Image Animation with Perturbed Masks

Yoav Shalev Lior Wolf
Blavatnik School of Computer Science, Tel Aviv University
yoavshalev@mail.tau.ac.il, wolf@cs.tau.ac.il

Abstract

We present a novel approach for image-animation of a source image by a driving video, both depicting the same type of object. We do not assume the existence of pose models and our method is able to animate arbitrary objects without the knowledge of the object's structure. Furthermore, both, the driving video and the source image are only seen during test-time. Our method is based on a shared mask generator, which separates the foreground object from its background, and captures the object's general pose and shape. To control the source of the identity of the output frame, we employ perturbations to interrupt the unwanted identity information on the driver's mask. A mask-refinement module then replaces the identity of the driver with the identity of the source. Conditioned on the source image, the transformed mask is then decoded by a multi-scale generator that renders a realistic image, in which the content of the source frame is animated by the pose in the driving video. Due to the lack of fully supervised data, we train on the task of reconstructing frames from the same video the source image is taken from. Our method is shown to greatly outperform the state-of-the-art methods on multiple benchmarks. Our code and samples are available at https://github.com/itsyoavshalev/Image-Animation-with-Perturbed-Masks.

1. Introduction

The ability to reanimate a still image based on a driving video has been extensively studied in recent years [12, 15, 20]. The developed methods achieve an increased degree of accuracy in both, maintaining the source identity, as extracted from the source frame, and in replicating the motion pattern of the driver's frame. In addition, the recent methods also show good generalization to unseen identities and are relatively robust, and have fewer artifacts than the older methods. The relative ease with how these methods can be applied out-of-the-box has led to their adoption in various visual effects.

Interestingly, some of the most striking results have been obtained with model-free methods, i.e., that do not rely, for example, on post-extraction models [10, 11, 16, 17, 22, 25]. This indicates that such methods can convincingly disentangle shape and identity from motion [7, 13].

There are, however, a few aspects in which such methods still need to improve. First, the generated videos are with noticeable artifacts. Second, some of the identity of the source image is lost and replaced by identity elements from the driving video. Third, the animation of the generated video does not always match the motion in the driver video.

Here, we propose a method that is preferable to the existing work in terms of motion accuracy, identity and background preservation, and quality of the generated video. Our method relies on a mask-based representation of the driving pose and explicit conditioning on the source foreground mask. Source and driver masks are extracted by the same network. The driver mask goes through an additional stage that replaces the identity information in the mask.

The reliance on masks has many advantages. First, it eliminates many of the identity cues from the driving video. Second, it explicitly models the region that needs to be replaced in the source image. Third, it is common to both source and driver, thus allowing, with proper augmentation, to train only on source videos. Fourth, it captures a detailed description of the object's pose and shape.

Interestingly, unlike many of the previous methods, we do not rely on GANs [9] to generate proper outputs from combinations of different inputs. Instead, we employ an encoder-decoder, in which the identity is manipulated in order to direct the networks toward employing specific parts of the information from each input. To summarize, our contributions are: (i) An image animation method that generalizes to unseen identities of the same type, and is able to animate arbitrary objects better than previous work; (ii) Innovative use of perturbations over masks, in order to interrupt the driver's identity, which is then replaced with the source's identity by the mask refinement module; (iii) A comprehensive evaluation of several different applications, which show a sizable improvement over the current image animation state-of-the-art.

2. Related Work

Much of the work on image animation relies on prior information on the animated object, in the form of explicit modeling of the object's structure, e.g., some methods animate a source image using facial landmarks [27, 28], while [15] developed a human-pose-guided image generator. However, in many applications, an explicit model is not available. Our method is model-free and able to animate arbitrary objects.

There are many model-free contributions in the field of image-to-image translation, where an image of one domain is mapped to an analog image of another domain. [11] learns a map between two domains using a conditional GAN. [22] developed a multi-scale GAN that generates high-resolution images from semantic label maps. [10] encodes images of both domains into a shared content space and a domain-specific style space. The content code of one domain is combined with the style code of the other domain, and then the image is generated using a domain-specific decoder. For this class of methods, the model is not able to generalize to other unseen domains of the same category without retraining. In contrast, for a given type of model (e.g. faces), our method is trained once, and able to generalize to unseen domains of the same type (e.g. the source and driving faces can be of any identity).

More related to our method is a method that assumes a reference frame for each video, and learns a dense motion field that maps pixels from a source frame to its reference frame, and another mapping from the reference frame to the driver's frame [25]. [16] extracts landmarks for driving and source images of arbitrary objects, and generates motion heatmaps from the key-points displacements. The heatmaps and the source image are then processed to generate the final prediction.

A follow-up work [17] extracts a first-order motion representation, consisting of sparse key-points and local affine transformations, with respect to a reference frame. The motion representation is then processed to generate a dense motion field, from the driver's frame to the source's, and occlusion maps to mask out regions that should be inpainted by the generator. This method, like ours, does not employ GANs. The main differences are that our method does not assume a reference frame, instead of key-points, we generate objects masks, which are more informative regarding pose and shape, and our innovative use of perturbations and the mask refinement module.

Other methods, including [7, 13], learn a part-based disentangled representation of shape and appearance, and try to ensure that local changes in appearance and shape remain local, and do not affect the overall representation. On the other hand, our method does not assume a predefined number of parts, and by using perturbations and the mask refinement module, it is able to better remove the driver's identity and inject that of the source.

When a source video is available, video-to-video translation methods [4, 12] may be used for motion transfer, by utilizing the rich appearance and pose information of the source video. Such methods learn a mapping between two domains and are able to generate realistic results, where the source video is animated by the driver video. These methods require a large number of source frames at train time, and require a long training process for every target subject. In contrast, our model is able to animate a single source image, which is unseen during training, and employs a driving video with another novel person.

