FACE ANONYMIZATION USING IMAGE OBFUSCATION

MANASVI PONAKA : 021120754E

CONTENTS

ABSTRACT

Recently, visual data is being generated at an unprecedented scale. People upload billions of photos daily on social media, and a high number of security cameras capture video data. As facial recognition technology has improved tremendously over the years and with the availability of huge visual data, facial data can be effortlessly collected and misused giving rise to innumerable privacy issues. Ironically such predicaments can be dealt with using our Facial recognition itself. It is from here that the field of Privacy-Preserving Deep Learning techniques arises. This project aims to implement a solution for our problem through facial obfuscation. Using State-of-the-art pre-trained Face detection model MTCNN along with a one-shot-learning pre-trained face recognition model[4] is used for classifying known and unknown faces in our reference data set. Various image obfuscation are implemented using OpenCV[10]. A Siamese neural network for face recognition is trained on a dataset constructed using OpenCV[10] and LFW image dataset. The results are compared to the State-of-the-art face recognition techniques. We focus on obscuring the facial data of an image without significantly affecting the information of other features of an image.

INTRODUCTION

With facial recognition technology at its best, many privacy concerns emerge. Through the uploaded of photos on social media, Image data sets used for Modelling and many other sources, a lot of unknown faces are being posted and published online. Privacy became a key concern of every individual. In this miraculous age of technology and abundance of data, It is very easy to exploit such sensitive data in wicked ways. So, using this research we take better precautions to avoid these uneventful incidents.

Despite this being an individual's privacy concern, with introduction of Strict laws restricting the usage and collection of private information, Visual data collection by various security cameras , autonomous vehicles, datasets published for academic usage and so on has become a serious topic of debate in recent times.

As mentioned above, the global problem of anonymous faces being posted online can be addressed and solved through image obfuscation to a great extent. With a set of labelled faces stored in our database, our task is to Anonymize unlabelled faces in the images to preserve privacy. The applied Image obfuscation techniques include blurring, pixelating , colour blocking. The existing models are designed to anonymize all the faces in the image.

We hypothesize to anonymize faces that are alien to our reference face dataset by following a series of machine learning techniques - Face detection , recognition and face masking filters.

When dealing with such real time problems with limited data , pre-trained state-of-the art-models potentially solve the issues. But Face recognition is such a field where the application plays an important role in the in choosing the trade-off between accurate predictions and data available. Using one shot learning pre-trained models , the prediction is probable to be inaccurate .We will see that in the following sections. To train a neural network with small data, a naïve approach is use data augmentation techniques to scale the cardinality of the dataset. In this research a Siamese neural network is developed based on the data collected using OpenCV[10] and LFW[5] dataset.

The potential contributions of this project are as follows.

- Automatic anonymization of unknown faces in an image, involving a series of tasks of face detection , recognition and obfuscation. These modified images are assumed to preserve most of the significant features of the image . This research can majorly contribute to the media ,entertainment and social media platforms.
- With one-shot-learning, pre-trained state-of-the-art face recognition model can classify if the detected faces in an image are familiar to the faces in the reference face dataset .This classification out-turn decides if a face should be anonymized or not. This is used a reference for our model's performance.
- Using a Siamese neural network ,We develop a program that can take an input image and generate an output image obfuscating all the faces that it is not trained on with a mask of user's choice.

- This research can set out to gauge further explorations on extending this techniques to videos and other face anonymizing problems.

With sufficient diversity in the images of the input faces , this model can become robust to different contrasts and saturations of images , beauty filters applied on faces and many more such scenarios.

