

# Natural and Effective Obfuscation by Head Inpainting

## Supplementary Materials

Qianru Sun<sup>1\*</sup> Liqian Ma<sup>2\*</sup> Seong Joon Oh<sup>1</sup>  
Luc Van Gool<sup>2,3</sup> Bernt Schiele<sup>1</sup> Mario Fritz<sup>1</sup>

<sup>1</sup>Max Planck Institute for Informatics, Saarland Informatics Campus

<sup>2</sup>KU-Leuven/PSI, Toyota Motor Europe (TRACE) <sup>3</sup>ETH Zurich

{qsun, joon, schiele, mfritz}@mpi-inf.mpg.de

{liqian.ma, luc.vangool}@esat.kuleuven.be vangool@vision.ee.ethz.ch

These supplementary materials include additional details in network architecture (§1) and training (§2), as well as extended figures and tables: §3 and §4 introduce extensions of Figure 5 and Table 1 in the main paper, respectively.

### 1. Network architectures

In Figure 1, Figure 2 and Figure 3, we present three architectures respectively for the Encoder of Landmark Generator  $G_L$ , the landmark Auto-encoder (for pre-training the AEDec) and the Head Generator  $G_H$ .

In Figure 3, we should note that the output of the deep network is the intact image (256x256x3) including the body and head. It is then post-processed by cropping and pasting based on the head mask and the blackhead image. Therefore, in the final output only the head region is generated.

### 2. Implementation details

Both the landmark generator and head generator are trained with the Adam optimizer [1] with the weights  $\lambda_L = 2$  (in the main paper Equation (3)) and  $\lambda_H = 50$  (in the main paper Equation (5)). Initial learning rates (for both generator and discriminator) are  $2 \times 10^{-5}$ , and it decays to half every 5,000 iterations.

For landmark generation models, the minibatch size of landmark generation models is set to 16; optimization stops after 10,000 iterations; each iteration consists of 5 and 1 parameter updates for the generator and the discriminator, respectively. We have 34,383 training data in total. Therefore, it is about 23.3 epoches for training the generator and 4.7 epoches for training the discriminator. For AEDec, we train the landmark Auto-encoder with the minibatch size 16; optimization stops after 60,000 iterations.

For head generation models, the minibatch size of landmark generation models is set to 6; optimization stops after 13,000 iterations; each iteration consists of 5 and 1 parameter updates for the generator and the discriminator, respectively. It is about 8.7 epoches for training the generator and 1.7 epoches for training the discriminator.

### 3. Visualization results

In this section, we show the visualization results using different landmark generation models, as a supplement to the Figure 5 of main paper. Specific landmark models are  $L_2$  ( $L_2$  loss was used in the Table 1 of main paper) with Scratch Decoder,  $L_2 + D_L$  with Scratch Decoder,  $L_2 + D_L$  with AE Decoder and  $L_2 + D_L$  with PDM Decoder.

Figure 4 presents the results with blurhead images as input. In most cases, we achieve the best visual quality as well as the lowest landmark generation errors using the PDMDec model.

Figure 5 presents the results with blackhead images as input. Similar to blurhead results, the PDMDec model contributes to the best visual quality. It is worth to note that the smaller landmark error (mean  $L_2$  distance) do not mean the better visualization quality. The prediction of face pose and position depends on the body/scene context in the blackhead case. The quality of the generation is evaluated according to the facial organ consistency when the pose and position are reasonable. The mean  $L_2$  distance to the detected landmarks (used as *ground truth*) is only a reference.

Additionally in Figure 6, we show some examples using direct copy-paste method, corresponding the “NN head copy-paste” row in the Table 1 of main paper. Candidate images are searched in the training data based on the normalized  $L_2$  distance between detected landmarks. The face poses match the bodies in most cases, but the method results in unpleasant output images.

\*Equal contribution.