3. Method

The method consists of four encoder-decoder networks: the mask generator m, the mask refinement network r, and the low and high-resolution frame generators ℓ and h. The networks transform a source frame $\boldsymbol{s}$ and a driving frame $\boldsymbol{d}$ into the generated high-resolution frame $\boldsymbol{f}$, where $\boldsymbol{f}$ contains the foreground and background of the source frame $\boldsymbol{s}$, such that the pose of the foreground object in $\boldsymbol{s}$ is modified to match that of the driver frame $\boldsymbol{d}$. This is done for each driving frame separately, and at test-time executed through the following process, as depicted in Fig. 1:

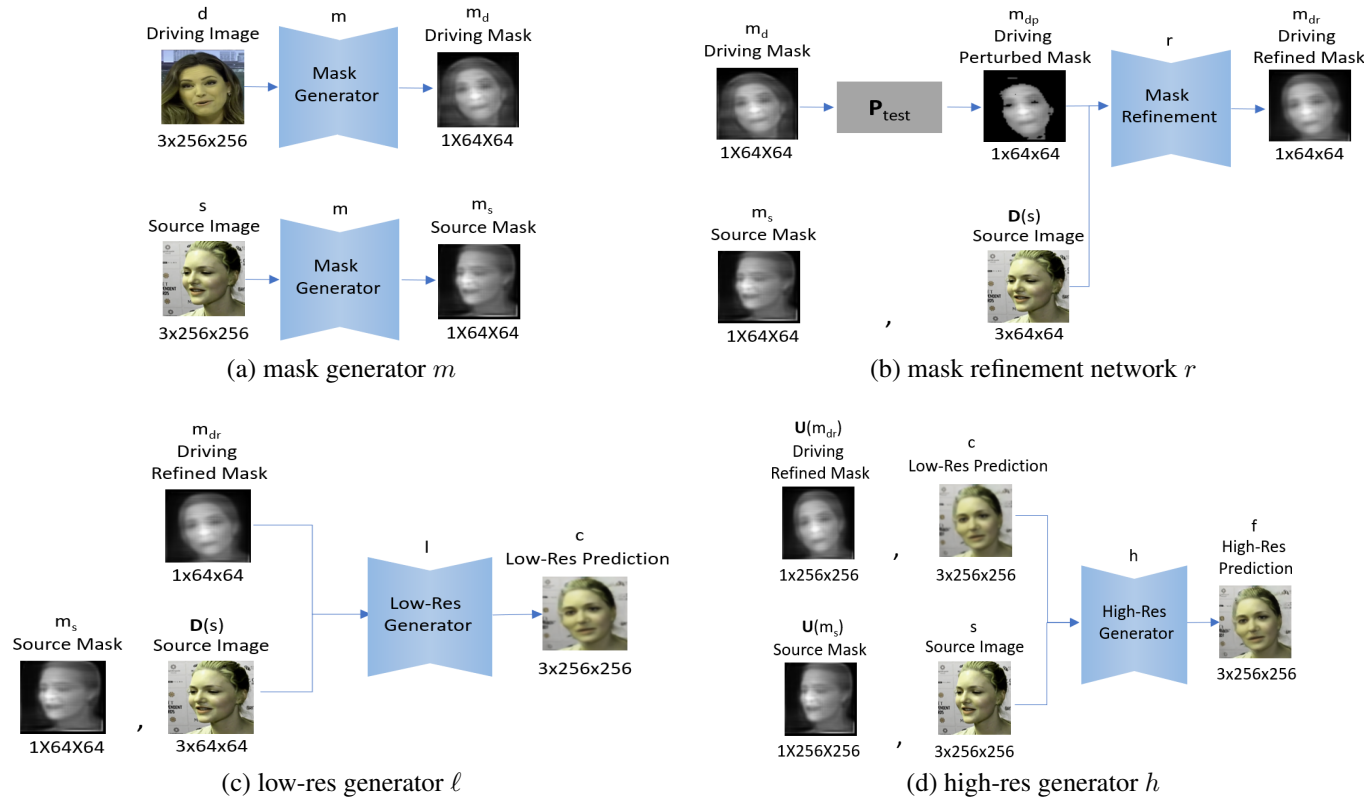
$$\boldsymbol{m}_s = m(\boldsymbol{s}) \tag{1}$$
$$\boldsymbol{m}_d = m(\boldsymbol{d}) \tag{2}$$
$$\boldsymbol{m}_{dp} = \mathbf{P}_{\text{test}}(\boldsymbol{m}_d) \tag{3}$$
$$\boldsymbol{m}_{dr} = r(\mathbf{D}(\boldsymbol{s}), \boldsymbol{m}_s, \boldsymbol{m}_{dp}) \tag{4}$$
$$\boldsymbol{c} = \ell(\mathbf{D}(\boldsymbol{s}), \boldsymbol{m}_s, \boldsymbol{m}_{dr}) \tag{5}$$
$$\boldsymbol{f} = h(\boldsymbol{s}, \mathbf{U}(\boldsymbol{m}_s), \mathbf{U}(\boldsymbol{m}_{dr}), \boldsymbol{c})\,, \tag{6}$$

where upper-cased bolded notations represent untrained operations, including $\mathbf{D}$ ($\mathbf{U}$), which is a downscale (upscale) operator, implemented using a bi-linear interpolation, that transforms an image of resolution 256×256 to an image of resolution 64×64 (or vice versa).

First, $\boldsymbol{m}_s$ and $\boldsymbol{m}_d$ are generated using the mask generator m. Next, the identity-perturbation operator $\mathbf{P}_{\text{test}}$ is applied on the driver's mask $\boldsymbol{m}_d$, by setting to zero pixels that are smaller than a threshold ρ. Considering typical face masks, e.g., the pixels in the areas of the eyes, mouth, and hair are with low intensities. Removing these pixels by applying $\mathbf{P}_{\text{test}}$, results in a much more generic face, interrupting the driver's identity. For each driver's mask, we set the threshold ρ to be the median pixel value.

Next, the refinement network r acts to generate the missing data of the perturbed mask $\boldsymbol{m}_{dp}$ and to replace the driver's identity with that of the source. It uses the source's frame and mask as a reference.

