Temporally coherent video anonymization through GAN inpainting

Thangapavithraa Balaji[1], Patrick Blies[2], Georg Göri[2], Raphael Mitsch[2], Marcel Wasserer[2] and Torsten Schön[3]
[1]Audi AG, thangapavithraa.balaji@audi.de, [2]Enlite.ai
[3]Technische Hochschule Ingolstadt, torsten.schoen@thi.de

***Abstract*— This work tackles the problem of temporally coherent face anonymization in natural video streams. We propose *JaGAN*, a two-stage system starting with detecting and masking out faces with black image patches in all individual frames of the video. The second stage leverages a privacy-preserving Video Generative Adversarial Network designed to inpaint the missing image patches with artificially generated faces. Our initial experiments reveal that image based generative models are not capable of inpainting patches showing temporal coherent appearance across neighboring video frames. To address this issue we introduce a newly curated video collection, which is made publicly available for the research community along with this paper[1]. We also introduce the Identity Invariance Score IdI as a means to quantify temporal coherency between neighboring frames.**

I. Introduction

Autonomous systems and applications relying on deep learning based image processing require to be trained on data which is collected in public areas such as rural street scenes. One prominent example in this context is the development of autonomous vehicles (e.g. self driving cars). Naturally, this data can include persons which did not explicitly accept the collection of their data or even might not be aware of it. To protect people's personal rights, the European Union adopted the General Data Protection Regulation (GDPR) in 2016 [5]. As a consequence of this regulation, it is required to anonymize persons in images taken from public areas which is typically done by cutting out, blurring or pixelating human faces [26][6][12]. Although anonymization is ensured, these approaches come with the major drawback of modifying the original image structure. When dealing with temporal video data, this effect is even more visible, as individually anonymized frames leave a flickering effect on the video sequence. In this paper, we introduce a temporally coherent video anonymization technique to solve these issues called JaGAN, the faceless network.

1) Our Contributions: Our proposed method follows a two step approach by first identifying and masking out faces in individual frames and then inpainting the missing content with an artificially generated new face using a temporally coherent Generative Adversarial Network [8]. We demonstrate that using such a method not only allows to generate consistent faces across several image frames within a video sequence, but also leads to more natural looking faces and a higher FID score [10] even in single images compared to state of the art methods [34][13][39]. To ensure a full anonymization without leaking any information from the original face, our approach avoids using any facial characteristics or features such as landmarks. Results in Section VI-A demonstrate that with our proposed method we improve the state of the art in landmark–free facial inpainting. In order to measure and compare the temporal coherency of generated faces along a sequence of frames, we introduce an identity invariance score in Section VI-B.1. Our results on single frames are evaluated on the FDF dataset [13] whereas for evaluating the temporal coherence video sequences of faces are required.

For this purpose, we present and release a novel large scale video dataset comprising approximately 400.000 sequences with a length of 30 frames each. All sequences in the collection have been crawled from Youtube – considering only videos released under the Creative Commons Licence – to provide a publicly available dataset for training and evaluation of temporally coherent generative inpainting methods. Fig. 3 shows an example of a video contained in the dataset along with the inpainting results of our image-based as well as video-based anonymization GAN.

II. Related Work

Ad hoc obfuscation techniques like blurring, pixelation and various other filters [26][6][12] are widely used to anonymize privacy sensitive information of individuals in single images. However, images anonymized using these techniques are susceptible to loss of quality and utility. McPherson *et al.* [23] demonstrate that persons anonymized using these techniques can also be identified using face recognition algorithms due to their robustness. Many ad hoc methods, to a certain degree, fail to remove privacy sensitive information and prove to be inadequate in de-identifying the persons in the images [30][4][27].

Since the introduction of Generative Adversarial Nets [8], there has been several attempts to adapt GAN models for tasks such as text-to-image generation [42][28], domain transfer [44][14][20], super resolution image generation [15][18] and image completion purposes [25][19][40].

The scope of this paper is to reuse the central idea of GANs to design a model for the task of temporally coherent video anonymization.

a) Face Completion: Face completion is a more challenging task compared to image completion, since for image completion the model can pick up contextually similar

