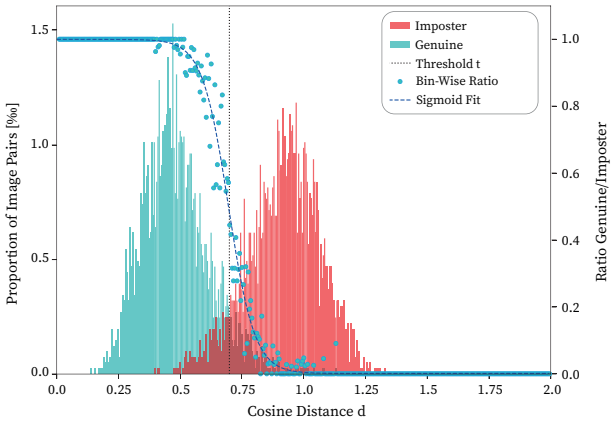

Figure 2. Histogram of cosine distances for the first fold of the LFW [11] dataset and the bin-wise ratio between genuine and imposter distance counts. The distances are derived from an ArcFace [5] model fine-tuned with OctupletLoss [14].

as uncertain. In contrast, a large distance close to 2 or a small distance close to 0 indicates high confidence in the model's prediction.

However, there is no clear rule on interpreting the absolute distance to the threshold t in terms of prediction confidence. For instance, the FaceTransformer [32] model in the work of Knoche *et al*. [14] has a threshold of $t \approx 0.2$, leading to highly imbalanced thresholds in the interval.

In this work, we aim for a more expressive metric and introduce a confidence score (C-Score) s, which takes not only this imbalance into account, but also exploits information from the distribution of correct and wrong predictions of the model for each dataset. Our C-Score is calculated as follows:

Given the cosine distance distribution derived from an arbitrary face verification model for genuine and imposter image pairs, we compute the histogram (*cf*. Figure 2) with 400 bins. The bin-wise ratio of the number of genuine examples to imposter examples follows an s-shaped distribution starting from 1 for $d < t$ and ending at 0 for $d > t$. The closer d gets to the threshold t, the more uncertain the prediction is due to more misclassifications in that range. We interpret the left part of the distribution ($d < t$) as the probability that a given genuine prediction is correct and the right part of the distribution ($d > t$) as the improbability that a given imposter prediction is correct. Then, we fit a logistic sigmoid curve $c(d)$ depending on the parameters L, d_0, k, b, defined as,

$$c = \frac{L}{(1 + e^{-k \cdot (d - d_0)})} + b \tag{2}$$

to the distribution of ratio values using the dogbox [28] algorithm. This enables a continuous mapping of arbitrary distance values. Because the fitted sigmoid curve c is an approximation, we clip the c to a range $[0, 1]$. Finally, to get a more intuitive score, we invert the improbability values and define our C-Score C as:

$$C = \begin{cases} c(d) & \forall \quad d \leq t \\ 1 - c(d) & \forall \quad d > t \end{cases} \tag{3}$$

As a result, we obtain our introduced C-Score C in the range of $[0.5, 1]$ for either genuine or imposter predictions and can interpret it as a probability for correctness. Notice that the calculation of C is done fold-wise, resulting in altering parameters for each fold of the dataset.

Finally, with C-Score C, we establish an additional value to the binary output prediction of a face verification system and thus make the prediction more meaningful. It is important to note that for the C-Score C, we gathered ground truth information of the dataset; hence, for the application of the model to field data, the parameters for the C-Score function need to be derived from a validation dataset.

3.2. Model-Agnostic Explanation Maps

The core principle of our model-agnostic explanation approach is visualizing the deviation between a non-occluded and an occluded image. If the feature distance between those two images is decreasing, we interpret the occluded area as dissimilar and vice versa similar for a greater distance. With systematic image occluding, we can then measure the influence on the cosine distance of every part of the image and hence, visualize this in a 2-D map.

