Supplementary Material: Self-supervised learning of a facial attribute embedding from video

Olivia Wiles*
ow@robots.ox.ac.uk
A. Sophia Koepke*
koepke@robots.ox.ac.uk
Andrew Zisserman
az@robots.ox.ac.uk

Visual Geometry Group
University of Oxford
Oxford, UK

We provide additional details about FAb-Net's architecture and training in Section 1, the self-supervised baselines in Section 2, pre-processing of the datasets in Section 3 and additional qualitative results in Section 4.

1 Additional details on architectures and training

Section 1.1 and Section 1.2 give additional details about the encoder-decoder architecture used and about the training respectively.

1.1 Architecture

The architecture of the encoders and decoders is based on pix2pix [12] but without skip-connections (see Fig. 1). It consists of encoders (with shared weights) and corresponding symmetrical decoders (also with shared weights). The face embedding vectors which are the outputs of the encoder have channel size 256. At the centre of the network, the source embedding vectors are concatenated pairwise with the target embedding vector. This 512-channel vector serves as input to the decoders.

The sampler decoders predict a $2 \times 256 \times 256$ output which determines how to sample from the source frame. When using multiple source frames, the network is augmented with confidence map decoders whose architecture is identical to the sampler decoders, except that the last layer outputs a 1-channel image. The confidence decoders also have shared weights for the different source frames.

1.2 Training and data augmentation

The images are first scaled to size 256×256. Because VoxCeleb1/VoxCeleb2 have different crops, VoxCeleb2 is re-cropped by padding the images by $[20, 80, 20, 30]$ pixels to the left, top, right, and bottom respectively and then taking a centre crop of size 190×190. Given the re-cropped VoxCeleb2 images and the VoxCeleb1 images, the images are augmented by

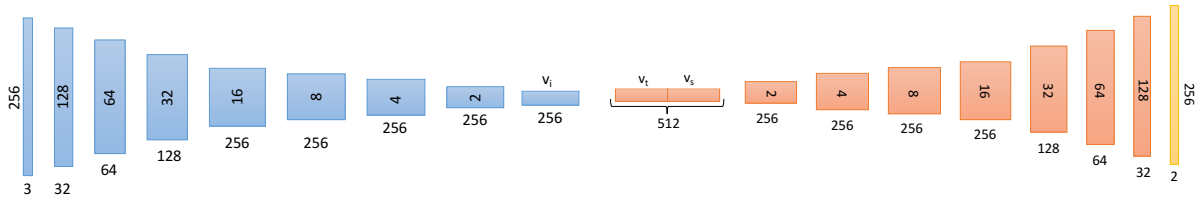

Figure 1: *Encoder-decoder*: In the encoder, each convolution is followed by a leaky ReLU (factor 0.2) and a batch-norm layer (except for the first which has no batch-norm); the convolutional filter sizes are 4×4 and the stride/padding is $2/1$. The face embedding vector has channel size 256. In the decoder, the face embedding vectors corresponding to the source/target frame (v_s, v_t) are concatenated giving a 512-dimensional embedding vector. In all decoder layers, the sequence of executions is: ReLU, bilinear upsampling, batch-norm. The convolutional filter sizes in the sampler decoder and the confidence decoder are 3×3, the stride/padding is $1/1$. The final $2 \times 256 \times 256$ result (or $1 \times 256 \times 256$ for the confidence decoders) is passed through a tanh layer to give FAb-Net's prediction for how to sample from the source frame.

taking random square crops with width in the range $[170, 190]$ for VoxCeleb2 and $[180, 200]$ for VoxCeleb1.

The models are trained using SGD with learning rate 0.001, momentum 0.9 and batch size $N = 8$ (unless the curriculum strategy for FAb-Net is used in which case $N = 32$). The learning rate is divided by a factor of 10 when the loss plateaus (unless the curriculum strategy for FAb-Net is used, in which case this occurs only after the curriculum strategy terminates).

2 Self-supervised baselines

FAb-Net is compared to an autoencoder and to two state-of-the-art self-supervised methods [3, 10]. FAb-Net's architecture but the appropriate loss functions and setups are used. More detail is given below. The baselines are trained with the same training hyperparameters and data augmentation as FAb-Net (please refer to Section 1.2).

Autoencoder. FAb-Net's encoder-decoder architecture are used. The autoencoder is trained to recreate the source frame via a 256-dimensional bottleneck vector, which is used later for evaluation. (The 256-dimensional vector output from the encoder serves as input to the decoder.)