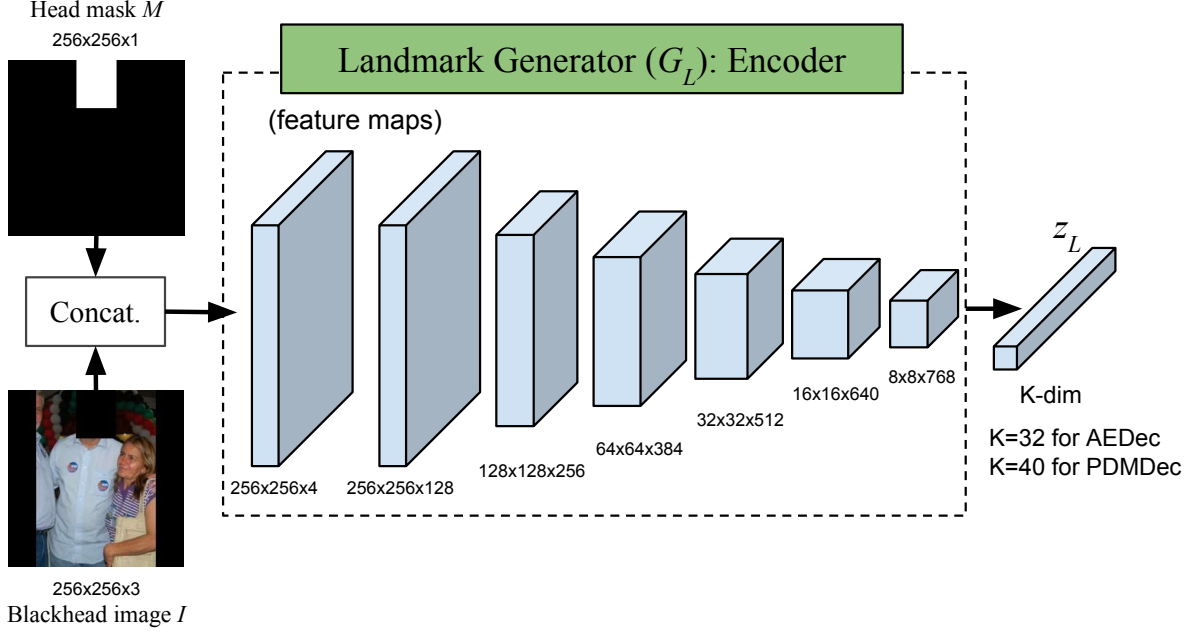


Figure 1: The architecture of the Encoder used in Landmark Generator  $G_L$ .

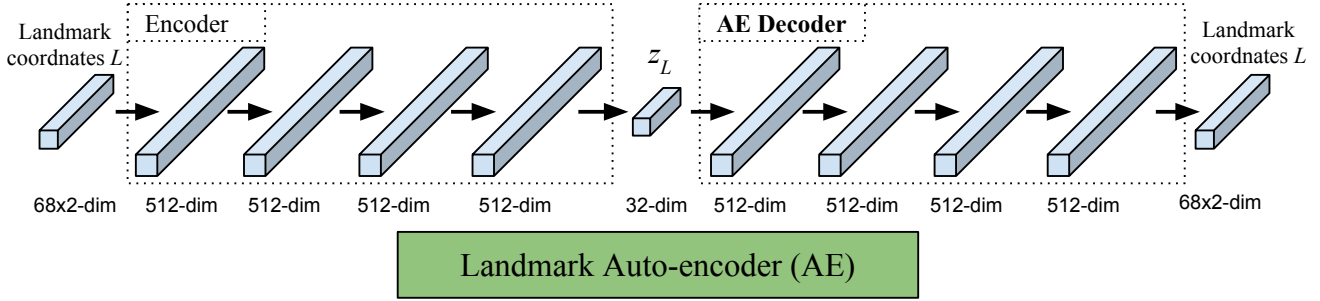


Figure 2: The architecture of the Auto-encoder used for pre-training the AE Decoder (AEDec). The pre-trained AE Decoder will be connected to  $G_L$  Encoder through the bottleneck layer  $z_L$ .