In the following, the procedure is explained in more detail: 1) First, we apply our proposed Algorithm 1, which can be formulated as

$$\begin{aligned} \circ\mathrm{cc}(\boldsymbol{I}) \mapsto \quad & \mathcal{O} := \{\boldsymbol{O}_1, \boldsymbol{O}_2, \ldots, \boldsymbol{O}_\mathrm{N}\}, \\ & \mathcal{M} := \{\boldsymbol{M}_1, \boldsymbol{M}_2, \ldots, \boldsymbol{M}_\mathrm{N}\} \end{aligned} \tag{4}$$

with $\boldsymbol{O} \in \mathbb{R}^{112 \times 112 \times 3}$, $\boldsymbol{M} \in \mathbb{R}^{112 \times 112}$, and $N = \lfloor (112 - p)/s \rfloor^2$, dependent on the patch size p and a stride s. Note that our masks $\boldsymbol{M}$ are sparsely populated; only the occluded areas contain the values 1. After utilizing $\circ\mathrm{cc}(\cdot)$ on our input image 2-tuple $(\boldsymbol{I}_1, \boldsymbol{I}_2)$, we retrieve a 2-tuple of occluded image sets $(\mathcal{O}_1, \mathcal{O}_2)$ and a 2-tuple of mask sets $(\mathcal{M}_1, \mathcal{M}_2)$.

In the next step, we extract the facial features $\boldsymbol{f} \in \mathbb{R}^{512}$ with a face verification network $\mathscr{N}(\cdot)$ for every single occluded image $\boldsymbol{O}$ in the 2-tuple $(\mathcal{O}_1, \mathcal{O}_2)$:

$$\mathcal{F} := \{\boldsymbol{f}_1, \boldsymbol{f}_2, \ldots, \boldsymbol{f}_N : \boldsymbol{f} = \mathscr{N}(\boldsymbol{O})\}, \tag{5}$$

and consequently, generate a 2-tuple of feature vector sets $(\mathcal{F}_1, \mathcal{F}_2)$. Then, we calculate the cosine distance $d(\cdot, \cdot)$ between all features in $\mathcal{F}_1$ and $\mathcal{F}_2$. To select the 2-tuple of pair-wise distances sets $(\mathcal{D}_1, \mathcal{D}_2)$, we employ the following three methods:

Algorithm 1: Systematic Image Occluding $\texttt{occ}(\cdot)$

Input: image $\boldsymbol{I}$
$s \longleftarrow$ stride
$p \longleftarrow$ size of patch
Start at top left corner of $\boldsymbol{I}$
while *within* $\boldsymbol{I}$ **do** move right s pixels
  while *within* $\boldsymbol{I}$ **do** move down s pixels
    $\boldsymbol{M} \longleftarrow$ draw a patch with size p at loc
    $\boldsymbol{O} \longleftarrow$ occlude $\boldsymbol{I}$ with patch of size p at loc
  end
end
Output: occluded images $\mathcal{O}$, masks $\mathcal{M}$

Method 1 selects the cosine distances $\mathcal{D}_1$ and $\mathcal{D}_2$ according to

$$\begin{aligned}\mathcal{D}_1 &:= \left\{ \sum_{j=1}^{N} \frac{d(\mathcal{F}_1^{(i)}, \mathcal{F}_2^{(j)})}{N} : \forall\, i \in [1, 2, \ldots, N] \right\} \\ \mathcal{D}_2 &:= \left\{ \sum_{i=1}^{N} \frac{d(\mathcal{F}_1^{(i)}, \mathcal{F}_2^{(j)})}{N} : \forall\, j \in [1, 2, \ldots, N] \right\}.\end{aligned} \tag{6}$$

With this selection, we extract the averaged influence of all occluded (at any location) image for one of the input images compared with the occluded (at a particular location) image of the other input image.