Gidaris et al. [3]. Gidaris et al. apply a rotation $\theta \in \{0°, 90°, 180°, 270°\}$ to an image and train a CNN to predict the rotation that has been applied. To implement this baseline, FAb-Net's encoder (illustrated in Fig. 1) is used. A linear layer (with input channel size 256 and output channel size 4) is appended followed by a softmax layer. The network is trained with a cross-entropy loss. The 256-dimensional embedding is used for evaluation.

Zhang et al. [10]. Zhang et al. split an input image into L and ab channels (e.g. grey and colour channels) and then learn to reconstruct the ab channels from the L channel and the L channel from the ab channels. To do this, they effectively split the network into two smaller networks, each with half the capacity of the original network. To implement this baseline,

the exact same encoder-decoder structure as FAb-Net is used, except that it is divided into two subnetworks, each with half the capacity of the original network at each layer (e.g. 16 channels in the first layer and 128 channels in the embedding for each sub-network). The MSE loss is used as the reconstruction loss. The concatenation of the two 128-dimensional embeddings gives a 256-dimensional embedding, which is used for evaluation.

3 Datasets

For our models and baselines the input image size is 256×256 except for the VGG-Face descriptor which requires input images of size 224×224.

300-W and MAFL. FAb-Net is evaluated on 300-W and MAFL using the procedure outlined in [8]. In order to make the images more similar to those of VoxCeleb+, the images are re-cropped to make a tighter crop around the face region (this is a fair comparison as [8, 9, 11] re-crop to make the images more similar to those in CelebA [6] which they use as their training set).

AFLW. The images in AFLW [5] are resized to 256×256 for our models and to 224×224 for the evaluation of the VGG-Face descriptor.

EmotioNet. For EmotioNet [1], the classifier is trained on the subset of the dataset that is automatically annotated. We divide this subset into a training and a validation set (with a 80/20 split). The faces are detected using dlib [4] and cropped to 256×256. This results in $743,033$ images for training and $185,759$ images for validation. The independent set of about 25k images is used as test set. After detecting faces, this gives $25,517$ images for testing. This evaluation is done for the 11 AUs used in track 1 of the EmotioNet challenge 2017 [2].

AffectNet. For our experiments on AffectNet [7], we use the manually annotated subset of the dataset. As the AffectNet test set is not released, we use the released validation set to test on and randomly divide the training set into a training and a validation subset (with a 85/15 split). The faces are detected using dlib [4] and cropped to 256×256. Furthermore, images annotated as 'non-face', 'none' or 'uncertain' are discarded. This results in $287,055$ images for training, $57,411$ for validation and $3,989$ images for testing. The test set is balanced across the different emotion categories whereas the training data is not.

4 Additional qualitative results.

Additional examples of learned confidence heatmaps are visualised in Fig. 2 and additional examples for retrieval are visualised in Fig. 3.