#### 4. Obfuscation performance against AlexNet

Experiments in the main paper have focused on the obfuscation performance with respect to a GoogleNet-based recognizer. However, as argued in the main paper, our obfuscation approach is *target-generic*: it is not generated with respect to a particular recognition system and is expected to work against a generic system.

This section additionally shows the obfuscation performance on an AlexNet-based recognizer. We use the same “feature extraction - SVM prediction” framework as in the main paper; we replace the feature extractor by AlexNet. See Table 1 for the quantitative comparison between GoogleNet and AlexNet recognizers.

The two recognizers exhibit different behaviours. First of all, on clean images, AlexNet performs worse than

GoogleNet ( $81.6\% < 85.6\%$ ), while on head-inpainted images, AlexNet shows greater robustness (e.g. 37.9% versus 45.1% on “Blur input -  $L_2 + D_L$  - PDMDec”). We also observe systematically less contributions from the head region: 72.2% (GoogleNet) versus 66.0% (AlexNet) on clean images, and consistent drop in head contribution on inpainted images (20% ~ 30%). AlexNet predictions are supported more by non-head regions, at least partially explaining its robustness against head obfuscation.

Although AlexNet recognizer turns out to behave quite differently from the GoogleNet model, we still reach the same conclusion regarding the superiority of our inpainting-based obfuscation over common patterns like blacking or blurring. For *body+head*, our inpainting method (“Blur/black input -  $L_2 + D_L$  - PDMDec”) decreases the recognition rate from 67.0% to 45.6% for blurheads, and

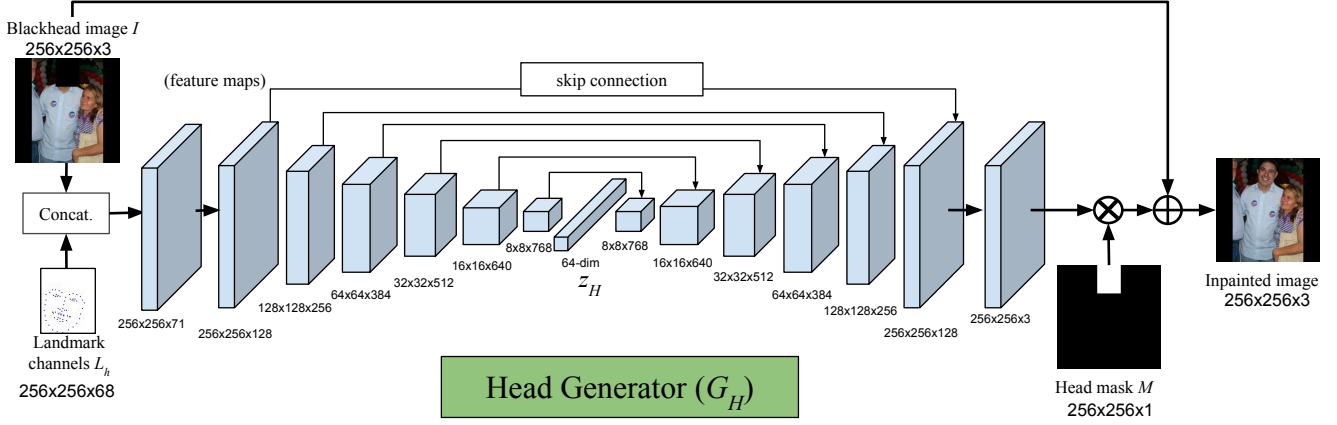


Figure 3: The network architecture of Head Generator  $G_H$ . It is based on the “U-net”. Noting that the landmark channels are 256x256x68 but are cropped to the head region size in this figure only for a better visualization of points.

Table 1: Evaluation of proposed obfuscation methods against two person recognizers in terms of person recognition rates. This table is an extension of the recognition results in Table 1 of the main paper.