Finally, the generated frame is being synthesized in a hierarchical process in which the coarse (low resolution)



(a) mask generator m

(b) mask refinement network r

(c) low-res generator ℓ

(d) high-res generator h

Figure 1. Overview of our method at test time. (a) Source and driving masks $\boldsymbol{m}_s$ and $\boldsymbol{m}_d$ are generated using the mask generator m. (b) The identity-perturbation operator $\mathbf{P}_{\text{test}}$ is then applied to the driver's mask, and along with a scaled-down version of the source's image $\mathbf{D}(\boldsymbol{s})$ and the source's mask $\boldsymbol{m}_s$, they are fed into the mask refinement network r, to generate the driver's refined mask $\boldsymbol{m}_{dr}$. (c) The refined mask $\boldsymbol{m}_{dr}$, the source's mask $\boldsymbol{m}_s$, and the scaled-down source's image $\mathbf{D}(\boldsymbol{s})$ are fed into the generator ℓ, which generate the initial prediction $\boldsymbol{c}$. (d) The scaled-up refined mask $\mathbf{U}(\boldsymbol{m}_{dr})$, the source image $\boldsymbol{s}$, the initial prediction $\boldsymbol{c}$, and the scaled-up source's mask $\mathbf{U}(\boldsymbol{m}_s)$ are fed into the generator h, in order to generate the final prediction $\boldsymbol{f}$.

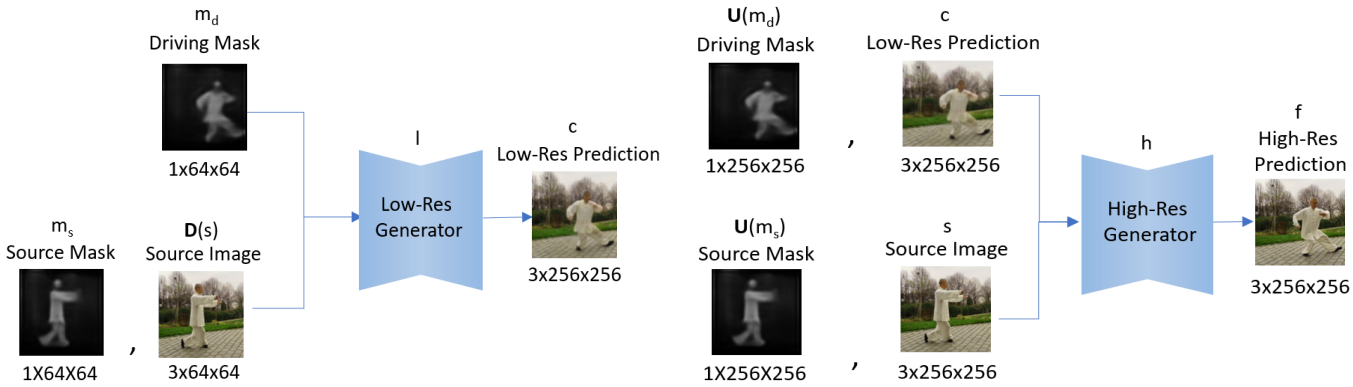
frame $\boldsymbol{c}$ is first generated using ℓ and is then refined by the network h. Both generators (ℓ, h) utilize the mask $\boldsymbol{m}_s$ to attend the foreground and background objects in the source frame $\boldsymbol{s}$, and to infer the occluded regions that need to be generated.

The refined driver's mask $\boldsymbol{m}_{dr}$ is the only conditioning on the frame generation process that stems from the driver's frame $\boldsymbol{d}$. It, therefore, needs to encode the pose of the foreground object in the driving frame. However, this has to be done in a way that is invariant to the driver's identity. For example, when reanimating person A based on a driver video of person B, the pose of B should be given, while discarding the body shape information of B. Otherwise, the generated frame could have the appearance of the source's foreground and a body shape that mixes that of the person in the source frame and that of the person in the driving frame. The perturbation operator $\mathbf{P}_{\text{test}}$ is, therefore, designed to interrupt the elements that are associated with the driver's identity, which encourages the refinement network r to project identity elements from the reference mask $(\boldsymbol{m}_s)$ and frame $(\boldsymbol{s})$. As a result, the proposed identity replacement stage does not modify the general pose of the driver's mask, but only replaces the driver's identity.

3.1. Training

Training is conducted using driving and source frames from the same video. The reason is that for the type of supervised loss terms we use, a ground-truth target frame is required. The main challenge is to keep the model robust enough for accepting at test time a driving frame $\boldsymbol{d}$ from another video.

The training pipeline is slightly modified from test time, in that an augmentation $\mathbf{A}$ is applied to the driving frame $\boldsymbol{d}$, and that a more elaborate perturbation $\mathbf{P}_{\text{train}}$ takes a place. In addition, since the source and driving frames are of the same identity, as shown in Fig. 2 both generators ℓ and h are using the driver's mask $\boldsymbol{m}_d$, instead of using the refined mask $\boldsymbol{m}_{dr}$, which is used only for training the refinement



(a) low-res generator ℓ (b) high-res generator h

Figure 2. The low-res and high-res generators at train time. Instead of getting the driver's refined mask $\boldsymbol{m}_{dr}$ as in test time, the two generators ℓ and h are using the driver's mask $m(\mathbf{A}(\boldsymbol{d}))$ and its up-scaled version $\mathbf{U}(m(\mathbf{A}(\boldsymbol{d})))$, respectively.

network r:

$$\boldsymbol{m}_d = m(\mathbf{A}(\boldsymbol{d})) \tag{7}$$

$$\boldsymbol{m}_{dp} = \mathbf{P}_{\text{train}}(\boldsymbol{m}_d) \tag{8}$$

$$\boldsymbol{m}_{dr} = r(\mathbf{D}(\boldsymbol{s}), \boldsymbol{m}_s, \boldsymbol{m}_{dp}) \tag{9}$$

$$\boldsymbol{c} = \ell(\mathbf{D}(\boldsymbol{s}), \boldsymbol{m}_s, \boldsymbol{m}_d) \tag{10}$$

$$\boldsymbol{f} = h(\boldsymbol{s}, \mathbf{U}(\boldsymbol{m}_s), \mathbf{U}(\boldsymbol{m}_d), \boldsymbol{c})\,, \tag{11}$$

where augmentation $\mathbf{A}$ is a color transformation that scales the input's brightness, contrast, and saturation by a random value drawn from $[0.9, 1.1]$, and shifts its hue by a random value drawn from $[-0.1, 0.1]$. The goal of this augmentation is to encourage the generated masks to be invariant to the input's appearance, despite the challenge mentioned above of training on frames from the same video.