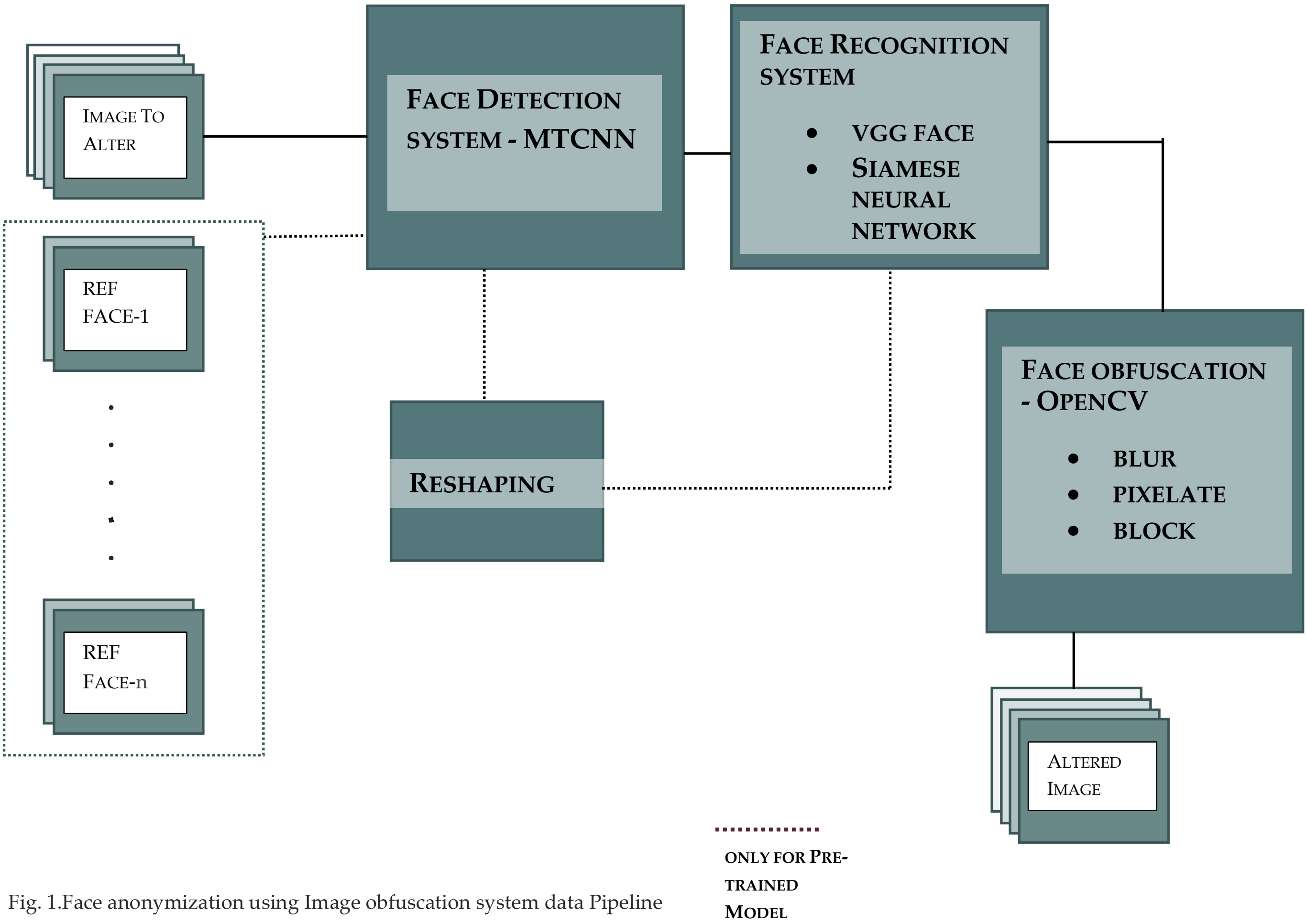

Fig. 1.Face anonymization using Image obfuscation system data Pipeline

RELATED WORK

The idea of Face anonymization is not novel yet the application at which this research is aimed at can be a pioneer. This project uses several existing techniques and a Siamese neural network for accurate face recognition .

Face detection and tracking has been a vigorous field of study in computer vision. Various methods have been proposed and followed since its origination. Some of them using Haar-Cascade classifier [17], some of them using convolutional neural networks[2].

The most recent advancement for an accurate face detection is introduced by Kaipeng,Z Et al. [2]. The proposed framework increases the depth for learning discriminating features better using 3 staged CNN networks to outperform state of the art approaches. A very accurate facial detection system is generated by changing the filter to 3*3, which gives more scope for learning features better in the CNN network and with each layer proceeding we get better at identifying the face location and key features. This architecture paved way to manifold applications such as in [1][3].

Research in the field of face recognition took many turns and tech giants such as Google, Facebook followed by several educational universities proposed state-of-the-art approaches. Facebook's DeepFace[16] uses a Deep neural net, and generated a very concise representation of a face that can also work for other datasets, rather than having thousands of appearance features, unlike the existed ones. The error is reduced from the then state of the art approaches by 27% closer to human perception in recognising faces.

This is followed by Google's FaceNet[6] and Oxford's VGG-Face[4] . FaceNet[6] uses a Siamese network to generate unified face Embeddings. The model is trained on Cross Entropy and back propagation of the error in similar spirit to our Siamese Neural network , which can be seen in the forthcoming sections.

