IS470 TERM REPORT
IDENTITY OBFUSCATION IN PERSON IMAGES

Student: Muhammad Naufal (ID: 01319703)
Supervisor: Prof. Sun Qianru

ABSTRACT

The increasing usage of social media means that everyday, around 1.8 billion photos are uploaded onto the internet [7]. With varying degrees of protection from each social media platform, the user is faced with varying degrees of threat to their privacy. They would also have no means of knowing if their privacy has been compromised, as there is no efficient method to trace every photo on the internet [6]. As such, there represents a need for AI systems to obfuscate images where private information is shown. Private information may range from face images, location images, as well as artifact images. However as majority of private information lies in our face images, that would be the main information that we would not want to compromise [1]. Once compromised, we are vulnerable to attacks by bad actors, such as scammers [20]. In order to prevent such an outcome, many methods have been introduced to obfuscate face images [28; 15]. They range from blurring faces, swirling, black-out, in-painting as well as masking [24]. This is still a much worked on topic in computer vision as many novel techniques have been introduced recently [27]. They range from using face in-painting as well as face de-identification [25; 15]. We propose a 3-stage framework for face image obfuscation, that leverages on using attribute manipulation, face landmarks and image context to generate new face images for purposes of obfuscation. Our experiments show that the proposed obfuscation method is effective against a machine person-recognizer whilst maintaining an acceptable level of human perception.

1 INTRODUCTION

1.1 PROBLEM STATEMENT

The rise of social media and information sharing has lead to increasing concerns and questions on privacy. Combined with advancements in the field of machine learning, this has lead to many privacy questions being unanswered. Armed with web-scapers at scale, and state-of-the-art person recognizers, companies such as 'Clearview AI' are able to identify a person with just a single face image [10]. Although such companies have claimed that the usage is mainly used to track down criminals, there is no guarantee that the technology would not fall into the hands of bad actors. In order to combat this, social media platforms allows users some degree of privacy protection by allowing users to set preferences for visibility of their photos. This can be done by setting filters to their pictures to blur faces or by purposely hiding the images from certain audiences [16]. However this approach has become increasingly ineffective due to the recent advances in machine learning to enable person recognition [21].

1.2 OBJECTIVES AND GOALS OF RESEARCH

We aim to explore new methods of obfuscation in person images. Particularly those related to conditional image generation. In our case, we want to explore the impact of face image generation conditional on face attributes, face landmarks and image context. Disentangling the image into these three factors will be done in stage-1.

We also aim to improve on the quality of face image generation, as well as to explore novel techniques on the conditional image generation process. Image reconstruction from the three factors will be done in stage-2.

Finally we aim to answer the question, would face attribute manipulation help in generating better quality face images for the purposes of identity obfuscation? The image generation task will be done in stage-3.

1.3 RELATED WORKS

1.3.1 SEMI SUPERVISED IMAGE RECONSTRUCTION

Our work in image reconstruction is closely related to that of [18] where the authors proposed a novel technique of person image generation via sampling disentangled factors of a person image. While the focus of the research was on the task of person re-identification via an entirely self-supervised pipeline, certain elements of their pipeline have been adapted to achieve our objectives. They disentangled the person images into pose, background and foreground, whereas we disentangle the face images into face landmarks, image context and face attributes. Hence the difference is that due to the self-supervise nature of their method, they were not able to extract image attributes. We do this by training a multi-label face attribute classifier (FAC), that can extract face attributes at test time. Further, as they intended to manipulate all three factors, they trained separate mapping functions for each factor, in comparison, our approach does not require a mapping function for each factor as the aim is to only manipulate face attributes, keeping the other two factors fixed.

1.3.2 CONDITIONAL IMAGE GENERATION ON FACE ATTRIBUTES

In [15] the authors aimed to achieve a higher level understanding of face image obfuscation, such that they were able to ask specific questions surrounding the privacy of an image. They introduced the Privacy-Preserving Attribute Selection (PPAS) algorithm to select and update face attributes, followed by image generation via style transfer. While they propose using attribute manipulation as an initial method before image generation, they do not take into account the obfuscative properties of the attributes. We take into account the properties of each attribute in the hope that we will be able to reconstruct higher quality images, rather than choosing the attributes to be manipulated at random. Further for the FAC task, the authors proposed using 40 different random forest classifiers whereas we propose using just a single multi-label classifier to extract face attributes. As mentioned in [19], such methods have higher cost in terms of run-time.