[1]Download script is available at: `https://github.com/cvims/jagan`

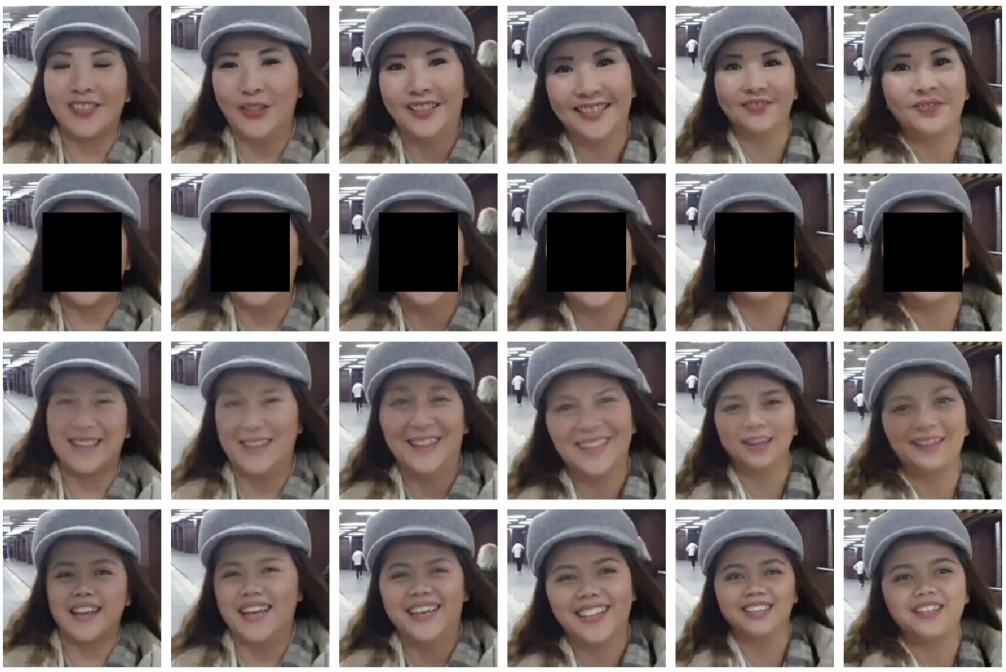


Fig. 1: Video dataset and generation examples (skipping 5 frames between each column). First row: real images. Second row: anonymization areas and conditional input for our generator. Third row: images generated with an image–based model. Note the shift in the identity of the generated face. Fourth row: video generative model enforcing temporal coherence. The identity of the generated faces remains the same across frames.

patterns to reuse them for filling up the missing regions. In contrast, face completion tasks require the generation of unique patterns and details not present in the input image (e.g., generating lips, eyes) [19][40]. The drawback of the works above is that the results are not convincing in terms of quality, anonymization (which was not their scope) and especially real-world applicability as the dataset used is CelebA containing only frontal facing faces. Wu *et al.* [39] propose an approach to perform inpainting with spatio-temporal consistency in videos.

b) Anonymization with Face Generation: Ren *et al.* [29] propose a video anonymization method to modify the privacy sensitive data of individuals on pixel level. Shirai *et al.* [33] demonstrate the concept of generating a surrogate image and performing a conditional based training. Wang *et al.* [38] propose the concept of video-to-video synthesis with conditional style transfer. The spatio-temporal adversarial objective introduced enforces the generation of temporally coherent videos. CIAGAN [22] not only performs face inpainting but controls the de-identification procedure as well. DeepPrivacy [13] is most similar to our work and serves as a basis for the development of our method as well as our experimental evaluation. The conditional generator allows to maintain a seamless transition between the generated face and the surrounding background.

III. Datasets

This section describe the image and video data used in our evaluations.

A. FDF Image Dataset

The Flickr Diverse Dataset (FDF) has been crawled from the website Flickr and contains 1.47 Million faces in the wild with a minimum resolution of 128×128 pixels. The faces have been automatically annotated with 7 facial landmarks as well as with a bounding box and have a large diversity in terms of age, ethnicity, facial painting, facial pose, occluding objects and image background. FDF is implicitly designed for training generative models with the purpose of face anonymization. Further details are given by Hukkelas *et al.* [13].

B. Video Dataset

We curate and release a large scale video dataset of human faces in 99,795 different videos to enable the training and evaluation of video based person anonymization methods as proposed in Section V. The Creative Commons licensed videos (*CC-BY*) are crawled from public YouTube channels and selected to preserve diversity in terms of gender, age, ethnicity and poses. The data collection was performed in a 2-stage process: first, we collected a dataset of *seed faces*, for which we then collected videos at a later stage. Fig. 2 shows and overview of the entire data set collection pipeline for which we describe the individual components in detail below.

1) Seed Faces Collection Process (Crawling Stage): In this section we describe our pipeline to collect *relevant* content from YouTube. By relevant, we refer to a training