Method 2 selects the cosine distances $\mathcal{D}_1$ and $\mathcal{D}_2$ according to

$$\begin{aligned}\mathcal{D}_1 &:= \big\{ d(\mathcal{F}_1^{(i)}, \mathscr{N}(\boldsymbol{I}_2)) : \forall\, i \in [1, 2, \ldots, N] \big\} \\ \mathcal{D}_2 &:= \big\{ d(\mathscr{N}(\boldsymbol{I}_1), \mathcal{F}_2^{(i)}) : \forall\, i \in [1, 2, \ldots, N] \big\}.\end{aligned} \tag{7}$$

This selection aims to measure the influence of one of the input images compared with the occluded (at any location) image of the other input image.

Method 3 selects the cosine distances $\mathcal{D}_1$ and $\mathcal{D}_2$ according to

$$\mathcal{D}_1 = \mathcal{D}_2 := \big\{ d(\mathcal{F}_1^{(i)}, \mathcal{F}_2^{(i)}) : \forall\, i \in [1, 2, \ldots, N] \big\}. \tag{8}$$

Here, we measure the distances between the co-located occludings of both input images.

Independent of the above-described methods, we obtain a 2-tuple of distance sets $(\mathcal{D}_1, \mathcal{D}_2)$, which is then compared with the original distance $d_{orig} = d(\boldsymbol{I}_1, \boldsymbol{I}_2)$ of both non-occluded input images. The difference in the distance in $\mathcal{D}_1$ or $\mathcal{D}_2$ compared with d_{orig} is the weight for it's corresponding occlusion mask in $\boldsymbol{M}$. After building the mean across all weighted masks, we generate similarity maps $\boldsymbol{S}$:

$$\boldsymbol{S} = \sum_{i=1}^{N} \frac{(d_i - d_{orig}) \cdot \boldsymbol{M}_\text{i}}{N} \tag{9}$$

with $d_i \in \mathcal{D}$ and $\boldsymbol{M}_\text{i} \in \mathcal{M}$. This allows visualizing the deviation caused by an occlusion at a particular location. The procedure described above is performed separately for every particular occlusion patch size $p \in \{7, 14, 28\}$ and a stride $s = 5$. The stride s reduces the number of images inferred by a factor of s^2. Consequently, we get three similarity maps $\boldsymbol{S}$ each for both input images $\boldsymbol{I}_1$ and $\boldsymbol{I}_2$. Finally, we calculate the weighted average of the similarity maps based on the size of the patch area:

$$\overline{\boldsymbol{S}} = \sum_{i=1}^{|p|} \frac{\boldsymbol{S}_i}{p_i^2 \cdot |p|} \tag{10}$$

The occurring raster artifacts, caused by using a stride instead of shifting the occlusion patch pixel by pixel, are compensated by applying a Gaussian-Blur to the mean similarity maps $\overline{\boldsymbol{S}}$ with an $s \times s$ kernel and $\sigma = s$, followed by normalization to the range $[-1, 1]$.

Ultimately, we generate a 2-tuple of X-Maps for a 2-tuple of input images $(\boldsymbol{I}_1, \boldsymbol{I}_2)$ via color blending $\texttt{blend}(\boldsymbol{I}, \overline{\boldsymbol{S}})$ (see Algorithm 2) with the corresponding $\overline{\boldsymbol{S}}_1$ and $\overline{\boldsymbol{S}}_2$.

Algorithm 2: Color Blending $\texttt{blend}(\cdot, \cdot)$

Input: image $\boldsymbol{I}$, similarity map *map*
$l \longleftarrow$ get luminance from: RGBtoHLS(img)
$h \longleftarrow$ get hue from: RGBtoHLS(map)
$s \longleftarrow$ get saturation from: RGBtoHSV(map)
$\boldsymbol{I}_b \longleftarrow$ HLStoRGB(h, l, s)
Output: blended image $\boldsymbol{I}_b$

The proposed approach generates an image-specific X-Map for both images of an arbitrary image pair. It highlights the similar and dissimilar regions of an image in terms of their identity features extracted from a face verification model.