1.3.3 MULTI-LABEL FACE ATTRIBUTE CLASSIFICATION

The authors in [19] introduce a novel method for FAC, where they incorporate multi-task learning and multi-label classification for the FAC task. In order to classify images they split the learning task into two stages, one with shared features and one with task-specific features. In comparison to our approach we do not adopt multi-task as the objective of our research is to disentangle the face attributes from the face landmarks, hence we opt for two distinct tasks instead, landmark detection and FAC. Further the authors proposed grouping attributes into 2 classes, subjective and objective. Our approach makes no distinction between attributes as we adopt a multi-label approach that treats each label as distinct, hence we do not take into account relations between attributes. Relations between attributes will only be accounted for at stage-2, the image reconstruction task.

1.4 RESEARCH MOTIVATION

Many methods have been introduced for effective image obfuscation as many novel techniques have been introduced recently [27]. The problem is also general in the sense that there does not appear to be a one-size fits all solution. There have been successful attempts at face swapping to obfuscate identities [2], however at the same time there have been successful attempts at detecting swapped faces [4]. Hence, there is a need for a variety of high-quality methods for obfuscation. There is also a difference between target generic and target specific obfuscation methods [25]. Ours aims to be a target generic method such that we do not discriminate between different machine recognizer. On top of this, the perception of the image is also of importance. We want the image to be of a decent enough quality such that it can obfuscate against both human and machine recognizer.

2 METHODOLOGY

2.1 OVERVIEW OF ALL 3-STAGES

Our goal is to generate realistic face images by manipulating face attributes. In order to manipulate the face attributes, we need to disentangle the face attributes from the face image first. This will then allow us to manipulate existing face attributes. In order to generate face images from a disentangled representation, we train an image reconstruction network that can reconstruct face images from these disentangled factors. We do this via a 3 stage framework, where training and testing is done in stage-1 and stage-2, and only testing is done in stage-3. In stage-1, as shown in Figure 1, we disentangle the face image into three parts, the face outline via landmark detection [11], the image context via face-masking and the face attributes via a face attribute classifier (FAC) with a ResNet-50 architecture [9]. In stage-2, we will focus on the task of image reconstruction from the three parts mentioned in stage-1. We do this in a self-supervised manner, via a auto-encoder with a 'U-Net' style architecture [22]. Before passing the three parts into the decoder, we upsample the 40 face attributes in order to fit the dimensions of our two other factors. In the final stage, using the trained decoder in stage-2, we will generate new faces by manipulating the face attribute combination of the input image. Stage-3 is our testing stage as want to see if the images generated would perform well against the state-of-the-art machine and human recognizer.

2.2 STAGE 1

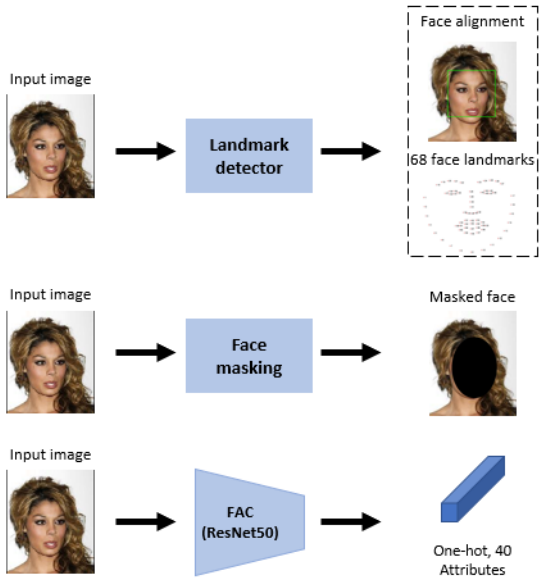


Figure 1: Overview of stage 1, landmark detection, face masking, FAC

2.2.1 LANDMARK DETECTION

For the landmark detection task, we follow the methods introduced in [23] using a pre-trained model available via dlib toolkit [12]. The model is able to track the 68 face landmarks in the form of *(x,y)* coordinates. These 68 points will be used in the next step which is face masking. The landmarks will allow us to align faces when reconstructing images in stage-2.