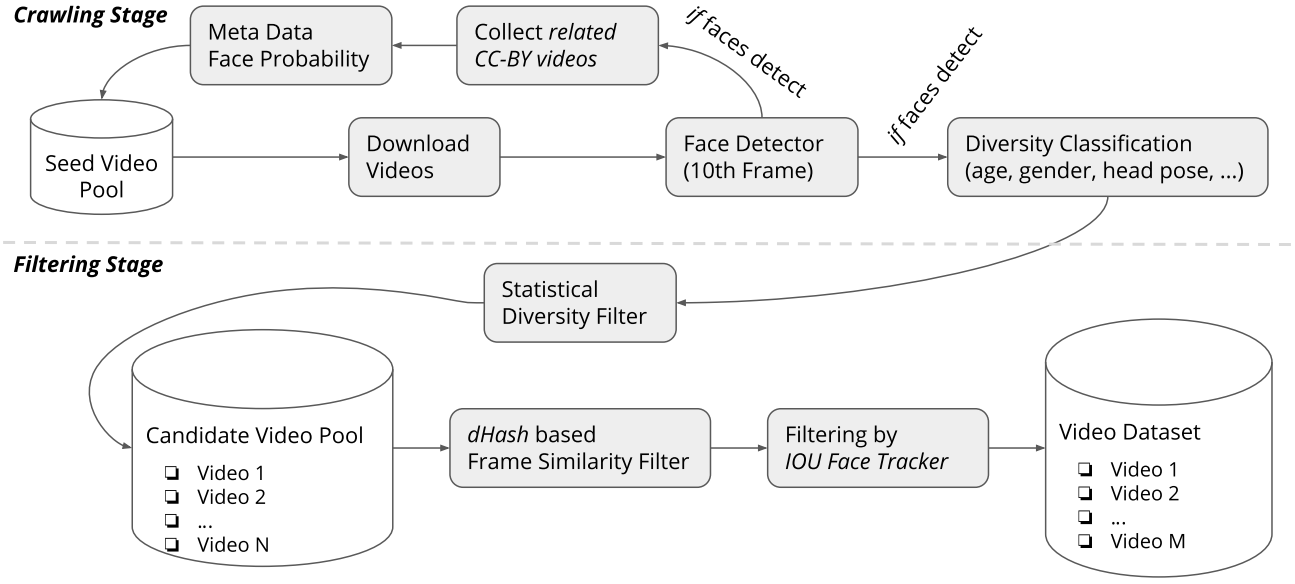


Fig. 2: Overview of video data set collection pipeline.

data set comprising images of faces that show a sufficiently large variability with respect to:

- faces with a large variety of angles and pitches
- faces with a large variety of ages
- female and male faces
- faces from people of different ethnicity

To start, we manually specify a set of seed videos for which we expect to have relevant facial examples for the anonymization task. For the first set of seed videos, we mainly used dashcam and outdoor videos. While running the pipeline, the list of seed videos is progressively extend by videos that have detected faces in them. This way, we end up having theoretically infinite supply of seed videos. At a later stage, after evaluation of the feature class distributions, we manually add queries to the seed video pool to obtain more instances of underrepresented categories.

For each video in the seed pool, we start by extracting the corresponding metadata and feed it as an input to a binary classifier predicting a probability that the video actually contains faces. This metadata pre-classification stage is video (image) agnostic and allows for an efficient way to filter the set of candidate videos. All videos passing this filter are forwarded to the subsequent stage of actual video analysis. Note, that the metadata classifier is repeatedly retrained with the growing set of actually analyzed videos serving as ground truth training data.

Each video passing the previous stage is further analysed, by extracting the 10th frame (empirically chosen) and by running a face detector (we have used a modified version of TinyFaces face detector [11]). If at least one face is found in this frame, we search YouTube for *related videos* and add those to the pool of seed videos. The faces detected in these videos are collected and added to a collection of seed faces.

Given the collection of seed face images we utilize a set of simple image classifiers tagging the individual images with the following classes for Ethnicity (*asian, black, white, other*), Age (*0-16, 16-50, 50+*) and Gender (*female, male*). For developing these image categorization models we use the UTKFace Dataset[43]. Additional we estimate the head rotation using PRN[7] and put the results into the following buckets (*front, half-pose, portrait*) [2].

For training, validation and test set, we select an equally distributed subset of sequences to contribute to the final dataset of seed faces. To reduce the risk of leakage of identities across the dataset splits, we used a single YouTube channel only for one split.

2) Video Collection Process (Filtering Stage): Given the pool of seed faces the video candidates are further analysed by trying to track these faces in the subsequent frames with an IOU Tracker [3]. If tracking is possible for at least 30 frames we further filter the sequences by ensuring that each frame in a series has a high *dHash* [17] similarity score. This last step discards sequences comprising abrupt changes such as a scene cuts. Regarding the minimum count of 30 frames we follow [38] serving as a basis for our video based anonymization models.

When compiling the dataset we crop centered on the face bounding box and scaled to size of 256px. The total number of faces contained in the final dataset to be released is shown in Table I,

Fig. 3 shows a representative subset of the frames contained in the dataset.

IV. GAN Based Image Anonymization

This section describes the architecture of our image based anonymization GAN along with the required pre–processing steps as well as the applied training procedure. Note that the scope of this work is not to improve the state of the art in

[2]We are aware that this automatic categorization is not ideal but keeping the scale of our dataset in mind so far the only feasible option.