VGG-Face[4] contributes in filtering out irrelevant details while analysing facial data. Deep CNN' s are used to achieve State-of-the-art performance without any Embellishments a triplet Embedding loss. In the following section the model is elaborated.

Coming to the final stage, Face anonymization ,many techniques have been proposed. One of the most advanced and complex approaches is using GAN's[18] for Face obfuscation[12][13][14]. Though the results are often prime , GAN's require high training data and computational resource to perform well.

A simpler obfuscation technique can be applied depending on the sensitivity of the application .Colour blocking is robust to neural network identification approaches[19].

MATERIALS AND METHODS

OVERVIEW

Aforementioned ,The flow of the project is divided into three phases .

1.Face detection

2.Face Recognition

3.Face anonymization

With state-of the-art models considered as benchmark,The project is designed using two approaches varying in Face Recognition stage:

- Pre-trained state-of the-art models
- Siamese neural network

FACE DETECTION

MTCNN - Multi-task Cascaded Convolutional Networks are used to face detection. The model is sourced from [A] , an open source library, based on [2] ,under MIT license. From a given input , it outputs bounding boxes for detected faces in an image along with the facial landmarks. Aforesaid the three stages in MTCNN use three neural networks and works like a pipeline. Stage One consist of a proposal network which will predicts potential face positions and their bounding boxes. A substantial number of faces are detected are made with plentiful probable false detections. Using the predictions of stage one , stage two- Refinement Network proceeds to refine and eliminate most of the false detections and aggregate bounding boxes. Stage three – Output Network adds facial landmarks predictions in the original MTCNN implementation[2]. Experimental results exhibit consistently that MTCNN's have the reliability of real-time performance and they consistently outperforms the sophisticated conventional methods across most of challenging benchmarks.

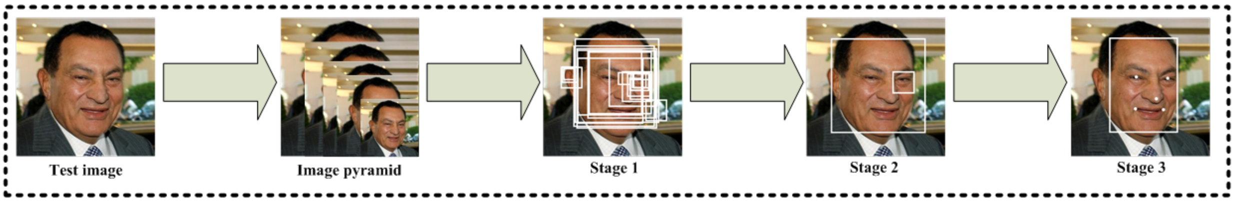

Fig. 2. MTCNN: pipeline of cascade framework used for face detection and landmark extraction.[2]
Stage 1:P-Net => Stage 2 : R-Net => Stage 3:O-Net [2]

FACE RECOGNITION

Arguably, the most influential stages in our research is Face recognition. This stage is followed by the Face detection stage , hence a list of faces detected in the image as passed through the face recognition model one by one depending on the approach we follow for Face recognition, our datasets differ.

Pretrained Model: VGG-Face2

DATA

The training data for this approach is a reference face dataset for one shot learning. The reference face dataset is constructed from crowd sourcing and seven participants aged under 30.provided the images and are used as labelled faces.

The images used for testing are crowd sourced too. The experiment is performed on ten images which consists of multiple faces in each image , familiar or unfamiliar to our reference face dataset.

Fig. 3. Some of the faces in reference face dataset

Fig. 4.Example of a testing image

PREPROCESSING

The images in the reference faces dataset are reshaped and processed to only include the face and dimension 105 x 105. These faces are sent through further processed to align the faces according the key points of a the face .Each image is sent through MTCNN network to find the keypoints of a face and align accordingly.

From the keypoints , the position of eyes is calculated and the if the line drawn between the centre of both is not parallel to the frame, the alignment is performed. The slope of the line is used to calculated the angle to align.

$$\theta = tan^{-1}(\frac{y_2-y_1}{(x_2-x1})$$

Angle from the gradient, m

A rotation matrix is designed and all the coordinates are translated using below formula. The argument for this transformation operation are the above obtained angle ,the centre of the image and each point in the image.