4. Results

4.1. Qualitative Results

This section provides X-Maps for a small selection of image pairs from the LFW [11] dataset. With the release of our proposed *eXplainable Face Verification* platform, its very easy to browse through all the generated X-Maps for several models.

The X-Maps of the genuine pairs in Figure 3 are dominated by green-colored facial regions, for indicating similarity. In example a), the X-Map reveals that the eyes and mouth of the subject seem to not play an essential role in the model's decision. The cosine distance will get even smaller for occlusions on those parts of the face. In b), the nose is the only facial part, which is less critical for the model's

prediction. In the genuine example pair c), eyes, nose, and mouth are highlighted, and push the distance closer to zero.

Not surprisingly, the X-Maps of the imposter pairs indicate more dissimilar facial regions than the genuine pairs' X-Maps. Whereas the nose of the subjects in pair f) is marked very dissimilar, that is the case for the eyes of pair e). Interestingly, the nose of pair d) is specified as a very similar facial region.

All three X-Maps indicate that the forehead is rather similar compared to the more distinctive facial parts such as the eyes, nose, and mouth.

4.2. Comparison of Explanation Maps

In this section, we compare our three proposed methods (*cf*. Subsection 3.2) of generating the X-Maps. Whereas the X-Map-I (*cf*. Equation (6)) technique considers both images occluded, the X-Map-II (*cf*. Equation (7)) technique compares a non-occluded image with an occluded image. The main difference between those two methods and X-Maps-III (*cf*. Equation (8)) are the co-located occlusions in both images, resulting in identical X-Maps for both images. Therefore, X-Map-III method is most expressive for normalized and frontal facial images with co-located facial parts.

In Figure 4, we present the different X-Maps for three example image pairs. The X-Map-I and X-Map-II methods indicate different parts of the face as similar or dissimilar, which makes it difficult for a human to interpret the explanation. Nevertheless, this enables better explainability for misaligned, varying pose, or occluded images. The bottom row (Method-III) reveals that the eyes in example pair a) most strongly impact the prediction into the imposter direction. In c), the same holds for the nose and mouth region. Compared to a) where the eyes are clearly visible and of good quality, in c), they are of very bad quality and hence, are not considered playing an essential role in the verification prediction from our algorithm. In all images, the forehead region is highlighted rather as similar, which is obviously due to the lack of information.

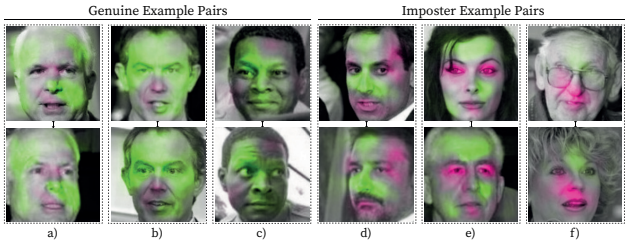

Figure 3. X-Maps for three genuine and three imposter example pairs from the LFW [11] dataset. Green colors indicate similar facial regions and red highlights dissimilar ones. All X-Maps are generated utilizing a FaceTransformer [32] model fine-tuned with OcutpletLoss [14].

4.3. Experiments with Cut-and-Paste Patches

Additionally, we conduct experiments with modified image pairs. This experiment investigates whether the replacement of particular facial regions in one image with a copy of the co-located region of the other image is successfully detected by our algorithm and described with high similarity in the X-Maps. Figure 5 depicts three facial replacements such as the eye region a), half side of the face b), or one eye and mouth c). In all three examples, our proposed algorithm highlights the copied facial regions as similar and the remaining facial area as dissimilar. In example c), this effect is most weakly pronounced.

4.4. Sensitivity Studies

We conduct multiple sensitivity studies to determine the influence of the size, edge quality, coloring, and shape of the patches used in our systematic image occlusion algorithm Algorithm 1.