Fig. 3: Example videos contained in the dataset. (For space reasons we only show every fifth time step of each 30 frame sequence.)

TABLE I: Size of train, validation and test sets

Set	No. of initial frames	Total no. of faces
Train	348,987	10,469,610
Validation	32,268	968,040
Test	33,571	1,007,130

image anonymization but to introduce temporal coherence when working with video data. We introduce and compare our model with a reference method [13] at this point to make sure that our design choices serve as a valid basis for the video anonymization network proposed in Section V.

A. Architecture

In general, our architecture (see Fig. 5) is an adaptation of [37] combined with ideas from other works such as the 2-stage approach proposed in [41], the spatially discounted reconstruction loss from [41] and R_1 regularization from [24].

When training on the FDF dataset, input images have the size 128×128, whereas for the proposed video dataset, the input images have the size 256×256. In the following, we denote the images dimensions simply with size $h \times w$.

1) The Coarse Stage: The coarse stage is based on the U–Net architecture [31] as shown in Fig. 4. From the pre–processing stage, we obtain a tensor of shape $(5, h, w)$[3]. This corresponds to an image with three color channels, one channel for the border mask and one channel for the anonymization area mask. As depicted in Fig. 4, several strided convolution–batch norm–leaky ReLU operations are applied to the input tensor. Each red arrow in the figure corresponds to a 4×4 convolution with stride 2 followed by batch normalization and a leaky ReLU activation function. The bottleneck layer has dimensions $(1000, 1, 1)$. After the bottleneck layer, we feed the tensor through several transposed convolution–batch norm–ReLU-concat skip connections operations. Each yellow arrow corresponds to a 4×4 transposed convolution with batch normalization and ReLU activation function. Each blue error corresponds to a copy–and–concat operation. The last layer in the coarse stage is a $\tanh$–layer.

2) The Fine Stage: The fine–stage generator network consists of three components: a convolutional front-end with strided convolutions (yellow in Fig. 5), a set of nine residual blocks (red in Fig. 5), a transposed convolutional back–end (implemented as upsampling + convolution, blue in Fig. 5) and finally a tanh output layer. The input to the fine stage again is a tensor of shape $(5, h, w)$. As the output of the coarse stage is only of size $(5, h/2, w/2)$ as proposed in [41] we rescale it to original input resolution $(5, h, w)$. Afterward we again combine it with the border and anonymization area mask before feeding it into the fine stage network.

3) The Discriminator: For the discriminator, we use the multi-scale discriminator approach of [37]. Like them, we use three discriminators with identical network structure operating at different image scales, leading to an increased receptive field of the discriminator. The real and synthesized images ($h \times w$ in our case) are down sampled by a factor of two and four, yielding images with resolutions $h/2 \times h/2$ and $h/4 \times h/4$ respectively. Each of the three discriminators is then trained to differentiate real and fake images at one of these scales. This way, the discriminator operating on the $h/4 \times h/4$ images has the largest receptive field and can guide the generator to generate globally consistent images while the discriminator operating on $h \times w$ encourages the discriminator to produce finer details.

4) Pre–Processing: The pre–processing pipeline is summarized in Fig. 6. First, faces are detected with DSFD. This yields bounding boxes for all faces in an image. Everything within the detected box is deleted (replaced with black pixels), resulting in the image shown in Fig. 6b. To get the context rectangle, the area of the bounding box is made quadratic by reducing the length of the larger edge (Fig. 6b) followed tripling the edge length of the resulting square (blue outline in Fig. 6c) Everything within the resulting square area is taken as context for the generator. If the enlarged bounding box extents beyond the border of the image, we add a corresponding region to the original image (Fig. 6d). The resulting image is centered on the face. We provide the

[3]We omit the batch size dimension for simplicity.

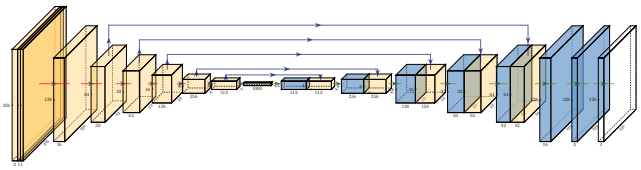


Fig. 4: Depiction of our coarse stage U–Net architecture for 256×256 input images. Yellow boxes correspond to feature maps after convolutions, blue boxes correspond to feature maps after transpose convolutions. The transparent box corresponds to a feature map after a tanh layer. Green arrows correspond to 4×4 (transpose) convolutions with batch normalization and Leaky ReLUs.