$\boldsymbol{P}_{\text{train}}$ performs the following two steps sequentially: (i) breaks the image vertically (horizontally) into six parts, and scales each part horizontally (vertically) by a random value drawn from $[0.75, 1.25]$. Next, it scales the entire output vertically (horizontally), by a random value drawn from $[0.75, 1.25]$. (ii) similarly to $\boldsymbol{P}_{\text{test}}$, sets to zero pixels that are smaller than a threshold value ρ, which we set to be the median pixel value of each mask. The goal of the resizing operation is to interrupt the driver's identity by modifying the proportions of its mask, e.g., in faces, it modifies the distance between the eyes, which results in an identity perturbation, while keeping the general pose. The thresholding operation eliminates low-intensity pixels (e.g. the boundary of the body, and the hair, eyes, and mouth areas), which are a major ingredient of the driver's identity. Without the listed operations of $\boldsymbol{P}_{\text{train}}$, we experienced a phenomenon where the mask refinement module r ignores the reference mask ($\boldsymbol{m}_s$) and frame ($\boldsymbol{s}$), i.e. applying $\boldsymbol{P}_{\text{train}}$ encourage the mask refinement network r to project the elements that are associated with the source's identity, which is crucial for the generation part.

All hyper-parameters values, including these constants, were selected using cross-validation and fixed throughout all experiments on all benchmarks.

Loss Terms The model is trained end-to-end using two loss terms: a mask refinement loss and a perceptual reconstruction loss. At train time, where the source and driving frames are of the same identity, the role of the mask refinement network r is to recover the missing data that was removed by the operator $\mathbf{P}_{\text{train}}$. Therefore, we minimize the L_1 loss of the driver's mask $\boldsymbol{m}_d$ and its refined mask $\boldsymbol{m}_{dr}$:

$$\mathcal{L}_{\text{mask}}(\boldsymbol{d}) = L_1(\boldsymbol{m}_{dr}, \boldsymbol{m}_d)\,. \tag{12}$$

For the image reconstruction loss of the generators ℓ and h, following [17] and based on the implementation of [21], we minimize a perceptual loss using the pre-trained weights of a VGG-19 model. For two images $\boldsymbol{a}$ and $\boldsymbol{b}$, the reconstruction loss terms using the j^{th} layer of the pre-trained VGG model are written as:

$$\mathcal{L}_{\text{VGG}}(\boldsymbol{a}, \boldsymbol{b})_j = \text{AVG}(|\mathbf{N}_j(\boldsymbol{a}) - \mathbf{N}_j(\boldsymbol{b})|) \tag{13}$$

where AVG is the average operator and $\mathbf{N}_j(\cdot)$ are the features extracted using the j^{th}-layer of the pre-trained VGG model. For the coarse and fine predictions $\boldsymbol{c}$ and $\boldsymbol{f}$, and a driving frame $\boldsymbol{d}$, we compute the following reconstruction loss for multiple resolutions:

$$\mathcal{L}_{\text{reconstruct}} = \sum_s \sum_j \mathcal{L}_{\text{VGG}}(\boldsymbol{c}_{\boldsymbol{s}}, \boldsymbol{d}_{\boldsymbol{s}})_j + \mathcal{L}_{\text{VGG}}(\boldsymbol{f}_{\boldsymbol{s}}, \boldsymbol{d}_{\boldsymbol{s}})_j$$

where input image $\boldsymbol{a}_{\boldsymbol{s}}$, has a resolution $\boldsymbol{s} \in [256^2, 128^2, 64^2]$. We use the first, third, and fifth ReLU layers of the VGG-19 model. Note that while VGG was designed for a resolution of 224^2, the first layers are convolutional, and can be used for an arbitrary input scale.

The combined loss is given by $\mathcal{L} = \lambda_1 \mathcal{L}_{\text{mask}} + \lambda_2 \mathcal{L}_{\text{reconstruct}}$, for weight parameters $\lambda_1 = 100$ and $\lambda_2 = 10$.

To avoid unwanted adaptation of the network m, the backpropagation of $\mathcal{L}_{\text{mask}}$ only updates the weights of the mask refinement network r. When backpropagating the second part of the reconstruction loss $\sum_s \sum_j \mathcal{L}_{\text{VGG}}(\boldsymbol{f_s}, \boldsymbol{d_s})_j$, only the generator h is updated. The Adam optimizer is employed with a learning rate of 2×10^{-4} and β values of 0.5 and 0.9. The batch size is 16. Following [17], we decay the learning rate at epochs 60 and 90, running for 100 epochs on NVIDIA Titan RTX. The mask refinement network r starts training after we complete the first training epoch, when the outputs of the mask generator m start to be meaningful.

The architecture of the networks is given in the supplementary materials, which also contain the source code.

4. Experiments

The training and evaluation were done using three different datasets, containing short videos of diverse objects. **Tai-Chi-HD** is a dataset containing videos of people doing tai-chi exercises. Following [17], 3,141 tai-chi videos were downloaded from YouTube. The videos were cropped and resized to a resolution of 256^2, while preserving the aspect ratio. There are 3,016 training videos and 125 test videos. **VoxCeleb** is an audio-visual dataset consist of short videos of talking faces, introduced by [14]. VoxCeleb1 is the collection used, and as pre-processing, bounding boxes of the faces were extracted and resized to 256^2, while preserving the aspect ratio. It contains an overall number of 18,556 training videos and 485 test videos. The **BAIR** dataset contains videos of Sawyer robotic arms interacting with objects [8]. It contains 42,880 training videos and 128 test videos, where each video consists of 30 frames with a resolution of 256^2. We were unable to obtain the UvA-NEMO dataset [6], which was utilized in some earlier contributions.