2.2.2 FACE MASKING

For face masking we use the ellipse method provided by openCV [3]. Using the face landmarks detected earlier, we will draw a black ellipse over the face region, such that only the image context is left. The parameters of the ellipse are set based on landmarks such as the forehead, chin, cheek and nose.

2.2.3 FACE ATTRIBUTE ESTIMATION

For face attribute classification, we adopt the ResNet-50 architecture as mentioned in [19]. We remove the final output layer and add a flatten layer before adding a final dense layer as our output layer. The final dense layer will have 40 units, corresponding to the 40 attributes we want to classify. For each unit in the output layer, we use *sigmoid* as the activation function. This will allow us to have 40 independent probabilities, each corresponding to one attribute. The solver we use is *Adam* with default settings [13]. For the loss function, we adopt the binary cross-entropy loss as in [8]. The loss function for each sample and each class is formulated as such:

$$L_{FAC}^{j} = -\sum_{i=1}^{2} t_i log(s_i) = -t_1 log(s_1) - (1 - t_1) log(1 - s_1) \tag{1}$$

Where *j* corresponds to the attribute or class and *i* corresponds to the label of ground truth. As our output is binary, t_1 is either 1 or 0, and we have substituted $t_2 = 1 - t_1$ in (1). s_1 is the output of our sigmoid activation, where $0 < s_1 < 1$, and we have substituted $s_2 = 1 - s_1$ in (1). We then extend this loss function to formulate the total loss function at each pass through the network as such:

$$L_{FAC} = \sum_{j=1}^{40} L_{FAC}^{j} \tag{2}$$

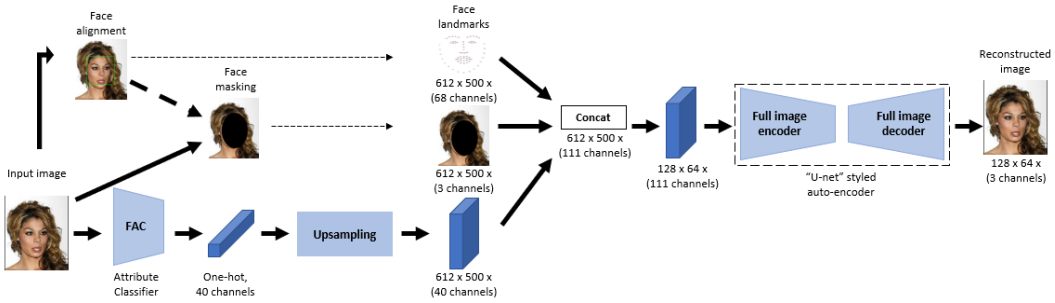


Figure 2: Overview of stage 2, image reconstruction

2.3 STAGE 2

2.3.1 IMAGE RECONSTRUCTION

For image reconstruction, we will adopt the method found in [18] where a self-supervised method was used to reconstruct person images for the task of re-identification, that is to augment images to create more images for model training purposes. Figure 2 gives the overview of the intended pipeline. Using the output from stage-1 and with further processing on the output of our FAC, we will have three different factors to concatenate before passing it through our auto-encoder. The three factors are the face landmarks, the face mask, as well as 40 face attributes. As in [26], by masking the face region, we are forcing the reconstruction network to predict the semantic parameter's using only the the context information and the face attributes, thus reducing the extent of facial identity information captured in the reconstructions. For the 40 face attributes, instead of using a one-hot encoding for each image, in relation to which labels are present and which are not, we upsample the one-hot vector to a 2D-array so that we are able to perform the concatenation operation. For the upsampling operation, we create an array of 1's if the attribute is present and an array of 0's otherwise. The order of attributes is kept consistent across all images. This is then followed by concatenating the three different tensors before passing the resultant tensor into the auto-encoder.

We will then begin the image reconstruction process. An overview of the reconstruction network is shown in Figure 3. The network is made up of blocks of convolutional layers. The base block 'ConvBlock2d' is made up of one convolutional 2D layer followed by a batch normalization layer and finally a rectified linear unit (ReLu) activation which is similar to the architecture proposed in the original 'u-net' paper [22]. The stack encoder is made up of two of these initial blocks followed by a max-pooling operation. The stack decoder is made up of three of these initial blocks followed by an up-sampling operation. The size of the blocks for the encoder and decoder is adopted from the decoder architecture found in [18]. For the loss function we will use the structural similarity index measure (SSIM) loss which is just $1 - SSIM$ to optimize the face image [29]. As mentioned by the authors in [5], using a non-perceptual loss such as the squared euclidean distance between the original image and the reconstructed image would result in blurry images, as the loss in the image space leads to an averaging of all likely locations. Therefore a perceptual loss is preferred.