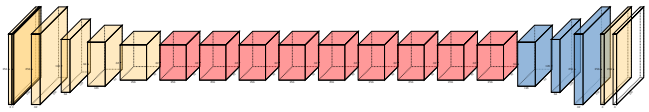


Fig. 5: Depiction of our fine stage architecture. Yellow boxes correspond to feature maps after convolutions, red boxes correspond to feature maps after residual blocks, blue boxes correspond to feature maps after transpose convolutions. The transparent box corresponds to the feature map after a tanh layer.

generator with additional information about the image border by supplying it with a border mask (blue in Fig. 6e). To indicate which area needs to be inpainted, we introduce an additional anonymization mask (green in Fig. 6e). The masks are concatenated to the image along the channel dimension, yielding our input tensor for the generator. In contrast to the model presented in [13] we avoid feeding facial landmarks as an input into the generator for two reasons: First, even though the image patch is replaced with black pixels, the position of landmarks might still leak information about the identity of the anonymized individual. Second, when extending the model towards video anonymization in Section V, inconsistent and noisy landmark detections in subsequent frames have a direct impact on the respective, generated video frames. In particular, false landmark detections lead to temporally inconsistent video frames with respect to facial features of the generated faces.

5) Optimization Details:

a) Loss Functions: After experimenting with different adversarial loss functions, such as the vanilla GAN loss [9], WGAN [2] and hinge–loss [35], we finally decided to apply the least squares loss from LSGAN [21] producing the most convincing results in terms of qualitative performance.

(a) (b) (c) (d) (e)

Fig. 6: Pre–processing steps for our image model

Another idea we reuse from [37] is that of the feature matching loss. The idea behind that loss term is to use the output of each layer of each of the three multi-scale discriminators to calculate the mean L_1-distance between the activation of generated and real images. This addition term helps to stabilize training by encouraging the generator to produce natural statistics at multiple scales for all layers of the discriminator.

In summary, we use the following loss components: For the coarse stage reconstruction loss and for the fine stage adversarial LSGAN loss, feature matching loss and spatially discounted reconstruction loss, additionally we regularize the discriminator with R_1 [24].

b) Training Procedure: For training we apply the two time–scale update rule from [10] and optimize the networks using Adam [16] with a learning rate of 0.0004 and a batch size of 256. We track the FID score [10] on the validation set for early stopping and best model selection. The selected model was trained over 89,928 mini-batches.

V. Temporally Coherent Video Anonymization

This section proposes a method for temporally coherent video anonymization. Starting from the image based model described above we establish temporally consistent generation results by implementing the temporal consistency terms added by [38]. Additional we apply *burn-in*, a technique to improve inference results for the starting frames of a video.

A. Architecture

The fine stage and the training regime described in Section IV are reused for the video model. However, in the discriminator and the input data representation we introduce major changes to support temporal coherence across the generated video frames. In addition to the existing image discriminator we use an additional video discriminator as described below.

1) Conditional Video Discriminator: Vid2Vid [38] proposes several methods to enforce temporal consistency, one of them being the conditional video discriminator, which we also employ for our video model. In summary, its purpose is to ensure that consecutive frames resemble the temporal dynamics of a real video, thereby enforcing temporal consistency between the frames. It does so by distinguishing between three frames of a generated and a real sequence. The video discriminator is multi–scale in space and time: the frame rates of the real and generated videos are downsampled by a factor of 3, skipping in–between frames in the process. This is done for up to three time scales ((t_0, t_{-1}, t_{-2}), (t_0, t_{-3}, t_{-6})) and (t_0, t_{-9}, t_{-18})) depending on how many generated frames are already available. The different time-scales encourage short– and long–term consistency. Together with 3 spatial scales per time scale, this results in up to 9 discriminators, which are used in training.

B. Pre–processing

Since we have conditionally strongly dependent frames in each sequence, one step to enforce temporal consistency is to extend the input that is fed into the generator network by adding past frames to the input tensor. This idea was proposed in [38]. They use two images from the past to condition the generator on the previous sequence. The reason for using two images is that they claim that fewer than two images lead to unstable results, while more than two images lead to a higher GPU workload and memory footprints without increasing quality. With this addition, the pre–processing pipeline changes in the following way: instead of feeding one image (+ anonymization and boundary masks) to the generator, we now feed three images (+ anonymization and boundary masks) to the generator: two images from the past and the current image that is to be inpainted at this step.

C. Burn-in Stage

Fig. 7a depicts the situation when starting to train or infer on a new sequence. As no previous images are available as an input for the generator when starting with a new sequence, we use zero–tensors as substitution during training. However, when performing inference, there is a noticeable effect of using zero–tensors at the beginning of a sequence: since we have no past information to condition the generator on, the identity of the newly generated faces changes significantly during the first frames. This can be clearly seen in Fig. 8. For this reason, we introduce a stage called *burn–in* for inference: Before we generate the first output frame, we create a series of 6 frames, where every subsequent generated frame gets the previous two as inputs. The number of *burn–in* frames is larger than the two conditional inputs, as we need multiple cycles to get rid of the suboptimal contribution of the initial frame.