First, we use three different patch sizes ($7 \times 7, 14 \times 14, 28 \times 28$ pixels) for our systematic image occlusion algorithm Algorithm 1 to visualize the effect of the occluding patches. As depicted in Figure 6 a), it strongly affects the resulting similarity maps. The smallest patch generates a more fine-grained similarity map and highlights small areas, which are not visible in the largest patch. In order to obtain one generalized X-Map (top right of Figure 6), with information from different levels of granularity, we merged three maps (all patches black colored, rectangular shaped and not Gaussian blurred) and weighted them based on the area of the patches (*cf*. Equation (10)).

Second, we analyze three different edge qualities in the patches, in the form of different levels of Gaussian blur-

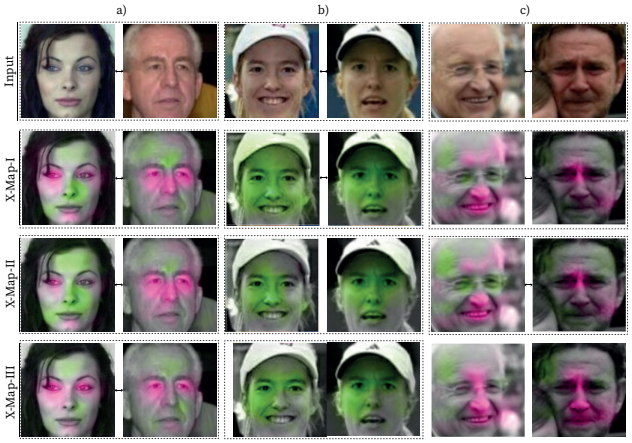

Figure 4. Comparison of our three proposed explanation maps algorithms for three example image pairs of the LFW [11] dataset. Green colors indicate similar facial regions and red highlights dissimilar ones. All X-Maps are generated utilizing a FaceTransformer [32] model fine-tuned with OcutpletLoss [14].

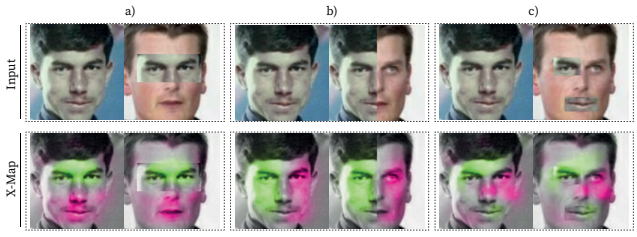

Figure 5. X-Maps (method-III) for three example image pairs of the LFW [11] dataset with modified facial regions. Green colors indicate similar facial regions and red highlights dissimilar ones. All X-Maps are generated utilizing a FaceTransformer [32] model fine-tuned with OcutpletLoss [14].

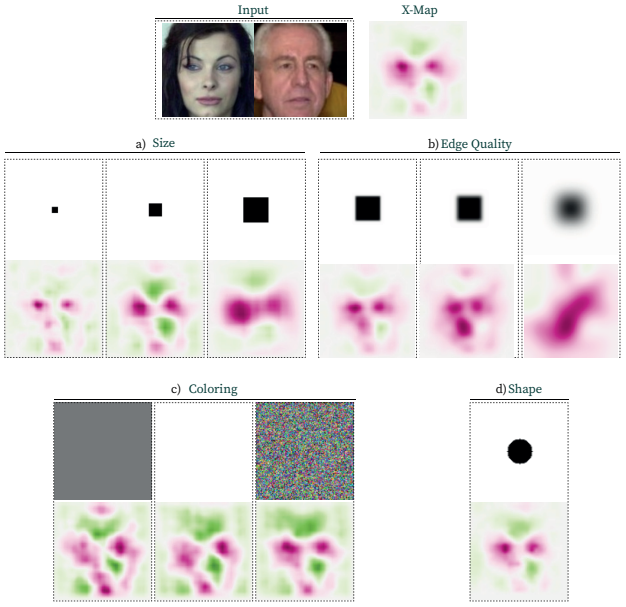

Figure 6. Similarity maps (method-III) for an example image pair of the LFW [11] dataset, generated with a different patch size, edge quality, coloring, and shape. Green colors indicate similar facial regions and red highlights dissimilar ones. All similarity maps are generated utilizing a FaceTransformer [32] model fine-tuned with OcutpletLoss [14].

ring. For both the kernel size and sigma, we use the values $\{7, 14, 56\}$. As seen in b), this affects the similarity maps regarding visual granularity.