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{bmatrix} \begin{bmatrix} x - x_C \\ y - y_C \end{bmatrix} + \begin{bmatrix} x_C \\ y_C \end{bmatrix}$$

Translated coordinates using a Rotation matrix

METHODS

These aligned faces are passed through pretrained State-of-the-art models. DeepFace[B] is an opensource implementation of various pretrained Deep learning Models for face recognition. It is licenced under MIT.VGG-Face is our chosen pre-trained model.

The VGG-Face CNN descriptors are computed using a CNN implementation based on the VGG-Very-Deep-16 CNN architecture as described in [4] and is evaluated on the Labelled Faces in the Wild dataset [5] .The architecture comprise 37 layers.

The model is learnt on triplet loss of face embeddings similar to FaceNet[6] model from Facebook.

Ours (VGG Face)	2.6 M	1	98.78

Fig. 5. The accuracy on LFW dataset[4]

layer	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
type	input	conv	relu	conv	relu	mpool	conv	relu	conv	relu	mpool	conv	relu	conv	relu	conv	relu	mpool	conv
name	–	conv1_1	relu1_1	conv1_2	relu1_2	pool1	conv2_1	relu2_1	conv2_2	relu2_2	pool2	conv3_1	relu3_1	conv3_2	relu3_2	conv3_3	relu3_3	pool3	conv4_1
support	–	3	1	3	1	2	3	1	3	1	2	3	1	3	1	3	1	2	3
filt dim	–	3	–	64	–	–	64	–	128	–	–	128	–	256	–	256	–	–	256
num filts	–	64	–	64	–	–	128	–	128	–	–	256	–	256	–	256	–	–	512
stride	–	1	1	1	1	2	1	1	1	1	2	1	1	1	1	1	1	2	1
pad	–	1	0	1	0	0	1	0	1	0	0	1	0	1	0	1	0	0	1

layer	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37
type	relu	conv	relu	conv	relu	mpool	conv	relu	conv	relu	conv	relu	mpool	conv	relu	conv	relu	conv	softmx
name	relu4_1	conv4_2	relu4_2	conv4_3	relu4_3	pool4	conv5_1	relu5_1	conv5_2	relu5_2	conv5_3	relu5_3	pool5	fc6	relu6	fc7	relu7	fc8	prob
support	1	3	1	3	1	2	3	1	3	1	3	1	2	7	1	1	1	1	1
filt dim	–	512	–	512	–	–	512	–	512	–	512	–	–	512	–	4096	–	4096	–
num filts	–	512	–	512	–	–	512	–	512	–	512	–	–	4096	–	4096	–	2622	–
stride	1	1	1	1	1	2	1	1	1	1	1	1	2	1	1	1	1	1	1
pad	0	1	0	1	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0

Fig. 6. Architecture of VGG-Face Model Resulting in a 4096d descriptor before the softmax layer[4]

Siamese Neural Network

DATA

The input to the Siamese neural network is a pair of images. It is constructed from positive, anchor and negative data sets. Positive and anchor images dataset are collected using OpenCV[5]. With A web camera the one participants pictures are taken in light and dark settings of natural light. For the negative images dataset LFW dataset[5].Each image collected is 250 x 250 pixel

Number of images in each set:

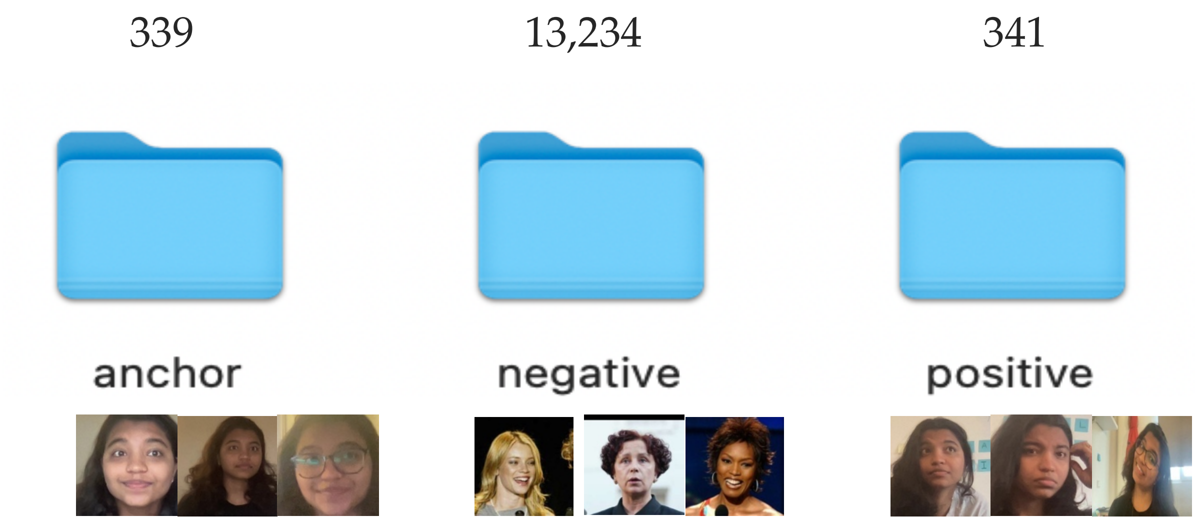

Fig. 7. Data Folders and Sample pictures from respective datasets

Positive and anchor images are of one female participant in early twenties. These images are captured with different variations such as – facial expression , face alignment , hair up and down , with accessories such as spectacles , different backgrounds.

PREPROCESSING

Data Augmentation

The anchor and the positive images are augmented using various image translation and transformation techniques. Each image is augmented to nine more images using a series of transformation such as change in brightness , contrast ,saturation , flipping.

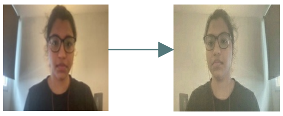

Fig. 8.An Example of the augmented image

The positive and negative images datasets are now augmented to sets of 3069 and 3051 images respectively.

Pre-process

3000 images are picked from each dataset a chosen randomly and siamese twins are created for positive and negative labels.

Positive_twin = (Anchor , Positive, 1)
Negative_twin = (Anchor , Negative , 0)

Each Image in the twin is pre-processed by resizing to 105 x 105 pixels and scaling each pixel between zero and one . The positive and negative twins data is concatenated and Shuffled for training.

TRAINING AND VALIDATION SETS

To evaluate the model's generalization capabilities , the shuffled data in the pre-process stage is divided into train and validation data .6000 tuples of image pairs and labels is divided into 4200 test and 1800 test images. Data is batched into batches of 16 and prefetched 8 for training.

METHOD

Our approach is to design a Siamese neural network to classify faces in an image. By construction it is a combination twin networks which accept distinct inputs but are joined by an energy function at the top and a classifier at its output. This unique structure of a Siamese neural network ensure great discriminative properties.

Twin network consist of two similar neural networks that generate Face Embeddings and the output layer is a distance layer followed a dense layer. The Face Embeddings are generated by a series of blocks of Convolution+ ReLU and Max Pooling Layers. Our standard model is a Siamese neural network with L-Fully connected layers and $h_{1,l}$ represents a hidden vector in layer l of first Embedding layer in the twin and $h_{2,l}$ is of the second Embedding layer in the twin network. A rectified linear (ReLU) is used in the first L-1 layers and hence $l \in \{ 1,\ldots..,L\text{-}1\}$

$$h_{1,m} = \max(0, W^{T}_{l-1,l}\, h_{1,(l-1)} + b_l)$$
$$h_{2,m} = \max(0, W^{T}_{l-1,l}\, h_{2,(l-1)} + b_l)$$

The shared N_{l-1} x N_l weight matrix $W_{l-1,l}$,connects N_{l-1} units in layer l-1 to N_l units in layer l and layer b_l is the standard bias vector for layer l [7].

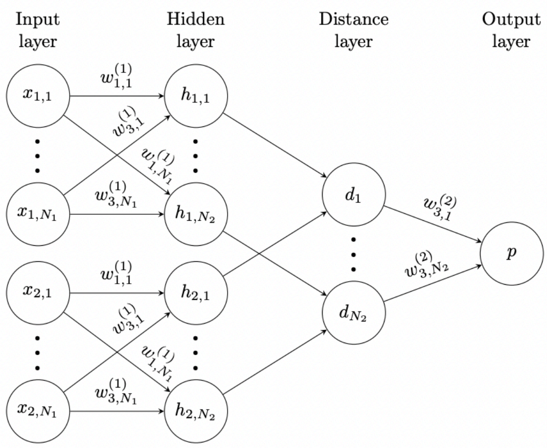

Fig. 9. A simple 2 hidden layer siamese network for binary classification with logistic prediction p. The structure of the network is replicated across the top and bottom sections to form twin networks, with shared weight matrices at each layer.[7]