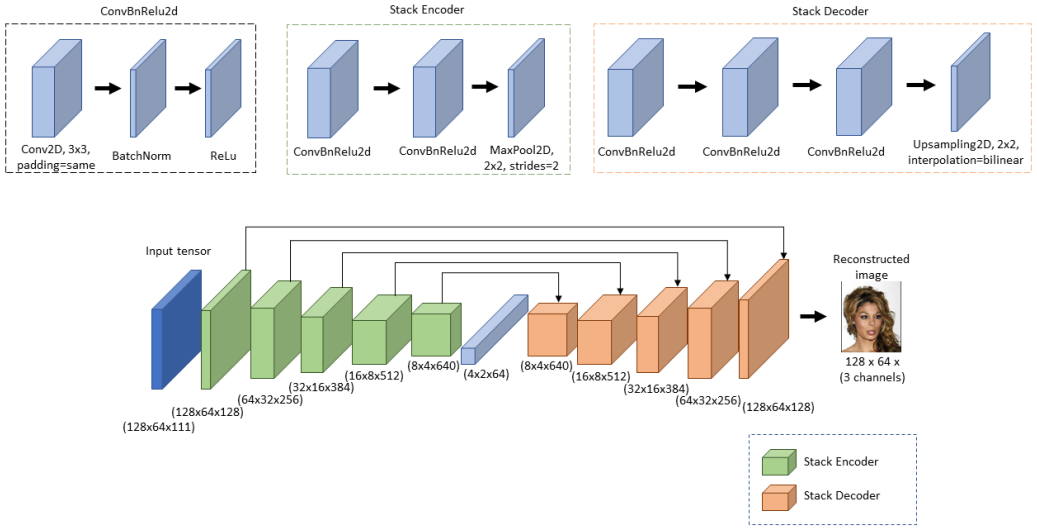


Figure 3: Architecture of recon-net

2.4 Stage 3

2.4.1 Face generation

For the task of face generation we will make use of the auto-encoder trained in stage-2. The overview of this stage is shown in Figure 4. A candidate image to be obfuscated will first be disentangled into the 3 factors as in stage-1. Before performing the final concatenation operation, we will manipulate the attributes factor. We state that attribute manipulation must be reasonable, attributes such as 'gender' and those related to gender should not be changed. On the other hand, attributes which have stronger obfuscation properties such as 'age' should be manipulated [15].

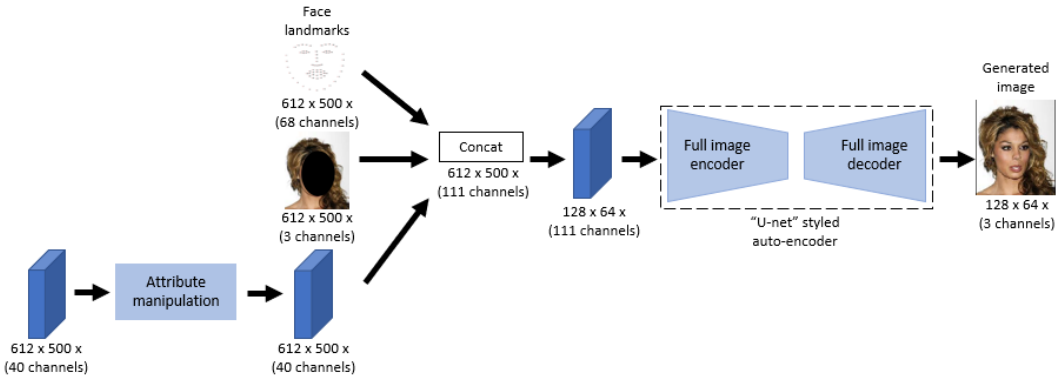


Figure 4: Image generation via face attribute manipulation

3 EXPERIMENTS

3.1 EXPERIMENT SETUP

For the setup of the experiment[1], we used the CelebFaces Attributes Dataset (CelebA) [17]. The data consists of 202,599 face image, with 10,177 identities and 40 binary attributes annotations per image. For training purposes, only a subset of the dataset is used. We use $subset = 0.3$ for our experiments. So for each train,val,test split from the original dataset, we take the first 30% of each of these splits for our experiments due to memory constraints. The next sections will detail the pre-processing steps as well as the outcome of the experiments for each stage. The hardware used for our experiments is one Nvidia RTX 2080ti and the deep learning framework is Keras with a Tensorflow backend.