Third, we vary the coloring of the patches and indicated in c) that it affects the similarity maps. The weakest similarity map is generated for black occlusions. The difference for gray, white, and noisy occlusion is only marginal.

Lastly, we investigate the effect of the shape of the patches. In d), we depict the X-Maps for rectangular and round shape. There are only minor differences visible in the similarity maps. We conclude that the shape of the patch has the most minor effect on our proposed X-Maps.

In summary, our sensitivity study reveals that the X-Maps content depends on the patch characteristics, and they should be adjusted carefully to the purpose and kind of data.

5. Web Platform

In this section, we will briefly describe our *eXplainable Face Verification* platform for presenting all the qualitative results of our approach and also help the community familiarizing with several test datasets and different model behaviors. This platform supports the visual understanding of the proposed algorithms. We also want our results to be easily accessible and publicly available.

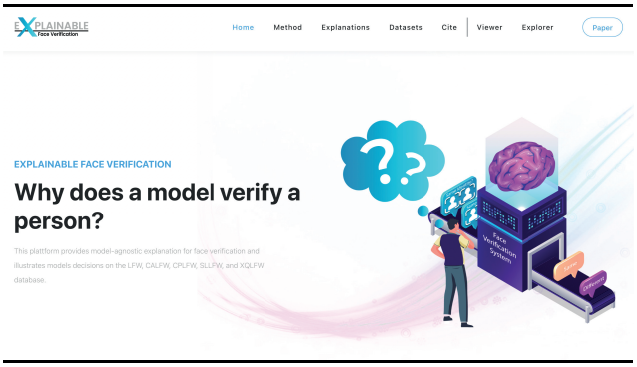

Figure 7. A screenshot of the landing page of our *eXplainable Face Verification* platform

The platform runs a flask [10] framework, which is connected to a database containing all the datasets and the metadata. The backend does all the sorting, filtering, and accessing to improve the user experience. The platform can be divided into three parts:

First, the landing page (see Figure 7) explains our method and gives an overview of the accessible data and a preview of the "viewer" functionality.

Second, the "explorer" page (see Figure 8) contains an interactive table where users can filter and sort the

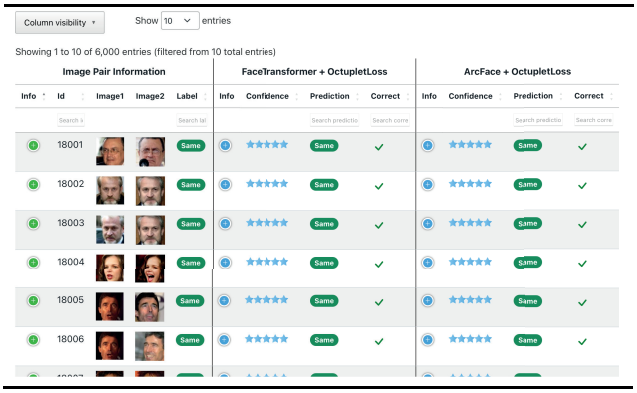

Figure 8. Screenshot of the "explorer" module of our proposed platform. It shows the interactive data table.

data. The image pairs and corresponding metadata, such as file path, label, identity, and image quality score for LFW [11], CALFW [31], CPLFW [30], SLLFW [6], and XQLFW [15], are stored in the table. Moreover, we added the results (*e.g.*, prediction, distance, threshold, confidence) from several face recognition models to the table.

Third, the purpose of the "viewer" page (see Figure 9) is to present our generated X-Maps in an interactive, adjustable way. The user can select different models, methods, and maps for each pair of images in the datasets.