We borrow and significantly expand the evaluation process of [17]. Our method is evaluated quantitatively and qualitatively for the tasks of both video reconstruction and image animation, where the source and driving videos are of different identities. Additionally, despite being model-free, we compare to model-based methods in the few-shot-learning scenario. In this case, our method, unlike the baseline methods, does not employ any few shot samples.

Multiple metrics are used for evaluation: **L1** is the L1 distance between the generated and ground-truth videos. **Average Key-points Distance (AKD)** measures the average distance between the key-points of the generated and ground-truth videos. For Tai-Chi-HD, we use the human-pose estimator of [3], and for VoxCeleb, we use the facial landmark detector of [2]. **Missing Key-points Rate (MKR)** measures the percentage of key-points that were successfully detected in the ground-truth video, but were missing in the generated video. The human-pose estimator of [3] outputs for every keypoint an indicator of whether it was successfully detected. Using this indicator, we measure MKR for the Tai-Chi-HD dataset. **Average Euclidean Distance (AED)** measures the average Euclidean distance in some embedding space between the representations of the ground-truth and generated videos. Following [17], we employ the feature embedding of [16]. **Structural Similarity (SSIM)** [24]: For VoxCeleb, we compare the structural similarity of the ground-truth driving frames and generated images. **Cosine Similarity (CSIM)**: For VoxCeleb, we measure the identity similarity of the generated and ground-truth source faces, by comparing the cosine similarity of embedding vectors generated by a face recognition network [5]. **Classification (CLS)**: For Tai-Chi-HD, we classify the generated frames using the Detectron2 framework [26], and measure the number of frames classified as a person. Specifically, we use the X101-FPN COCO instance-segmentation model. **Intersection Over Union (IOU)**: For Tai-Chi-HD, we calculate the IOU of the segmentations of the generated and driving videos. The segmentations are generated using the same model we use for classification. **Facial Expression Similarity (FES)**: For VoxCeleb, we measure the facial expression similarity of generated and driving frames using the FER classifier (https://github.com/justinshenk/fer), which supports seven different emotions.

4.1. Video Reconstruction

The video reconstruction benchmarks follow the training procedure in that the source and target frames are from the same video. For evaluation, the first frame of a test video is used as the source frame, and the remaining frames of the same video as the driving frames. The goal is to reconstruct all the frames of the test video, except the first.

L1, AKD, MKR, and AED are compared with the state-of-the-art model-free methods, including X2Face of [25], MonkeyNet of [16], and the method suggested by [17], which we refer to as FOMM. The results are reported in Tab. 1. Evidently, our method outperforms the baselines for each of the datasets and all metrics by a significant margin, except for the AKD measure on the VoxCeleb dataset, where accuracy was decreased by 2.7%. The most significant improvement is for the Tai-Chi-HD dataset, which is the most challenging dataset, because it consists of diverse movements of a highly non-rigid body.

In order to verify that the improvement over the baselines is not due to their smaller bottleneck size, we re-trained FOMM and MonkeyNet on all three datasets, using a wider bottleneck, and evaluated the video reconstruction task. We used 365 key-points for FOMM, which are equivalent to 2190 floating-point numbers, and 440 key-points for MonkeyNet, which are equivalent to 2200 floating-point numbers. As reported in Tab. 2, we saw no improvement.

Next, we follow [28] and compare SSIM and CSIM with X2Face, Pix2PixHD [23], and the FSAL method [28].

	Tai-Chi-HD				*VoxCeleb*			*BAIR*
Method	L1	AKD	MKR	AED	L1	AKD	AED	L1
X2Face	0.080	17.654	0.109	0.272	0.078	7.687	0.405	0.065
MN	0.077	10.798	0.059	0.228	0.049	1.878	0.199	0.034
FOMM	0.063	6.862	0.036	0.179	0.043	**1.294**	0.140	0.027
Ours	**0.047**	**4.239**	**0.015**	**0.147**	**0.034**	1.329	**0.130**	**0.021**

Table 1. Video reconstruction results. MN=Monkey-Net.

	Tai-Chi-HD				*VoxCeleb*			*BAIR*
Method	L1	AKD	MKR	AED	L1	AKD	AED	L1
MN	– The wider bottleneck model diverged –							
FOMM	0.068	8.561	0.043	0.196	0.050	1.525	0.165	0.028
Ours	**0.047**	**4.239**	**0.015**	**0.147**	**0.034**	**1.329**	**0.130**	**0.021**

Table 2. Video reconstruction using a wider bottleneck for baselines. MN=MonkeyNet

Method	#FT	SSIM ↑	CSIM ↑
X2Face	1/8/32	0.68/0.73/0.75	0.16/0.17/0.18
P2PHD	1/8/32	0.56/0.64/0.70	0.09/0.12/0.16
FSAL	1/8/32	0.67/0.71/0.74	0.15/0.17/0.19
Ours	**0**	**0.80**	**0.70**

Table 3. Few-shot learning results for VoxCeleb. Unlike baselines, we do not perform identity fine-tuning. #FT=number of frames used for finetuning. P2PHD=Pix2PixHD.

The baselines are evaluated in the few-shot-learning setting, where models are fine-tuned on a set of size #FT, consisting of frames of a person that was not seen during the initial meta-learning step. After the fine-tuning step, the evaluation is done on a hold-out set, consisting of unseen frames of the same person. The evaluation is done for VoxCeleb and the results are reported in Tab. 3. As can be seen, our method generalizes better and outperforms the baselines in SSIM and even more so in CSIM. This is especially indicative of the method's capabilities, since (i) we skip the fine-tuning step for our model (in our case #FT $= 0$), and (ii) X2Face and FSAL were designed specifically for faces, while our method is model-free and generic.