Following the L-1 feed-forward layers , a distance measure is computed to compare the features obtained from the Embedding layers. sigmoid activation function is applied on L1- distance calculated between the embeddings generated by the twin networks. It induces a similarity score between the two feature vectors. The role of this fully connected layer is to join the Siamese twins.

$$p = \sigma \left(\sum_j \alpha_j \, | \mathrm{h}^{(j)}_{1,l} - \mathrm{h}^{(j)}_{2,l} | \right)$$

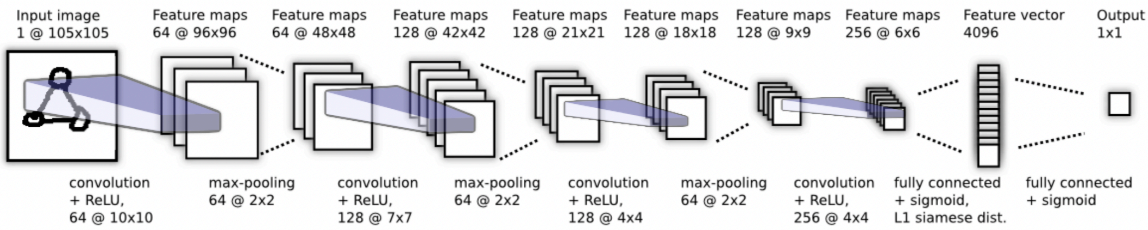

Fig. 10.Architecture of one Embedding layer followed by the fully connected layer (Siamese twin not depicted)[7]

The Convolutional layers in the Architecture are designed with varying filter size and a fixed stride of 2.A ReLu is applied on the output feature maps followed by max-pooling with filter size 2 and stride 2.

The k-th filter map in each layer takes the following form[7]

$$\mathrm{h}^{(k)}_{1,m} = \text{max-pool}(\max(0, \mathrm{W}^{(k)}_{l-1,l} \star \mathrm{h}_{1,(l-1)} + \mathrm{b}_l), 2)$$

$$\mathrm{h}^{(k)}_{2,m} = \text{max-pool}(\max(0, \mathrm{W}^{(k)}_{l-1,l} \star \mathrm{h}_{2,(l-1)} + \mathrm{b}_l), 2)$$

Where $\mathrm{W}_{l-1,l}$ is the 3-dimensional tensor representing the feature maps for layer l and $\star$ is a convolutional operator.

The Feature map from the last convolutional layer is flattened by a fully connected layer upon which a distance metric is induced and passed through the sigmoid output function.

LEARNING

Loss Function

Let M be the batch size and I is the index of the i-th batch. Now let $\mathbf{y}(x^{(i)}_1, x^{(i)}_2)$ be a length M vector containing the label 0 or 1 depending on the class of images in the Siamese twin x_1 , x_2 . Our binary classifier is imposed a regularised cross Entropy objective of the form:

$$L(x_1^{(i)}, x_2^{(i)}) = y(x_1^{(i)}, x_2^{(i)}) \log p(x_1^{(i)}, x_2^{(i)}) + (1 - y(x_1^{(i)}, x_2^{(i)})) \log (1 - p(x_1^{(i)}, x_2^{(i)})) + \boldsymbol{\lambda}^T |\mathbf{w}|^2 \quad \textbf{[7]}$$

Creating Siamese twins for training induces a contrastive behaviour in the loss. Distance between similar faces is reduced and dissimilar faces is increased.

Optimization

The above loss function is combined with an Adam optimizer to follow a standard back propagation algorithm. On a fixed batch size of 16 with an initial learning rate of 1e-4 ,our model updates weights using back propagation. In the optimizer the learning rate is updated adaptively ,to minimizing the loss function with each epoch.

Training

The model is trained on 4200 tuples of Siamese twins and their respective labels for 30 Epochs. Each Image is of resolution 105 x 105 pixels. The Face recognition model is implemented on Tensorflow[8] on 2 NVIDIA Tesla V100 SXM2 16GB GPUs on the UL HPC platform[9].

IMAGE OBFUSCATION

Each face sent through the face verification system is classified by the face recognizer and labelled. The bounding boxes of set of faces that do not get recognized are sent through the anonymization function to mask them and output the image anonymizing all unrecognized faces.

OpenCV[10] is an open source library used for image processing and computer vision tasks. Using a gaussian blur , colour blocking masks and pixelation techniques are used in the stage.