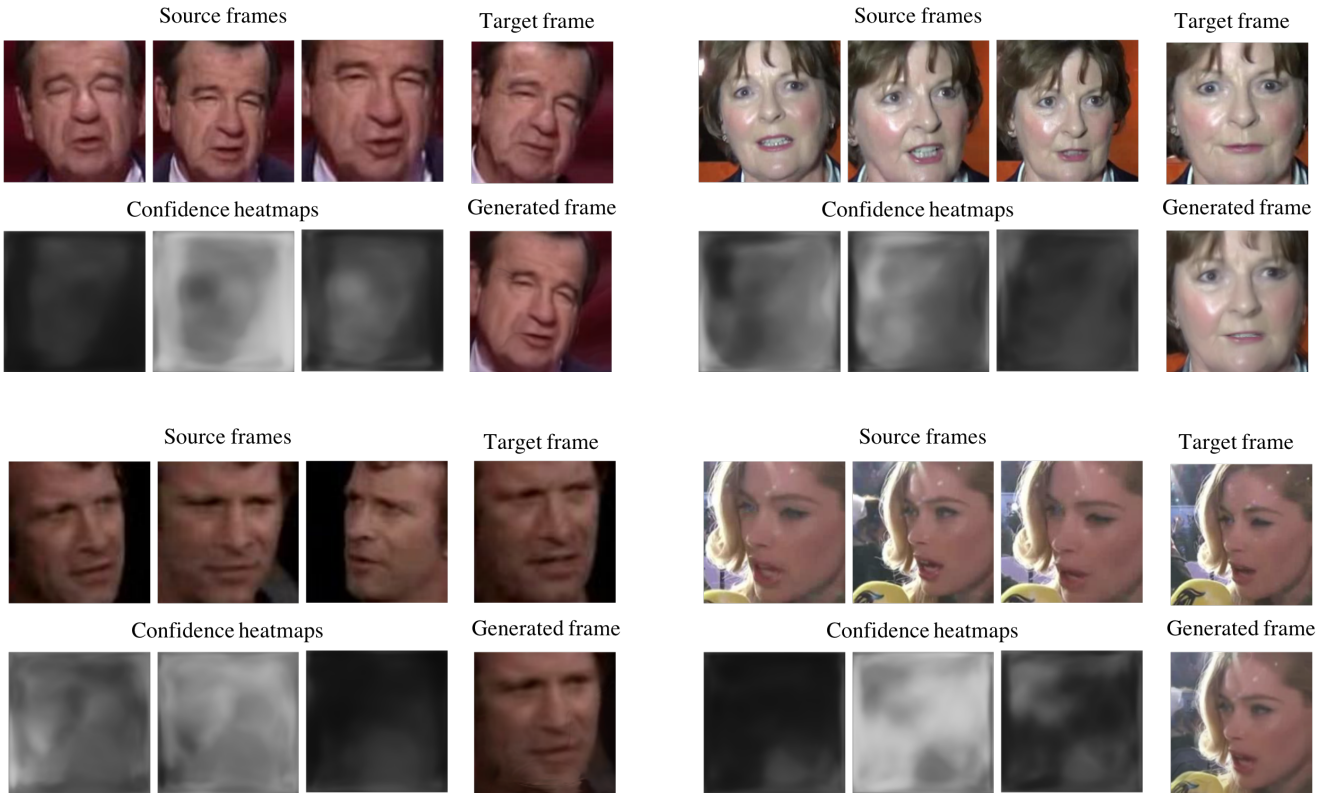

Figure 2: Additional examples of confidence heatmaps predicted by FAb-Net for the given source and target frames. These examples demonstrate how the confidence maps allow the network to focus on certain source frames or parts of different source frames. In the top left example, the network chooses one eye from the rightmost source frame even though its pose is quite different as compared to that of the target frame. In the bottom two examples, the source frames with pose or zoom dissimilar to that of the target frame are discarded.