Obfuscation method			Obfuscation against person recognizer					
Input	Landmark		GoogleNet			AlexNet		
	Loss	Decoder	head	body+head	head contrib.	head	body+head	head contrib.
Original	No head inpainting		85.6%	88.3%	72.2%	81.6%	85.3%	66.0%
Original	NN head copy-paste		1.2%	7.1%	67.5%	1.4%	6.1%	46.2%
Blur	No head inpainting		52.2%	71.6%	3.2%	52.0%	67.0%	20.6%
Blur	Detected landmarks		43.7%	51.7%	70.8%	49.0%	48.9%	37.2%
Blur	$L_2$	Scratch	36.2%	48.4%	66.8%	44.6%	44.6%	36.7%
Blur	$L_2+D_L$	Scratch	38.0%	48.4%	66.6%	44.9%	45.1%	38.9%
Blur	$L_2+D_L$	AEDec	37.5%	48.0%	66.1%	43.9%	45.0%	37.5%
Blur	$L_2+D_L$	PDMDec	37.9%	49.1%	66.7%	45.1%	45.6%	38.0%
Black	No head inpainting		2.1%	67.0%	14.0%	2.1%	63.2%	1.7%
Black	Detected landmarks		10.1%	21.4%	70.8%	11.4%	20.5%	46.3%
Black	NN landmarks		7.9%	20.4%	71.3%	10.1%	19.0%	46.0%
Black	$L_2$	Scratch	5.8%	17.4%	73.6%	7.5%	16.3%	49.0%
Black	$L_2+D_L$	Scratch	5.8%	17.2%	71.4%	7.5%	16.4%	47.4%
Black	$L_2+D_L$	AEDec	5.6%	17.4%	72.5%	7.5%	17.0%	48.7%
Black	$L_2+D_L$	PDMDec	5.6%	17.4%	71.0%	7.4%	16.6%	51.2%

from 63.2% to 16.6% for blackheads. Finally, we again observe that the contribution from head region increases as our method inpaints realistic head images. This leads to the same conclusion as for GoogleNet in the main paper: inpainted head images direct recognizer attention to head region, inducing a wrong decision based on the inpainted head.

## References

- [1] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv*, 1412.6980, 2014. 1









































	Blurhead Image	Original Image	$L_2$ (Scratch)	$L_2+D_L$ (Scratch)	$L_2+D_L$ (AEDec)	$L_2+D_L$ (PDMDec)
ID: 663			 Landmark $L_2$ : 5.85	 3.25	 3.06	 2.07
ID: 663						
ID: 659			 2.68	 1.96	 3.01	 1.90
ID: 659						
ID: 690			 6.54	 2.35	 4.21	 2.56
ID: 690						
ID: 691			 2.66	 2.21	 1.91	 1.39
ID: 691						

Figure 4: Visualization results on PIPA dataset. The input is blurhead image both for landmark generation and head generation. Landmark generation error (the distance to the detected ones) is given under each instance.


























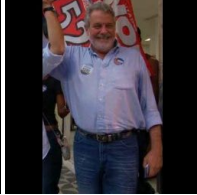














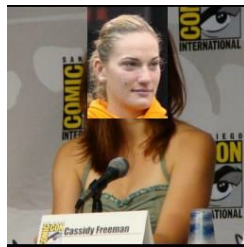
	Blackhead Image	Original Image	$L_2$ (Scratch)	$L_2+D_L$ (Scratch)	$L_2+D_L$ (AEDec)	$L_2+D_L$ (PDMDec)
ID: 663			 Landmark $L_2$ : 17.03	 9.82	 10.99	 14.40
ID: 663						
ID: 659			 16.74	 24.60	 25.52	 15.91
ID: 659						
ID: 690			 1.90	 3.37	 2.62	 2.73
ID: 690						
ID: 691			 4.57	 2.84	 3.97	 3.98
ID: 691						

Figure 5: Visualization results on PIPA dataset. The input is blackhead image both for landmark generation and head generation. Landmark generation error (the distance to the detected ones) is given under each instance.



ID: 663



ID: 659



ID: 690



ID: 691

Figure 6: Visualization results using the direct copy-paste method, corresponding the “NN head copy-paste” row in the Table 1 of main paper. Candidate head images are searched in the training data.