The blurring technique applied is using gaussian blur with a radius 20. A best fitting eclipse extraction of the face is done based on the bounding boxes obtained by the face detector. An over lay image defined with above mention gaussian blur filter and masked on to area defined by the eclipse which represents the face.

A pixelation algorithm is defined by implementing a down sampling technique with a filter of 10 x 10 pixels. The bounding boxes obtained are used to crop the face and pixelate it.

With a colour blocking mask ,the area covering the bounding boxes of the unrecognized faces is filled with a solid colour , in our case white.

Original image 1.Blurring. 2.Pixelation 3.Colur block mask

Fig. 11. An example obfuscation applied on faces.

RESULTS AND DISCUSSION

EVALUATION METRICS

By now it is evident that our research's performance is heavily relied on the face detector and face recogniser. The face detector MTCNN is bench marked on WIDER FACE[11] dataset with 95.4% accuracy. The Accuracy of the pre-trained face recognition model is trained on 2622 identities and 2.6M images .It is evaluated on the LFW data set with accuracy 98.95% for facial verification.

The Siamese Model ,the probabilistic prediction score from the dense layer is divided into two classes based on a Threshold of 0.5. If the prediction score is above 0.5 , its rounded to 1 else 0. This prediction is used to evaluate the model performance and we developed an accuracy of 80.5% accuracy on the 1800 testing samples in the test data. The Face recognition network had a precision and recall of 1 on the testing data.

The loss , precision and recall are plotted for each epoch during training.

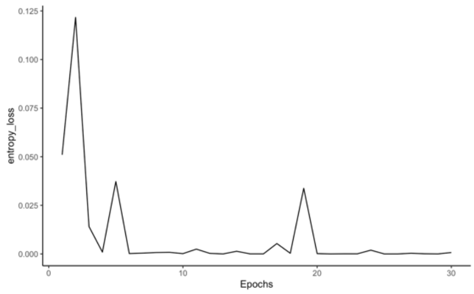

Fig 12 : entropy loss vs Epoch

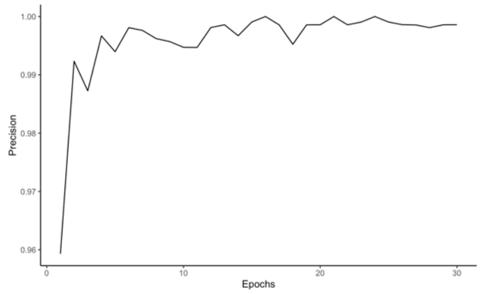

Fig 13 : Precision vs Epoch

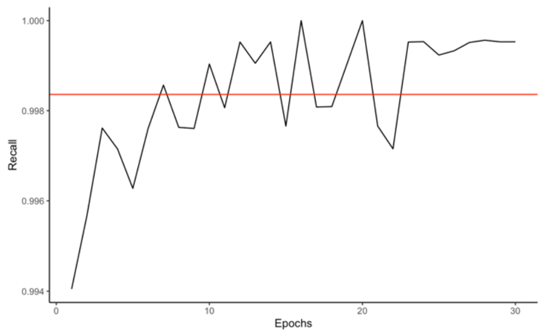

Fig 14 : Recall vs Epoch

A sample of the output from the Face anonymization model based on VGG-Face face recognizer can be found below.

Ref faces Images :

Fig. 15. output images with Anonymization : Pixelation

Fig. 16. output images with Anonymization : Blur

Fig. 16. output images with Anonymization : Colour Blocking mask

COMPARING RESULTS

As the Siamese network is trained on only one face we have only one face in reference image dataset. The comparison of the output images of the models based on pre-trained face recognition model VGG-Face and the Siamese Neural network are shown below.


Figure 17: Reference face and test Image

Fig. 18. VGG-Face model output images with Anonymization : Pixelation (3 faces are recognised and thus not blurred)

Fig. 19. Siamese Neural network output images with anonymization : Blurring

Fig. 19. Siamese Neural network output images with anonymization : Pixelation

DISCUSSON

In Future work we can address many limitations of the model and explore various other techniques for training the model . One of them can be introducing a contrastive loss function along with a Euclidean distance(L2) layer. The number of hidden in the model can be modified and regularization methods such as dropout can be introduced. The Siamese model for face recognition is trained only on one participant making it hard to generalise.

Choosing Evaluation metric such as ROC-AUC , Precision-Recall AUC for the face recognition model serves the purpose well rather than accuracy , precision and recall.