This can be seen as a kind of bootstrapping since we generate a new face and use this newly generated face as artificial past information that we use to condition the next generated face on. Our experiments in Section VI reveal that the burn-in stage not only has a positive effect on subjectively perceived image quality but also on quantitative evaluation measures, which is why we use it for all inference calculations.

D. Optimization Details

We use the same losses and regularization as in the image case. Noteworthy is that we had to adjust the weight balance of the different loss terms, due the inclusion of the video discriminator. Also we trained the network with one joint step for discriminator and generator using Adam with a learning rate of 0.0004 and a batch size of 96.

As with the image model, we monitor a quantitative metric on the validation set to determine when to stop training and which model to pick for inference. As evaluation metric we use the FVD score proposed by [38] and [36], an extension of the FID to videos. The selected model was trained for 1,285,490 mini-batches.

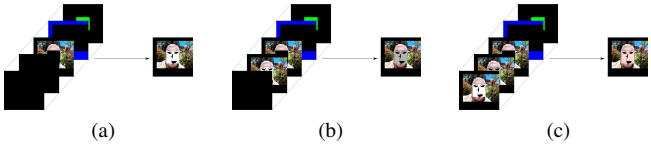


Fig. 7: Burn–in process: 7a shows the input tensor for the video model at the first step of the sequence: since there are no past generated images, we concatenate tensors filled with zeroes as past images. 7b depicts the second time step: we take the image that was generated in the first time step and use it as past image. 7c depicts the third time step: we take the image that was generated in the second time step and use this as the second past image. The cropped rectangular region is the missing part of the face that is to be anonymized by the model, green is the bounding box given by the face detector, blue is the border mask.

Fig. 8: The effect of burn–in for inference. Top row: without burn–in, bottom row: with burn–in.

VI. Experimental Evaluation

This section provides an experimental evaluation of both, the image based as well as the video based anonymization networks.

A. Image Models

To verify that the image–based model serves as a valid starting point for the video model presented in Section V we carry out initial experiments by comparing it to the anonymization GAN presented in [13]. We train our image model with the FDF dataset released by [13]. We then anonymize the faces of the FDF validation set and calculate the FID score as a quantitative evaluation measure. The results can be found in Table II. With our architecture, we achieve state of the art results for facial inpainting without using landmarks as an input to the generator. Given this result we conclude that our model serves as a valid starting point for the video anonymization setting as well.

B. Video Models

In this section, we evaluate the proposed video model in terms of quality (via the FVD score [38], [36]) as well as for temporal coherent face generation across multiple frames for the video model.

1) The Identity Invariance Score (IdI): To the best of our knowledge, there is no method measuring the temporal consistency in a way that humans perceive it. That is why we also provided videos for the relevant models as supplementary material for visual inspection. In these

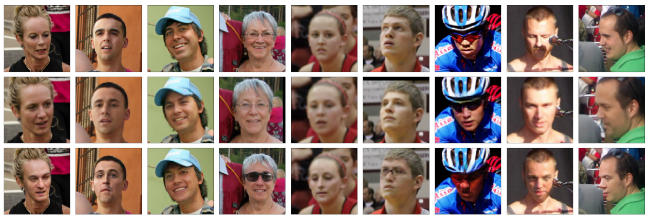


Fig. 9: Example images generated from the FDF validation set. The first row shows the original images, the second row shows images inpainted by our image model, the third shows images inpainted by DeepPrivacy 46M with LM [13].

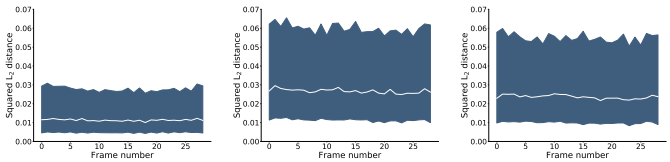
TABLE II: FID Scores calculated on FDF validation set (lower is better). Our method, trained on the FDF training dataset improves the state of the art in landmark–free face inpainting on the FDF validation dataset from 3.36 to 1.97.

Model	FID score
Our Image Model, trained on FDF training dataset [13]	1.97
DeepPrivacy 12M with LM [13]	2.71
DeepPrivacy 12M without LM [13]	3.36
DeepPrivacy 46M with LM [13]	1.84

videos a lack of temporal consistency can be observed as flickering and warping faces with changing identities. To obtain a quantitative metric for temporal coherence, we postulate that the temporal coherence in generated videos and an invariance of the identity of the generated faces are correlated, i.e., measuring identity invariance will allow conclusions to be drawn about temporal consistency. For this reason, we introduce the Identity Invariance (IdI) score.