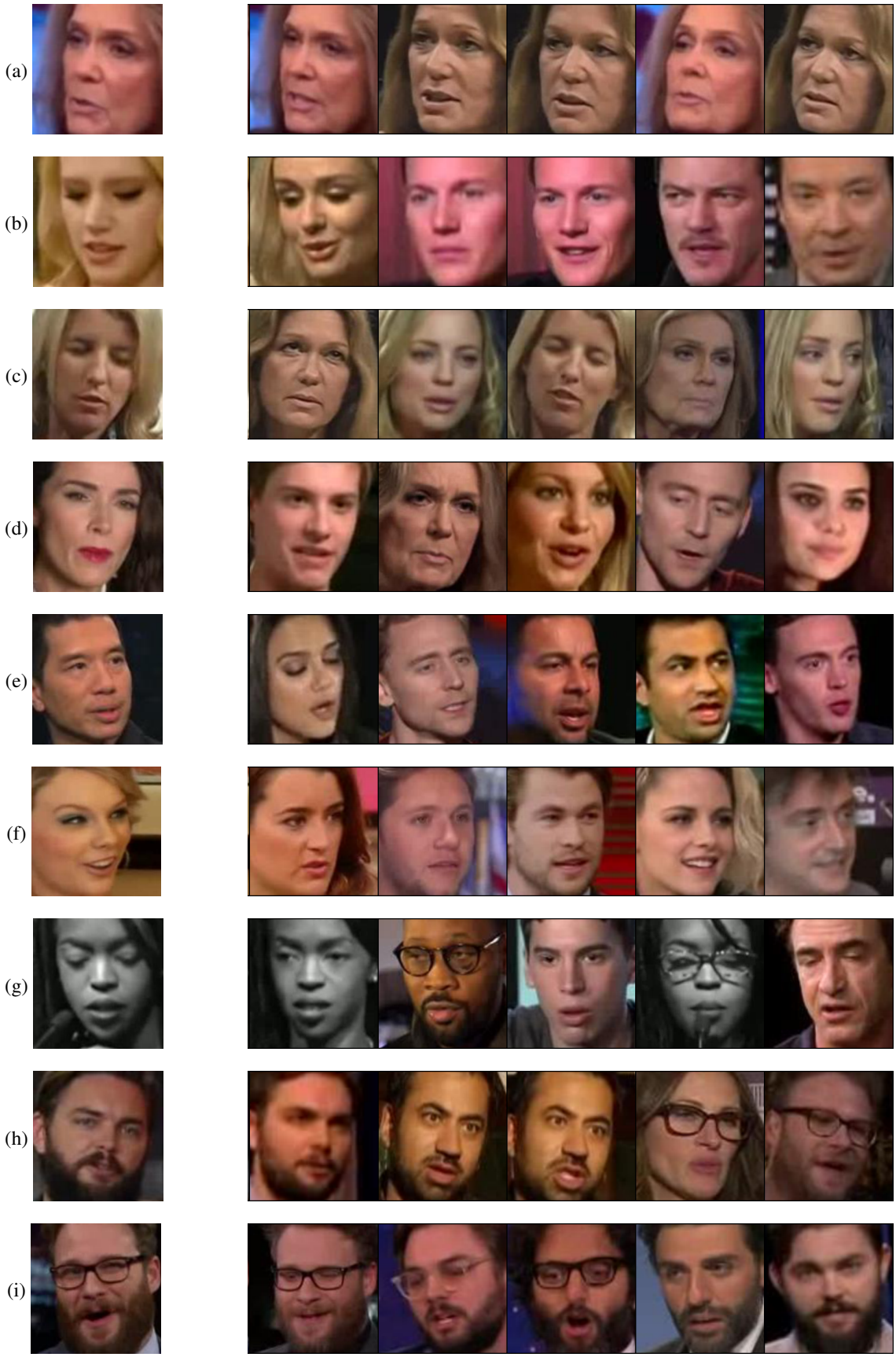

Figure 3: Additional results for nearest neighbour retrieval using the embedding learned by FAb-Net. The embedding captures facial attributes, e.g. pose and expression, as it succeeds in retrieving images with similar facial attributes given a query image.

References

[1] C. F. Benitez-Quiroz, R. Srinivasan, and A. M. Martinez. EmotioNet: An accurate, real-time algorithm for the automatic annotation of a million facial expressions in the wild. In *Proc. CVPR*, 2016.

[2] C. F. Benitez-Quiroz, R. Srinivasan, Q. Feng, Y. Wang, and A. M. Martinez. Emotionet challenge: Recognition of facial expressions of emotion in the wild. *arXiv preprint arXiv:1703.01210*, 2017.

[3] S. Gidaris, P. Singh, and N. Komodakis. Unsupervised representation learning by predicting image rotations. In *Proc. ICLR*, 2018.

[4] D. E. King. Dlib-ml: A machine learning toolkit. *The Journal of Machine Learning Research*, 10:1755–1758, 2009.

[5] M. Koestinger, P. Wohlhart, P. M. Roth, and H. Bischof. Annotated Facial Landmarks in the Wild: A Large-scale, Real-world Database for Facial Landmark Localization. In *Proc. First IEEE International Workshop on Benchmarking Facial Image Analysis Technologies*, 2011.

[6] Z. Liu, P. Luo, X. Wang, and X. Tang. Deep learning face attributes in the wild. In *Proc. ICCV*, 2015.

[7] A. Mollahosseini, B. Hasani, and M. H. Mahoor. AffectNet: A database for facial expression, valence, and arousal computing in the wild. *IEEE Transactions on Affective Computing*, 2017.

[8] J. Thewlis, H. Bilen, and A. Vedaldi. Unsupervised learning of object landmarks by factorized spatial embeddings. In *Proc. ICCV*, 2017.

[9] J. Thewlis, H. Bilen, and A. Vedaldi. Unsupervised learning of object frames by dense equivariant image labelling. In *NIPS*, 2017.

[10] R. Zhang, P. Isola, and A. A. Efros. Split-brain autoencoders: Unsupervised learning by cross-channel prediction. In *Proc. CVPR*, 2017.

[11] Y. Zhang, Y. Guo, Y. Jin, Y. Luo, Z. He, and H. Lee. Unsupervised discovery of object landmarks as structural representations. In *Proc. CVPR*, 2018.

[12] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. *Proc. ICCV*, 2017.