With more time and resource , data collection and training can be performed better, thus making the model more robust to diverse faces. This model can not only be used for anonymizing faces but can be extended to various domains where security is a key concern.

CONCLUSION :

We have presented a model to anonymize unfamiliar faces in an image. Many variations in anonymization has been presented and colour blocking is one of the safest filters when it comes to reproducing the anonymized faces[15].GAN's are extensively being used for such techniques but time and data and computing resources required for training such networks is a huge drawback .In real time applications prediction with such complex networks can be bothersome.

Coming to our two approaches, though the performance metric, accuracy is high for the VGG-Face pretrained model, our Siamese neural network performed better on the test images. Data collection and dataset curation for Siamese model is can be tedious but the results are favourable.

In future work, with more participants and diversity in aspects such as age ,gender and race adds more identities to the anchor and positive images data for better generalization of the face embeddings creation. The weights of this embedding layer can be used for a state-of-the art approach. With additional resources and time, the hyper parameters for the network are be tuned for better performance.

REFERENCES

[1] Ma, M.and Wang, J. (2018), *Multi-View Face Detection and Landmark Localization Based on MTCNN,* 2018 Chinese Automation Congress (CAC), Xi'an, China, pp. 4200-4205. doi: 10.1109/CAC.2018.8623535.

[2] Kaipeng, Z., Zhanpeng, Z., Zhifeng L. (2016), *Joint Face Detection and Alignment using Multi-task Cascaded Convolutional Networks,* [http://arxiv.org/abs/1604.02878 arXiv:1604.02878].

[3] Meng, Y., Seyed, Y. N et al (2020): *Minor Privacy Protection Through Real-time Video Processing at the Edge*; [https://arxiv.org/abs/2005.01178].

[4] Parkhi, O. M., Andrea, V., Zisserman, A. (2015), *A.Deep Face Recognition.*

[5] Huang, G. B., Ramesh, M., Berg, T., Learned-Miller, E. (2007), *Labeled faces in the wild: A database for studying face recognition in unconstrained environments.*

.

[6] Schroff, D. K., Philbin, J (2015). *Facenet: A unified embedding for face recognition and clustering*.

[7] Koch, R. G. et al (2015). *Siamese Neural Networks for One-Shot Image Recognition*

[8] Abadi, M. et al. (2016), *TensorFlow: A System for Large-Scale Machine Learning.,12th USENIX symposium on operating systems design and implementation (OSDI 16)*.

[9] Varrette, S., Bouvry, P., Cartiaux, H., Georgatos, F. (2014), *Management of an academic HPC cluster: The UL experience,* International Conference on High Performance Computing & Simulation (HPCS).

[10] Bradski, G., Adrian, K (2018), *Learning OpenCV: Computer vision with the OpenCV library*.

[11] Yang, S, et al. (2016), *Wider face: A face detection benchmark." Proceedings of the IEEE conference on computer vision and pattern recognition*.

[12] Tao, L., Lei, L. (2019), *AnonymousNet: Natural Face De-Identification With Measurable Privacy, IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*

[13] Yang, K.,Yau, J., Fei-Fei, L., et al (2021), *A Study of Face Obfuscation in ImageNet,* [https://arxiv.org/abs/2103.06191].

[14] Hukkelås, Håkon, Rudolf Mester, and Frank Lindseth. "*Deepprivacy: A generative adversarial network for face anonymization." International symposium on visual computing. Springer, Cham, 2019.*

[15] Sun, Qianru, et al. "*A hybrid model for identity obfuscation by face replacement." Proceedings of the European conference on computer vision (ECCV)*. 2018.

[16] Taigman, Yaniv, et al. "*DeepFace: Closing the Gap to Human-Level Performance in Face Verification." Proceedings of the IEEE conference on computer vision and pattern recognition. 2014.*

[17] Padilla, Rafael, C. F. F. Costa Filho, and M. G. F. Costa. "*Evaluation of haar cascade classifiers designed for face detection." World Academy of Science, Engineering and Technology 64 (2012): 362-365.*

[18] Goodfellow, Ian, et al. "*Generative adversarial nets." Advances in neural information processing systems 27 (2014).*
[19] Ahonen, Timo, et al. "*Recognition of blurred faces using local phase quantization." 2008 19th international conference on pattern recognition. IEEE, 2008.*

[AMTCNN : HTTPS://GITHUB.COM/IPAZC/MTCNN

[B] DEEP FACE : HTTPS://GITHUB.COM/SERENGIL/DEEPFACE