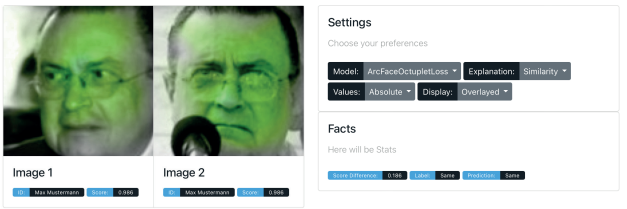

Figure 9. Screenshot of the "viewer" module of our proposed platform. It shows interactive X-Maps for an example image pair and corresponding metadata.

The limitations of the *eXplainable Face Verification* platform can be summarized as follows: 1) The datasets are limited to LFW and its derivatives. 2) We only applied our approach to face verification datasets. 3) The platform presents results for a small portion of existing face recognition networks.

6. Conclusion and Future Work

This work conducts further research on explainable face verification and proposes a novel strategy to generate three different explanation maps and a confidence score for a face verification model's prediction.

With our *eXplanable face verification* platform, we contribute a tool to further investigate the behavior of state-of-the-art face recognition networks and demonstrate the interpretability and accuracy of our approach.

However, our proposed X-Map algorithm can only highlight highly locally appearing similarities. Hence, our method cannot reveal more global similarities, such as skin color or the shape of the face.

Although our work focuses explicitly on faces, the approach is not limited to the faces domain and can potentially be applied to other binary decision problems.

In the future, we want to use the C-Scores and X-Maps for a joint application of human and machine face verification. We are planning to investigate whether a machine face verification algorithm can successfully, with the help of humans, solve the edge cases in face verification. We want to achieve this by filtering out the problematic cases based on the C-Score and using the X-Maps to support humans in their decision.

References

[1] Fadi Boutros, Naser Damer, Florian Kirchbuchner, and Arjan Kuijper. Elasticface: Elastic margin loss for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1578–1587, 2022.

[2] Chunshui Cao, Xianming Liu, Yi Yang, Yinan Yu, Jiang Wang, Zilei Wang, Yongzhen Huang, Liang Wang, Chang Huang, Wei Xu, et al. Look and think twice: Capturing top-down visual attention with feedback convolutional neural networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2956–2964, 2015.

[3] Aditya Chattopadhay, Anirban Sarkar, Prantik Howlader, and Vineeth N Balasubramanian. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In *2018 IEEE winter conference on applications of computer vision (WACV)*, pages 839–847. IEEE, 2018.

[4] Piotr Dabkowski and Yarin Gal. Real time image saliency for black box classifiers. *Advances in neural information processing systems*, 30, 2017.

[5] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4690–4699, 2019.

[6] Weihong Deng, Jiani Hu, Nanhai Zhang, Binghui Chen, and Jun Guo. Fine-grained face verification: Fglfw database, baselines, and human-dcmn partnership. *Pattern Recognition*, 66:63–73, 2017.

[7] Rachel Lea Draelos and Lawrence Carin. Hirescam: Faithful location representation in visual attention for explainable 3d medical image classification. *arXiv preprint arXiv:2011.08891*, 2020.

[8] Danilo Franco, Nicolò Navarin, Michele Donini, Davide Anguita, and Luca Oneto. Deep fair models for complex data: Graphs labeling and explainable face recognition. *Neurocomputing*, 470:318–334, 2022.

[9] Ruigang Fu, Qingyong Hu, Xiaohu Dong, Yulan Guo, Yinghui Gao, and Biao Li. Axiom-based grad-cam: Towards accurate visualization and explanation of cnns. *arXiv preprint arXiv:2008.02312*, 2020.

[10] Miguel Grinberg. *Flask web development: developing web applications with python*. " O'Reilly Media, Inc.", 2018.

[11] Gary B Huang, Marwan Mattar, Tamara Berg, and Eric Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. In *Workshop on Faces in 'Real-Life' Images: Detection, Alignment, and Recognition*, 2008.