4.2. Image Animation

The task of image animation is to animate a source image using a driving video. The object and its background in the source and driving inputs may have different identities and appearances. In the experiments, the first frame of a source video is used for encoding the appearance, and all frames of the driving video are used for driving the object's motion. A video is generated where the content of the source frame is animated by the driving video.

To evaluate the alignment between the generated and driving videos, we measure AKD, MKR, and IOU for the Tai-Chi-HD dataset, and FES and CSIM for the VoxCeleb dataset. AKD, MKR, and IOU are irrelevant for the VoxCeleb dataset, because a perfect match may indicate an identity loss. The reason is that the facial key-points and segmentations of different people have different ratios, and therefore cannot be compared. This is not the case for the Tai-Chi-HD dataset, where the camera is far from the person, and the body proportions are almost identical across different identities. Measuring CLS for the Tai-Chi-HD dataset provides differentiation, while for VoxCeleb, our method and FOMM are both almost 100% accurate, and the improvement we present is negligible. Measurements are not available for the Bair dataset, due to the lack of a pre-trained classifier and a keypoint detector for the Sawyer robotic arm. For the following experiments, 100 pairs with different identities were randomly selected from the test set of each dataset. The quantitative animation results are presented in Tab. 4 and in Tab. 5. As can be seen, our method is better for all metrics by a significant margin. See CSIM analysis for the ablation models in section 4.2.

	Tai-Chi-HD				*VoxCeleb*
Method	AKD↓	MKR↓	CLS↑	IOU↑	FES↑
X2Face	22.799	0.140	0.870	0.558	28.0%
MonkeyNet	17.308	0.104	0.852	0.634	38.2%
FOMM	10.218	0.044	0.957	0.864	48.4%
Ours	**7.809**	**0.020**	**0.994**	**0.875**	**52.2%**

Table 4. Quantitative evaluation for image animation.

To evaluate the robustness for different levels of changes in pose, between source and driving frames, we extend the AKD, MKR, and FES experiments. Based on the AKD score between source and driving frames, we split the test set into three sub-sets, where the first sub-set contains the frames with the lowest score, and so on. We compare to FOMM, the most competitive method, and report the results in Tab. 6. As can be seen, our method better preserves the driver's pose and expression, even for large changes.

Sample results compared to the baseline methods are shown in Fig. 3. For VoxCeleb, our method better preserves the identity of the source, and the facial expressions of the generated frames are more compatible with that of the driver. For the Tai-Chi-HD dataset, the baseline methods tend to generate infeasible poses for the fourth generated frame, while we do not. Unlike FOMM, we well maintain environment elements, such as the stick on the top-right of the generated frame. For the BAIR dataset, the images generated by our method are the sharpest, and it is the only method that places the generated object in the right position. Note that the samples were selected to match those of [16], and not by us.

Ablation The main challenge is to replace the identity on the driver's mask with that of the source while keeping on the driver's pose. We do that in two steps: (i) the driver's identity is interrupted by applying $\mathbf{P}_{\text{test}}$, (ii) the refinement

Model	X2Face	MN	FOMM	no_pert	no_ref	no_id	low_res	Ours
OpenFace [1]	0.512	0.544	0.620	0.625	0.487	0.522	0.632	**0.642**
DeepFace [19]	0.528	0.580	0.646	0.648	0.515	0.546	0.658	**0.676**
DeepID [18]	0.799	0.827	0.953	0.917	0.756	0.786	0.948	**0.963**

Table 5. CSIM for VoxCeleb, including the ablation models.

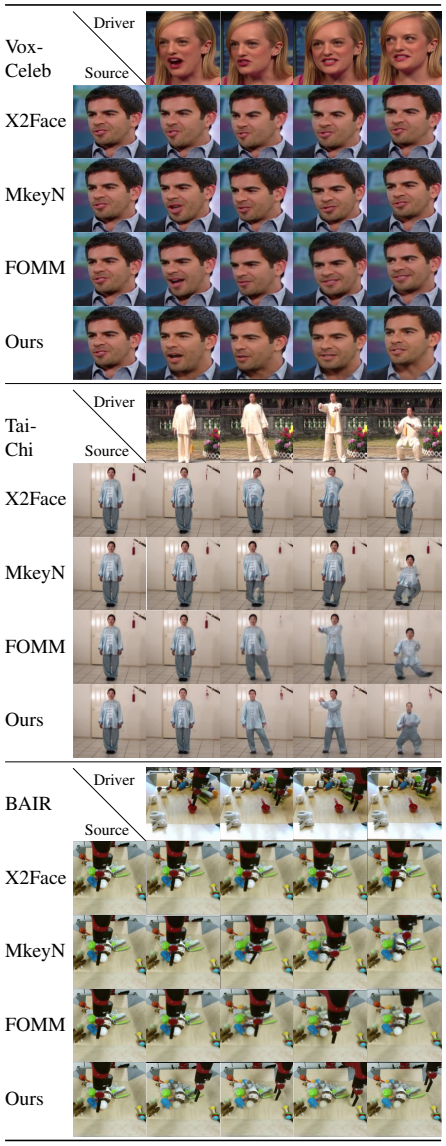


Figure 3. Sample animation results on the three datasets. We use the same samples as evaluated by FOMM.

network r acts to replace the driver's identity with that of the source. To evaluate the roles of $\mathbf{P}_{\text{test}}$ and r, we evaluate three partial methods: no_pert, no_ref, and no_id, where the first, second, or both steps are removed, respectively.