To measure the average changes of the identity of a generated face in a sequence of frames, we proceed in the following way: First, we detect the faces with DSFD and calculate the Facenet [32] embeddings for each detected face. This results in sequences of Facenet embeddings with a length of 30 frames per sequence. We then calculate the squared L_2 distance of the Facenet embeddings between every two subsequent frames. This provides us a measure to estimate how much the identity of inpainted faces in two subsequent frames differ, which should be in turn correlated with temporal consistency. We use the squared L_2 distance here because this is what the authors of Facenet use in their paper to determine differences in identities. The distances are averaged over the whole sample.

To arrive at a point estimate for the temporal consistency across the test dataset, we take the median of the distances over the whole random sample for each frame. Plots of these medians over frames can be found in Fig. 10. We propose the *IdI Score* as the ratio of the median of squared L_2 distances of real frames over the median of squared L_2 distances of generated frames. The reason why we take the ratio over the distances of the original test set is that there are naturally occuring changes in the Facenet embeddings when a face moves. To take this into consideration, we normalize by the distance of the real distances.



(a) Test Set (b) Image Model (c) Video Model

Fig. 10: These plots show the median along with the IQR of the squared L_2 distance between the Facenet embeddings of 2 adjacent frames in a sequence of 30 frames. The distances are aggregated over every sequence in a random sample over the test set, which consists of about 50,000 frames. Smaller distances indicate fewer changes of identity in the sequence.

TABLE III: Metrics calculated on our test set (for the FVD scores and the distances, lower is better, for the IdI score, higher is better)

Model	FVD score	Med. sq. L_2	IdI score
Test Set	n/a	0.0113	1.00
Our Image Model	110	0.0268	0.42
Our Video Model	**59**	0.0237	**0.48**

Table III shows the IdI scores of our image and video datasets and for the real faces in the test set.

VII. Conclusion

We propose and release a large scale facial video dataset diverse in terms of age, gender, ethnicity and head poses for training and testing video based anonymization methods. Given this dataset, we develop a GAN-based method for inpainting generated faces into anonymized images and videos that is not dependent on any landmarks. On single frame images, we improved the state of the art FID score from 3.36 to 1.97 for landmark–free face inpainting. When working with video data our approach is able to generate coherent faces across sequences. It reduces identity shift across individual frames in comparison to an image (single frame) based inpainting model. In order to measure the degree of coherency, we introduced the identity invariance score and show in experimental evaluations that our video model helps to keep generated faces consistent over a sequence of frames.

We hope that the provided dataset as well as our baseline models help to foster further research in the field of image preserving face anonymization.

References

[1] *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*. IEEE Computer Society, 2018.

[2] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein GAN. *arXiv e-prints*, page arXiv:1701.07875, Jan. 2017.

[3] E. Bochinski, V. Eiselein, and T. Sikora. High-speed tracking-by-detection without using image information. In *14th IEEE International Conference on Advanced Video and Signal Based Surveillance, AVSS 2017, Lecce, Italy, August 29 - September 1, 2017*, pages 1–6. IEEE, 2017.

[4] M. Boyle, C. Edwards, and S. Greenberg. The effects of filtered video on awareness and privacy. In *Proceedings of the 2000 ACM Conference on Computer Supported Cooperative Work*, CSCW '00, page 1–10, New York, NY, USA, 2000. Association for Computing Machinery.

[5] Council of European Union. General data protection regulation (EU) no 2016/679, 2016. `http://data.europa.eu/eli/reg/2016/679/oj`.

[6] J. L. Crowley, J. Coutaz, and F. Bérard. Perceptual user interfaces: Things that see. *Commun. ACM*, 43(3):54–ff., Mar. 2000.

[7] Y. Feng, F. Wu, X. Shao, Y. Wang, and X. Zhou. Joint 3d face reconstruction and dense alignment with position map regression network. In *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XIV*, volume 11218 of *Lecture Notes in Computer Science*, pages 557–574. Springer, 2018.

[8] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc., 2014.

[9] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. C. Courville, and Y. Bengio. Generative adversarial nets. In , *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 2672–2680, 2014.

[10] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In , *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 6629–6640, 2017.

[11] P. Hu and D. Ramanan. Finding tiny faces. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[12] S. E. Hudson and I. Smith. Techniques for addressing fundamental privacy and disruption tradeoffs in awareness support systems. In *Proceedings of the 1996 ACM Conference on Computer Supported Cooperative Work*, CSCW '96, page 248–257, New York, NY, USA, 1996. Association for Computing Machinery.

[13] H. Hukkelås, R. Mester, and F. Lindseth. Deepprivacy: A generative adversarial network for face anonymization. In *Advances in Visual Computing*, pages 565–578, Cham, 2019. Springer International Publishing.

[14] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[15] T. Karras, T. Aila, S. Laine, and J. Lehtinen. Progressive Growing of GANs for Improved Quality, Stability, and Variation. *arXiv e-prints*, page arXiv:1710.10196, Oct. 2017.