3.2 STAGE 1

3.2.1 LANDMARK DETECTION

For landmark detection, the pre-processing steps include taking the subset of the original data. We then end up with the following sample sizes, $N_{train} = 48831, N_{val} = 5960, N_{test} = 5988$. We also resize the image to fit the requirements of the pre-trained detector.

For the face alignment and landmark detection task, the steps we proposed is as follows, we will first detect the face and if the face is not detected, we will filter away these images. For those that have faces detected, we can then move on to predict the 68 face landmarks. We are then left with the following sample sizes, $N_{train} = 47345, N_{val} = 5787, N_{test} = 5840$

3.2.2 FACE MASKING

For face masking this is done using the ellipse method provided by [3] to create personalized face mask. We also select specific landmark points such as the nose, the forehead, chin and cheeks for the required arguments in order to draw the ellipse. The output for this is the masked array of shape $(612, 500, 3)$.

3.2.3 FACE ATTRIBUTE CLASSIFICATION

For face attribute classification, the pre-processing steps include first normalizing the data and then one-hot encoding the 40 face attributes. We then used a raw ResNet-50 architecture. For the hyper-parameters of our model we set it as such, $(batchsize = 64, epochs = 15)$. For our baseline model, the optimizer that we chose is the Adam with the default learning rates. We set our activation functions in our output layer as sigmoid activations such that we can get the probabilities of each class. The corresponding loss function that we will use for each node is the binary cross-entropy. For evaluation criterion, the authors in [19] used the approach of average accuracy over the labels, we will adopt this approach. The experimental results are shown in Table 1. We managed to get an average test accuracy of 89.07% of our baseline classifier which is a little off from the top accuracy that the authors managed to achieve. However owing to the fact that we did not manage to train on the entire dataset, it would not be wise to compare with the performance of their experiment. We also trained another classifier however this time, we set the learning rate to a constant rate of 0.0001. The average accuracy for this model however dropped to 87.9%.

Table 1: Face attribute classification results

Setting	**Average Test Accuracy**	**Runtime**
adam	89.07%	53 mins
adam with lr=0.0001	87.90%	50 mins

[1] https://github.com/Juznauf/IS470-public

3.3 STAGE 2

3.3.1 CONCATENATION

For the concatenation operation in stage-2, it involves us converting the initial landmark shape to a tensor. We convert the initial landmark shape from $(68, 2)$ to $(612, 500, 68)$. We also have to upsample the face attributes and then finally we concatenate the three inputs and resize the resultant array from $(612, 500, 111)$ to $(128, 64, 111)$. As the resultant batch of images $(47345, 128, 64, 111)$ is far too large to fit into memory, we create a custom generator class which processes and creates batches of images for model training.

3.3.2 RECONSTRUCTION NETWORK

The architecture of the auto-encoder is shown in Figure 3. It consists of convolutional blocks and skip connections. We do not need to employ the cropping operation in our skip connections as we have set the padding to output the same size for all layers. The resultant model has a total of 45,121,667 trainable parameters. For the hyper-parameters of our model we set it as such, $(batchsize = 16, epochs = 15)$. The optimizer we adopt is the Adam optimizer and the loss function is the SSIM loss. The training results are shown in Table 2. For the SSIM score, a higher score is better. We also show a sample of images reconstructed at epochs 3,10 and 15 which can be found in Figure 5.