Ablation and intermediate results generated by our pipeline are shown in Fig. 4. As can be seen, the generated masks $\boldsymbol{m}_s$ and $\boldsymbol{m}_d$ capture very accurately the object's pose and shape, and the mask refinement network r successfully

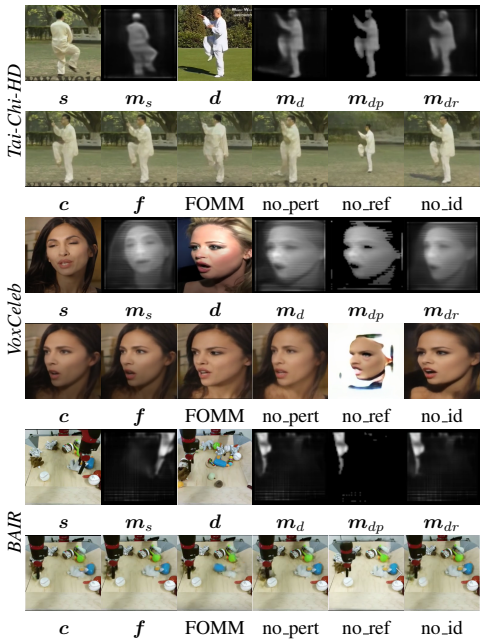


Figure 4. Intermediate results generated by our method. The generated frame $\boldsymbol{f}$ is compared to FOMM and ablation models. From left to right: the source frame $\boldsymbol{s}$, source mask $\boldsymbol{m}_s$, the driving frame $\boldsymbol{d}$, the driving mask $\boldsymbol{m}_d$, the perturbed driving mask $\boldsymbol{m}_{dp}$, the refined driving mask $\boldsymbol{m}_{dr}$, the low-res prediction $\boldsymbol{c}$, the high-res prediction $\boldsymbol{f}$, FOMM's result, and the ablations: no_pert drops $\mathbf{P}_{\text{test}}$, no_ref omits the mask refinement r, and no_id omits both.

apply the source's identity to the driver's mask. Comparing the generated frame $\boldsymbol{f}$ to that of FOMM, we notice that for the Tai-Chi-HD dataset, the pose of the generated body using our method is much more compatible with that of the driver's, where FOMM's model generates a distorted body.

For VoxCeleb, using our method, the identity of the source is better preserved, as it also reveals a small portion of the teeth, as the driver does. For the BAIR dataset, unlike FOMM, our method was able to inpaint the occluded surface, including the white and blue items on the right of the generated frame. Examining the generated frames of the ablation models shows that both steps, identity-perturbation and mask refinement, are critical. The frames generated by no_pert and no_id have significant traces of the driver's identity. This is especially clear for VoxCeleb, on the forehead area of no_pert and the general appearance for no_id. Similarly, for Tai-Chi-HD, the frame generated by no_ref contains traces from the driver's environment, and for the other datasets, it generates distorted results.

Next, we evaluated the following ablation models. *no_color_aug*, where the color augmentation is eliminated at train time. *h_update_l*, where the high-res generator h keeps updating the weights of the low-res generator ℓ. *h_update_m*, where the high-res generator h keeps updating

Method	AKD ↓	MKR↓	FES ↑
FOMM	10.218\8.629\9.958\12.364	0.044\0.042\0.042\0.049	48.4% \48.9% \50.7% \45.6%
Ours	**7.809\6.431\7.433\8.909**	**0.020\0.017\0.020\0.025**	**52.2% \54.3% \53% \49.3%**

Table 6. AKD and MKR for Tai-Chi-HD. FES for VoxCeleb. All are reported for the Full\1st \2nd \3rd sets.

Method	L1	AKD	AED
No_color_aug	0.045	1.863	0.159
H_update_m	0.041	1.829	0.161
H_update_l	0.039	1.412	0.142
Full method	**0.034**	**1.329**	**0.130**

Table 7. Ablation analysis on the reconstruction task for VoxCeleb.

Dataset	X2Face	MN	FOMM	no_pert	no_ref	no_id	Ours
Tai-Chi	(0%,0%)	(4%,4%)	(16%,8%)	(2%,2%)	(0%,0%)	(0%,0%)	**(78%, 86%)**
VoxCeleb	(0%,0%)	(6%,4%)	(10%,10%)	(12%,10%)	(0%,0%)	(0%,0%)	**(72%,76%)**
BAIR	(0%,4%)	(6%,6%)	(14%,8%)	(20%,16%)	(0%,0%)	(0%,0%)	**(60%, 66%)**

Table 8. The ratio (Quality, Motion-fidelity) of best videos selected for each method, including ablations. MN=Monkey-Net.

the weights of the mask generator m. The ablation models were trained on the VoxCeleb dataset and evaluated on the video reconstruction task. Results are presented in Tab. 7. As can be seen, using the color augmentation and limiting the task of the high-res generator h for adding fine details, helps the model to converge faster.

Next, we analyze the importance of the suggested modules for identity preservation, using the CSIM between source and generated frames. The results are shown in Tab. 5. As can be seen, removing the refinement step (no_ref, no_id) dramatically degrades the CSIM score, and applying $\mathbf{P}_{\text{test}}$ helps r to better inject the source's identity. It is also can be seen that low_res results in a lower CSIM score, which verifies the effectiveness of h.

In Fig. 5 we show an example for the visual improvement of $\boldsymbol{f}$ over $\boldsymbol{c}$. The environment in both examples and the face of the man in the left example are much sharper in $\boldsymbol{f}$. In addition, we show an example where the generated masks reflects very well whether the subject is facing back or not.

User study To further qualitatively evaluate our method and compare it with existing work, we presented volunteers with a source image, a driving video, and four randomly ordered generated videos, one for each baseline method. They were asked to (i) select the most realistic animation of the source image, and (ii) select the video with the highest fidelity to the driver video. For each of the $n = 25$ participants, we repeated the experiment three times, each time using a different dataset and a random test sample.

The results, see Tab. 8, are highly consistent with the quantitative results, and indicate that the quality and the animation of the videos our method generated, contain fewer artifacts and are better synchronized with the driver videos. In addition, it can be seen that the refinement network r is the most important module for quality and for motion, and that the perturbation operator $\mathbf{P}_{\text{test}}$ is much more needed in Tai-Chi-HD and VoxCeleb.

Figure 5. (left) $\boldsymbol{f}$ is sharper than $\boldsymbol{c}$. (right) back & front masks.

Limitations While better than the baselines, there are artifacts and identity loss for extreme changes in pose and shape. Additionally, since the perturbation operator sets to zero mask's pixels that are smaller than a threshold ρ, some pose information may be lost. Other failure cases are ambiguities on generated masks, e.g. for Tai-Chi-HD, when the hands are overlapped, the generator may struggle to understand which one is on top. This limitation also exists for the baseline methods including key-points methods.