[12] Marco Huber, Philipp Terhörst, Florian Kirchbuchner, Naser Damer, and Arjan Kuijper. Stating comparison score uncertainty and verification decision confidence towards transparent face recognition. *arXiv preprint arXiv:2210.10354*, 2022.

[13] Ashkan Khakzar, Soroosh Baselizadeh, Saurabh Khanduja, Christian Rupprecht, Seong Tae Kim, and Nassir Navab.

Improving feature attribution through input-specific network pruning. *arXiv preprint arXiv:1911.11081*, 2019.

[14] Martin Knoche, Mohamed Elkadeem, Stefan Hörmann, and Gerhard Rigoll. Octuplet loss: Make face recognition robust to image resolution. *arXiv preprint arXiv:2207.06726*, 2022.

[15] Martin Knoche, Stefan Hörmann, and Gerhard Rigoll. Cross-quality LFW: A database for analyzing cross-resolution image face recognition in unconstrained environments. In *16th IEEE International Conference on Automatic Face and Gesture Recognition (FG)*, pages 1–5, 2021.

[16] Kunpeng Li, Ziyan Wu, Kuan-Chuan Peng, Jan Ernst, and Yun Fu. Tell me where to look: Guided attention inference network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9215–9223, 2018.

[17] Shen Li, Jianqing Xu, Xiaqing Xu, Pengcheng Shen, Shaoxin Li, and Bryan Hooi. Spherical confidence learning for face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15629–15637, 2021.

[18] Yu-Sheng Lin, Zhe-Yu Liu, Yu-An Chen, Yu-Siang Wang, Ya-Liang Chang, and Winston H Hsu. xcos: An explainable cosine metric for face verification task. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 17(3s):1–16, 2021.

[19] Jiaheng Liu, Haoyu Qin, Yichao Wu, and Ding Liang. Anchorface: Boosting tar@far for practical face recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2022.

[20] Qiang Meng, Shichao Zhao, Zhida Huang, and Feng Zhou. Magface: A universal representation for face recognition and quality assessment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14225–14234, 2021.

[21] Domingo Mery. True black-box explanation in facial analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1596–1605, 2022.

[22] Domingo Mery and Bernardita Morris. On black-box explanation for face verification. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3418–3427, 2022.

[23] P Jonathon Phillips, Carina A Hahn, Peter C Fontana, David A Broniatowski, and Mark A Przybocki. Four principles of explainable artificial intelligence. *Gaithersburg, Maryland*, 2020.

[24] Harish Guruprasad Ramaswamy et al. Ablation-cam: Visual explanations for deep convolutional network via gradient-free localization. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 983–991, 2020.

[25] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016.

[26] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.

[27] Poornima Singh Thakur, Pritee Khanna, Tanuja Sheorey, and Aparajita Ojha. Explainable vision transformer enabled convolutional neural network for plant disease identification: Plantxvit. *arXiv preprint arXiv:2207.07919*, 2022.

[28] C Voglis and IE Lagaris. A rectangular trust region dogleg approach for unconstrained and bound constrained nonlinear optimization. In *WSEAS International Conference on Applied Mathematics*, volume 7, 2004.

[29] Haofan Wang, Zifan Wang, Mengnan Du, Fan Yang, Zijian Zhang, Sirui Ding, Piotr Mardziel, and Xia Hu. Score-cam: Score-weighted visual explanations for convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 24–25, 2020.

[30] Tianyue Zheng and Weihong Deng. Cross-pose lfw: A database for studying cross-pose face recognition in unconstrained environments. *Beijing University of Posts and Telecommunications, Tech. Rep*, 5, 2018.

[31] Tianyue Zheng, Weihong Deng, and Jiani Hu. Cross-age LFW: A database for studying cross-age face recognition in unconstrained environments. `arXiv:1708.08197`, 2017.

[32] Yaoyao Zhong and Weihong Deng. Face transformer for recognition. `arXiv:2103.14803`, 2021.