Table 2: Face reconstruction results

Setting	**Test SSIM Score**	**Runtime**
adam (*lr*=2e-5, $\beta_1 = 0.5, \beta_2 = 0.999$)	0.8752	30 hrs

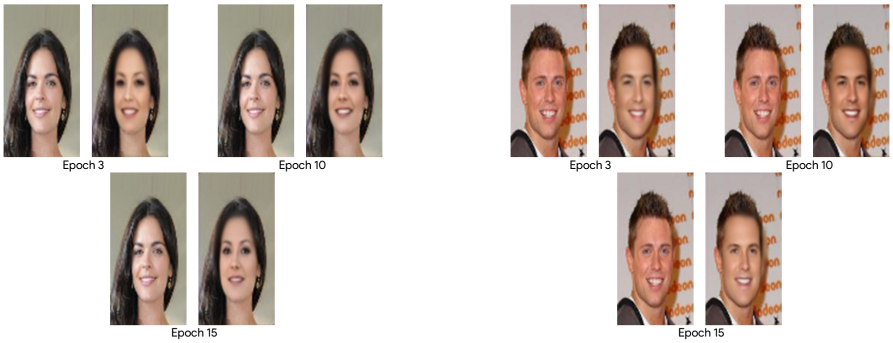


Figure 5: Reconstructed image pairs (original, reconstructed)

3.4 STAGE 3

3.4.1 GENERATION

The attributes which we chose to manipulate for our experiments are 'pointy_nose', 'smiling', 'young' and 'big_nose'. In order to manipulate the attributes, we modified the data generator such that for the above 4 attributes, their attribute vectors would be inverted. If the attribute is present, we change it to missing and vice versa. We present a sample of obfuscated face images in Figure 6.

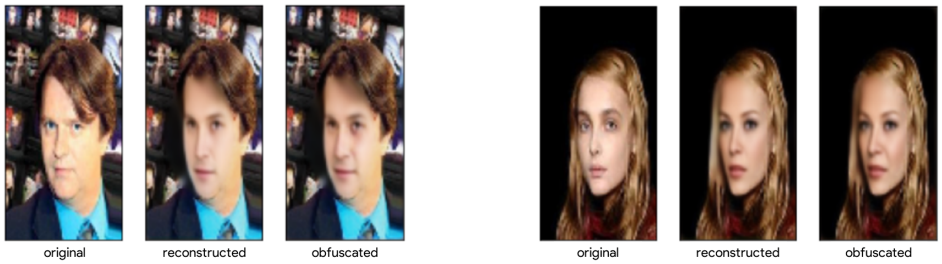


Figure 6: sample of obfuscated images

3.5 EVALUATION

We evaluate the effectiveness of our proposed obfuscation via a machine person-recognizer. As the identities in CelebA are separated by the researchers across the different splits, we were not able to train and test a person recognizer using the train/test split given by the researchers. To circumvent this, we instead used a subset of 20 identities selected from the portion of the test set which we did not use for the earlier experiments.

The pre-processing steps comprise of resizing the images to $(128, 64)$ to be consistent with our pre-processing for image generation and then to $(227, 227)$ to fit the requirements of our network. We then normalized the pixels and end up with the following sample sizes, $N_{train} = 270, N_{test} = 30$.

The architecture of the person recognizer is adapted from AlexNet [14]. We then extend the original network with 6 additional dense layers. The optimizer we adopt is the Adam optimizer and the loss function is the categorical cross-entropy loss. The output layer consists of 20 units which corresponds to our 20 identities.

The results of our network when tested against the original images and the obfuscated images are shown in Table 3. We find that when the obfuscation involves changing all attributes, the test accuracy drops by about 30% and when we apply the obfuscation used in section 3.4.1, the test accuracy drops by about 24%. However the trade-off for changing all attributes is that the human perception of the image is lost, which is in contrast with changing just the above-mentioned attributes. We suggest changing just the above 4 attributes in order to balance between effective obfuscation and human perception of the obfuscated images.