As a video generation method, considerations should be made towards the possible use of the generated output in a harmful way. For example, the generated videos of talking heads can be used as part of a system for manipulating speech content. Our hope is that studying such methods in an open way would enable the mitigation of such risks through better detection methods and by raising awareness.

5. Conclusions

A novel method for conditionally reanimating a frame is presented. It utilizes a masking mechanism for encoding pose information. Our method is able to effectively extract both the source and the driving masks, while accurately capturing the shape and foreground/background separation, and recovering an identity-free pose representation of the driver. Our results outperform the state of the art by a sizable margin on the available benchmarks.

Acknowledgments

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant ERC CoG 725974). The contribution of the first author is part of a PhD thesis research conducted at Tel Aviv University.

References

[1] Brandon Amos, Bartosz Ludwiczuk, and Mahadev Satyanarayanan. Openface: A general-purpose face recognition library with mobile applications. Technical report, CMU-CS-16-118, CMU School of Computer Science, 2016. 7

[2] Adrian Bulat and Georgios Tzimiropoulos. How far are we from solving the 2D & 3D Face Alignment problem? (and a dataset of 230,000 3D facial landmarks). *arXiv e-prints*, page arXiv:1703.07332, Mar. 2017. 5

[3] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. *arXiv e-prints*, page arXiv:1611.08050, Nov. 2016. 5

[4] Caroline Chan, Shiry Ginosar, Tinghui Zhou, and Alexei A. Efros. Everybody Dance Now. *arXiv e-prints*, page arXiv:1808.07371, Aug. 2018. 2

[5] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4690–4699, 2019. 5

[6] Hamdi Dibeklioğlu, Albert Ali Salah, and Theo Gevers. Are you really smiling at me? spontaneous versus posed enjoyment smiles. In *European Conference on Computer Vision*, pages 525–538. Springer, 2012. 5

[7] Aysegul Dundar, Kevin J. Shih, Animesh Garg, Robert Pottorf, Andrew Tao, and Bryan Catanzaro. Unsupervised Disentanglement of Pose, Appearance and Background from Images and Videos. *arXiv e-prints*, page arXiv:2001.09518, Jan. 2020. 1, 2

[8] Frederik Ebert, Chelsea Finn, Alex X. Lee, and Sergey Levine. Self-Supervised Visual Planning with Temporal Skip Connections. *arXiv e-prints*, page arXiv:1710.05268, Oct. 2017. 5

[9] Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. *Deep learning*, volume 1. MIT Press, 2016. 1

[10] Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. Multimodal Unsupervised Image-to-Image Translation. *arXiv e-prints*, page arXiv:1804.04732, Apr. 2018. 1, 2

[11] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-Image Translation with Conditional Adversarial Networks. *arXiv e-prints*, page arXiv:1611.07004, Nov. 2016. 1, 2

[12] Hyeongwoo Kim, Pablo Garrido, Ayush Tewari, Weipeng Xu, Justus Thies, Matthias Niessner, Patrick Pérez, Christian Richardt, Michael Zollhöfer, and Christian Theobalt. Deep video portraits. *ACM Transactions on Graphics (TOG)*, 37(4):1–14, 2018. 1, 2

[13] Dominik Lorenz, Leonard Bereska, Timo Milbich, and Bjorn Ommer. Unsupervised part-based disentangling of object shape and appearance. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10955–10964, 2019. 1, 2

[14] A. Nagrani, J. S. Chung, and A. Zisserman. Voxceleb: a large-scale speaker identification dataset. In *INTERSPEECH*, 2017. 5

[15] Yurui Ren, Xiaoming Yu, Junming Chen, Thomas H Li, and Ge Li. Deep image spatial transformation for person image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7690–7699, 2020. 1, 2

[16] Aliaksandr Siarohin, Stephane Lathuiliere, Sergey Tulyakov, Elisa Ricci, and Nicu Sebe. Animating arbitrary objects via deep motion transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 1, 2, 5, 6

[17] Aliaksandr Siarohin, Stéphane Lathuilière, Sergey Tulyakov, Elisa Ricci, and Nicu Sebe. First order motion model for image animation. In *Advances in Neural Information Processing Systems*, pages 7137–7147, 2019. 1, 2, 4, 5

[18] Yi Sun, Xiaogang Wang, and Xiaoou Tang. Deep Learning Face Representation by Joint Identification-Verification. *arXiv e-prints*, page arXiv:1406.4773, June 2014. 7

[19] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1701–1708, 2014. 7

[20] Ting-Chun Wang, Ming-Yu Liu, Andrew Tao, Guilin Liu, Jan Kautz, and Bryan Catanzaro. Few-shot video-to-video synthesis. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019. 1

[21] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Guilin Liu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. Video-to-Video Synthesis. *arXiv e-prints*, page arXiv:1808.06601, Aug. 2018. 4

[22] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 1, 2

[23] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8798–8807, 2018. 5

[24] Zhou Wang, Alan C Bovik, Hamid R Sheikh, Eero P Simoncelli, et al. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 5

[25] Olivia Wiles, A. Sophia Koepke, and Andrew Zisserman. X2Face: A network for controlling face generation by using images, audio, and pose codes. *arXiv e-prints*, page arXiv:1807.10550, July 2018. 1, 2, 5

[26] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. https://github.com/facebookresearch/detectron2, 2019. 5

[27] Egor Zakharov, Aleksei Ivakhnenko, Aliaksandra Shysheya, and Victor Lempitsky. Fast Bi-layer Neural Synthesis of One-Shot Realistic Head Avatars. *arXiv e-prints*, page arXiv:2008.10174, Aug. 2020. 2

[28] Egor Zakharov, Aliaksandra Shysheya, Egor Burkov, and Victor Lempitsky. Few-shot adversarial learning of realistic neural talking head models. In *Proceedings of the IEEE*

International Conference on Computer Vision, pages 9459–9468, 2019. 2, 5