[16] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[17] N. Krawetz. `http://www.hackerfactor.com/blog/?/archives/529-Kind-of-Like-That.html`.

[18] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi. Photo-realistic single image super-resolution using a generative adversarial network. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 105–114, July 2017.

[19] Y. Li, S. Liu, J. Yang, and M. Yang. Generative face completion. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 5892–5900, 2017.

[20] F. Luan, S. Paris, E. Shechtman, and K. Bala. Deep photo style transfer. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6997–7005, July 2017.

[21] X. Mao, Q. Li, H. Xie, R. Y. K. Lau, Z. Wang, and S. P. Smolley. Least squares generative adversarial networks. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 2813–2821. IEEE Computer Society, 2017.

[22] M. Maximov, I. Elezi, and L. Leal-Taixé. CIAGAN: conditional identity anonymization generative adversarial networks. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 5446–5455. IEEE, 2020.

[23] R. McPherson, R. Shokri, and V. Shmatikov. Defeating Image Obfuscation with Deep Learning. *arXiv e-prints*, page arXiv:1609.00408, Sept. 2016.

[24] L. M. Mescheder, A. Geiger, and S. Nowozin. Which training methods for gans do actually converge? In , *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *JMLR Workshop and Conference Proceedings*, pages 3478–3487. JMLR.org, 2018.

[25] M. Mirza and S. Osindero. Conditional Generative Adversarial Nets. *arXiv e-prints*, page arXiv:1411.1784, Nov. 2014.

[26] C. Neustaedter and S. N. Greenberg. Balancing privacy and awareness in home media spaces 1. 2003.

[27] E. M. Newton, L. Sweeney, and B. Malin. Preserving privacy by de-identifying face images. *IEEE Trans. on Knowl. and Data Eng.*, 17(2):232–243, Feb. 2005.

[28] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee. Generative Adversarial Text to Image Synthesis. *arXiv e-prints*, page arXiv:1605.05396, May 2016.

[29] Z. Ren, Y. J. Lee, and M. S. Ryoo. Learning to anonymize faces for privacy preserving action detection. In , *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part I*, volume 11205 of *Lecture Notes in Computer Science*, pages 639–655. Springer, 2018.

[30] S. Ribaric and N. Pavesic. An overview of face de-identification in still images and videos. *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, 04:1–6, 2015.

[31] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In , *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2015 - 18th International Conference Munich, Germany, October 5 - 9, 2015, Proceedings, Part III*, volume 9351 of *Lecture Notes in Computer Science*, pages 234–241. Springer, 2015.

[32] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 815–823. IEEE Computer Society, 2015.

[33] S. Shirai and J. Whitehill. Privacy-preserving annotation of face images through attribute-preserving face synthesis. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2019, Long Beach, CA, USA, June 16-20, 2019*. Computer Vision Foundation / IEEE, 2019.

[34] T. Tanimura, M. Kawano, T. Yonezawa, and J. Nakazawa. Ganonymizer: Image anonymization method integrating object detection and generative adversarial network. In *3rd EAI International Conference on IoT in Urban Space*, pages 109–121, Cham, 2020. Springer International Publishing.

[35] D. Tran, R. Ranganath, and D. M. Blei. Hierarchical Implicit Models and Likelihood-Free Variational Inference. *arXiv e-prints*, page arXiv:1702.08896, Feb. 2017.

[36] T. Unterthiner, S. van Steenkiste, K. Kurach, R. Marinier, M. Michalski, and S. Gelly. Towards Accurate Generative Models of Video: A New Metric & Challenges. *arXiv e-prints*, page arXiv:1812.01717, Dec. 2018.

[37] T. Wang, M. Liu, J. Zhu, A. Tao, J. Kautz, and B. Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018* [1], pages 8798–8807.

[38] T. Wang, M. Liu, J. Zhu, N. Yakovenko, A. Tao, J. Kautz, and B. Catanzaro. Video-to-video synthesis. In , *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada.*, pages 1152–1164, 2018.

[39] Y. Wu, V. Singh, and A. Kapoor. From image to video face inpainting: Spatial-temporal nested gan (stn-gan) for usability recovery. 11 2019.

[40] R. A. Yeh, C. Chen, T. Lim, A. G. Schwing, M. Hasegawa-Johnson, and M. N. Do. Semantic image inpainting with deep generative models. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 6882–6890. IEEE Computer Society, 2017.

[41] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang. Generative image inpainting with contextual attention. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018* [1], pages 5505–5514.

[42] H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang, X. Huang, and

D. Metaxas. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 5908–5916, Oct 2017.

[43] Z. Zhang, Y. Song, and H. Qi. Age progression/regression by conditional adversarial autoencoder. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 4352–4360. IEEE Computer Society, 2017.

[44] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, 2017.