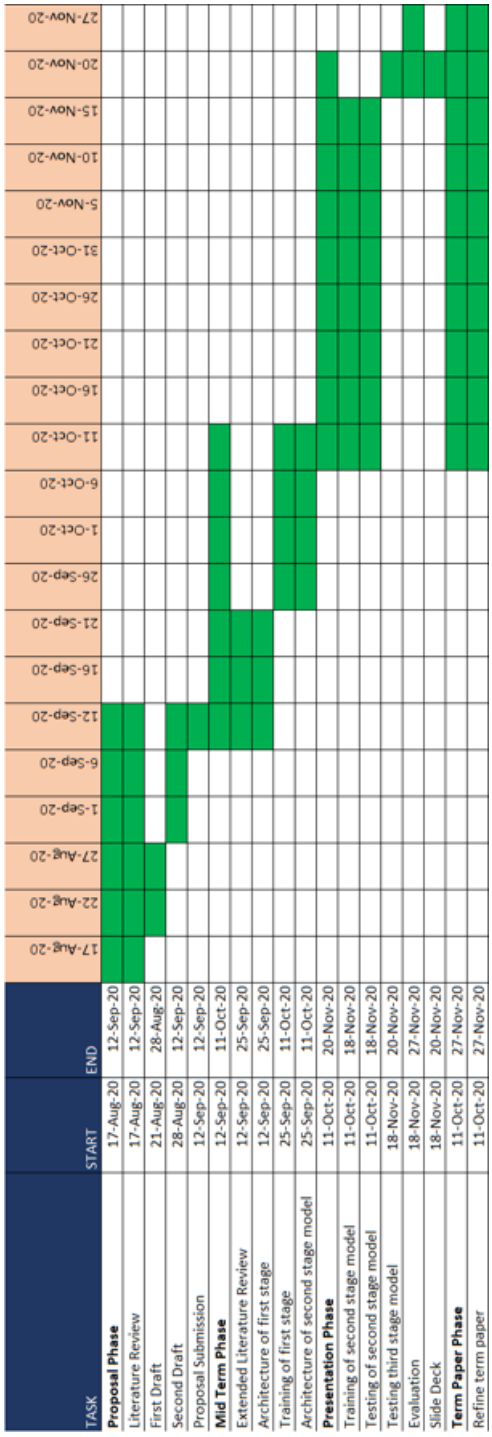
Table 3: Evaluation of obfuscation

Image settings	**Test Accuracy**
original	60.00%
obfuscated(all)	30.00%
obfuscated('pointy_nose', 'smiling', 'young' and 'big_nose')	36.67%

4 CONCLUSION

Our research has shown that generating face images via face attribute manipulation does enable effective obfuscation albeit to a certain degree. In the future, we hope to extend on the work that has been done with further research on the topic of conditional image generation as well as further experimentation to improve on the baseline performances that have been shown in this paper. We also note that the 3-stage framework which we have proposed could potentially be implemented in a real world setting, such as automatically generating obfuscated face images when confronted with web-scrapers at scale.

5 TIMELINE OF WORK



TASK	START	END
Proposal Phase	17-Aug-20	12-Sep-20
Literature Review	17-Aug-20	12-Sep-20
First Draft	21-Aug-20	28-Aug-20
Second Draft	28-Aug-20	12-Sep-20
Proposal Submission	12-Sep-20	12-Sep-20
Mid Term Phase	12-Sep-20	11-Oct-20
Extended Literature Review	12-Sep-20	25-Sep-20
Architecture of first stage	12-Sep-20	25-Sep-20
Training of first stage	25-Sep-20	11-Oct-20
Architecture of second stage model	25-Sep-20	11-Oct-20
Presentation Phase	11-Oct-20	20-Nov-20
Training of second stage model	11-Oct-20	18-Nov-20
Testing of second stage model	11-Oct-20	18-Nov-20
Testing third stage model	18-Nov-20	20-Nov-20
Evaluation	18-Nov-20	27-Nov-20
Slide Deck	18-Nov-20	20-Nov-20
Term Paper Phase	11-Oct-20	27-Nov-20
Refine term paper	11-Oct-20	27-Nov-20

ACKNOWLEDGEMENTS

This research would not have been possible without the advise and patience of Prof. Sun Qianru. I went from a complete novice in computer vision, to someone who has gained a little intuition about this exciting field and for that, I am very thankful.

REFERENCES

[1] Abhijit Ahaskar. How posting photos online can compromise privacy, Mar 2019. URL https://www.livemint.com/technology/tech-news/how-posting-photos-online-can-compromise-privacy-1553787545932.html.

[2] Dmitri Bitouk, Neeraj Kumar, Samreen Dhillon, Peter Belhumeur, and Shree K Nayar. Face swapping: automatically replacing faces in photographs. In *ACM SIGGRAPH 2008 papers*, pp. 1–8. 2008.

[3] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[4] Xinyi Ding, Zohreh Raziei, Eric C Larson, Eli V Olinick, Paul Krueger, and Michael Hahsler. Swapped face detection using deep learning and subjective assessment. *EURASIP Journal on Information Security*, 2020:1–12, 2020.

[5] Alexey Dosovitskiy and Thomas Brox. Generating images with perceptual similarity metrics based on deep networks, 2016.

[6] Mary Eising. How to find stolen photos on the internet 2019 guideline. URL https://www.copytrack.com/how-to-find-stolen-images/.

[7] Rose Eveleth. How many photographs of you are out there in the world?, Nov 2015. URL https://www.theatlantic.com/technology/archive/2015/11/how-many-photographs-of-you-are-out-there-in-the-world/413389/#:~:text=In2014,accordingtoMary,intotal150yearsago.

[8] Raul Gomez. Understanding categorical cross-entropy loss, binary cross-entropy loss, softmax loss, logistic loss, focal loss and all those confusing names, 2018. URL https://gombru.github.io/2018/05/23/cross_entropy_loss/.

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

[10] Kashmir Hill. The secretive company that might end privacy as we know it, Jan 2020. URL https://www.nytimes.com/2020/01/18/technology/clearview-privacy-facial-recognition.html.

[11] Vahid Kazemi and Josephine Sullivan. One millisecond face alignment with an ensemble of regression trees. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1867–1874, 2014.

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

[13] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017.

[14] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.

[15] Tao Li and Lei Lin. Anonymousnet: Natural face de-identification with measurable privacy. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 0–0, 2019.

[16] Yifang Li, Nishant Vishwamitra, Hongxin Hu, Bart P Knijnenburg, and Kelly Caine. Effectiveness and users' experience of face blurring as a privacy protection for sharing photos via online social networks. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 61, pp. 803–807. SAGE Publications Sage CA: Los Angeles, CA, 2017.

[17] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

[18] Liqian Ma, Qianru Sun, Stamatios Georgoulis, Luc Van Gool, Bernt Schiele, and Mario Fritz. Disentangled person image generation. In *The IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[19] Longbiao Mao, Yan Yan, Jing-Hao Xue, and Hanzi Wang. Deep multi-task multi-label cnn for effective facial attribute classification, 2020.

[20] USC Suzanne Dworak-Peck School of Social Work staff. Stalking in the age of social media, Feb 2018. URL https://news.usc.edu/135757/stalking-in-the-age-of-social-media/.

[21] Seong Joon Oh, Rodrigo Benenson, Mario Fritz, and Bernt Schiele. Faceless person recognition: Privacy implications in social media. In *European Conference on Computer Vision*, pp. 19–35. Springer, 2016.

[22] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation, 2015.

[23] Adrian Rosebrock. Facial landmarks with dlib, opencv, and python, Apr 2020. URL https://www.pyimagesearch.com/2017/04/03/facial-landmarks-dlib-opencv-python/.

[24] Mahmood Sharif, Sruti Bhagavatula, Lujo Bauer, and Michael K Reiter. Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition. In *Proceedings of the 2016 acm sigsac conference on computer and communications security*, pp. 1528–1540, 2016.

[25] Qianru Sun, Liqian Ma, Seong Joon Oh, Luc Van Gool, Bernt Schiele, and Mario Fritz. Natural and effective obfuscation by head inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5050–5059, 2018.

[26] Qianru Sun, Ayush Tewari, Weipeng Xu, Mario Fritz, Christian Theobalt, and Bernt Schiele. A hybrid model for identity obfuscation by face replacement. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 553–569, 2018.

[27] J. Tekli, B. al Bouna, R. Couturier, G. Tekli, Z. al Zein, and M. Kamradt. A framework for evaluating image obfuscation under deep learning-assisted privacy attacks. In *2019 17th International Conference on Privacy, Security and Trust (PST)*, pp. 1–10, Los Alamitos, CA, USA, aug 2019. IEEE Computer Society. doi: 10.1109/PST47121.2019.8949040. URL https://doi.ieeecomputersociety.org/10.1109/PST47121.2019.8949040.

[28] Xiao Yang, Yinpeng Dong, Tianyu Pang, Jun Zhu, and Hang Su. Towards privacy protection by generating adversarial identity masks. *arXiv preprint arXiv:2003.06814*, 2020.

[29] Zhou Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4): 600–612, 2004. doi: 10.1109/TIP.2003.819861.