Facial Image-to-Video Translation by a Hidden Affine Transformation

Guangyao Shen*† thusgy2012@gmail.com Tsinghua University
Wenbing Huang hwenbing@126.com Tencent AI Lab
Chuang Gan‡ ganchuang1990@gmail.com MIT-IBM Watson AI Lab
Mingkui Tan mingkuitan@scut.edu.cn Pengcheng Laboratory
Junzhou Huang joehhuang@tencent.com Tencent AI Lab
Wenwu Zhu‡ wwzhu@tsinghua.edu.cn Tsinghua University
Boqing Gong§ boqinggo@outlook.com Tencent AI Lab

ABSTRACT

There has been a prominent emergence of work on video prediction, aiming to extrapolate the future video frames from the past. Existing temporal-based methods are limited to certain numbers of frames. In this paper, we study video prediction from a single still image in the facial expression domain, a.k.a, **facial image-to-video translation**. Our main approach, dubbed **AffineGAN**, associates each facial image with an expression intensity and leverages an affine transformation in the latent space. AffineGAN allows users to control the number of frames to predict as well as the expression intensity for each of them. Unlike previous intensity-based methods, We derive an inverse formulation to the affine transformation, enabling **automatic** inference of the facial expression intensities from videos — manual annotation is not only tedious but also ambiguous as people express in various ways and have different opinions about the intensity of a facial image. Both quantitative and qualitative results verify the superiority of AffineGAN over the state of the arts. Notably, in a Turing test with web faces, more than 50% of the facial expression videos generated by AffineGAN are considered real by the Amazon Mechanical Turk workers. This work could improve users' communication experience by enabling them to conveniently and creatively produce expression GIFs, which are popular art forms in online messaging and social networks.

CCS CONCEPTS

• **Computing methodologies → Computer vision**.

KEYWORDS

face manipulation; image-to-video translation; GAN

*Work performed when Guangyao Shen was an intern at Tencent AI Lab under the supervision of Wenbing Huang.
†Beijing National Research Center for Information Science and Technology (BNRist).
‡Corresponding authors.
§Now at Google.

MM '19, October 21–25, 2019, Nice, France

ACM ISBN 978-1-4503-6889-6/19/10…$15.00
https://doi.org/10.1145/3343031.3350981

ACM Reference Format:
Guangyao Shen, Wenbing Huang, Chuang Gan, Mingkui Tan, Junzhou Huang, Wenwu Zhu, and Boqing Gong. 2019. Facial Image-to-Video Translation by a Hidden Affine Transformation. In *Proceedings of the 27th ACM International Conference on Multimedia (MM '19), October 21–25, 2019, Nice, France.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3343031.3350981

1 INTRODUCTION

GIF is a popular art form in online messaging and social networks. As a picture is worth a thousand words, GIFs, as short video sequence containing multiple pictures, convey vivid contents like humor and emotion. It improves users' communication experience significantly if the users can conveniently and creatively make GIFs, that is, generate video from simple inputs like few still images and several control conditions. There has been a prominent emergence of work on video prediction [3, 13, 14, 19, 24, 25, 30], which aims to extrapolate the future frames from the past. Despite the progress, most previous works [3, 13, 19, 22, 24–26, 29, 30] are **temporal-based**, which require predicted videos to be pre-defined numbers of frames, hence users are unable to control the temporal lengths or varying speeds of the videos once the models have been trained. Thus, the temporal-based methods are not so flexible for users to generate interesting videos.

In this paper, we are concerned with generating video sequences of facial expressions from a single still image. Unlike temporal-based work, we do not limit our output to any specific time span. Instead, we aim to predict a full **intensity-based** procedure of an expression change from the neutral state to the peak (e.g., laugh loudly). Users can control the frame rate of this procedure and generate as many frames as possible to unfold the expression. To achieve this, we assign each video frame a non-negative scalar, indicating the relative intensity of the current expression in the full neutral-to-peak procedure. Given an input image, our model hallucinates a sequence of video frames with an increasing series of non-negative expression intensities. This series can be customized and, in the extreme case, can contain a single scalar which instructs the model to generate only one frame corresponding to that intensity.

One of the major challenges is how to train such a model that ***automatically infers the expression intensities*** from the training videos. Most previous intensity-based approaches for expression synthesis manually label the intensities of frames by their temporal positions in the training video [7, 15]. In this way, their training

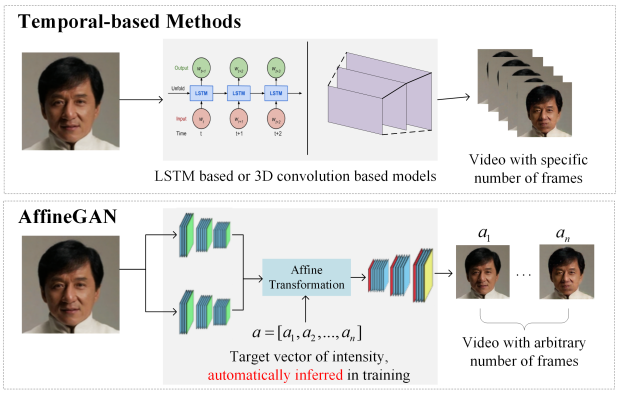

Figure 1: An illustration of predicting "drumming cheeks" expression. Compared to temporal-based methods, AffineGAN can predict videos of arbitrary temporal lengths. Superior to previous intensity-based methods, it can automatically infer the intensities without dense annotations.

processes demand complete videos as targets, thus are unfeasible to deal with incomplete, periodic, or unordered frames that are usually encountered in practice. Also, we contend that hand-craft annotations are not only costly but also vague. While a fairly good consensus could be reached at what is a neutral state, people tend to have different opinions about the peak expression. In other words, user annotation would inevitably cause ambiguity and confuse the learner. Moreover, if we have multiple videos of a person expressing the same emotion, her/his peak state may not appear in all the videos. The speeds of the expressions could also vary from one video to another. Consequently, if we resort to manually annotating the expression intensities, it would be costly and challenging to align the videos to each other precisely.

In order to address the above issues, we make a mild assumption enabling our model to **derive the expression intensity automatically** for any training frame: there exists a latent space in which the codes of frames take the affine form, i.e., $f_t = f_0 + a_t f_\Delta$, where f_t and a_t are respectively the latent code and expression intensity of the t-th frame, f_0 is the latent code of a neutral frame, and f_Δ encodes the direction to move from the neutral state to the current expression. We jointly learn all of them with the mere annotation of a neutral face frame per training video. The key is to relate the unknown expression intensity a_t with the codes of the training frames, by which deriving a_t is possible based on the aforementioned affine transformation. Fig. 2 depicts our approach. Both quantitative and qualitative results verify the superiority of AffineGAN over the state of the arts. Especially, in a Turing test with web faces, more than 50% of the generated expression videos are considered real by the Amazon Mechanical Turk workers. Finally, experiments also demonstrate the effectiveness of our approach in learning from incomplete, periodic, or unordered training frames.

We summarize the contributions of our paper as follows.

- We develop the AffineGAN for predicting facial expression videos of arbitrary temporal lengths from a single still image. More importantly, it can automatically infer the expression intensities from both training and test frames.
- AffineGAN can handle the training videos that are practically incomplete, unordered or periodic, requiring the mere annotation of only a neutral face frame per video.
- AffineGAN generates more realistic facial expressions than several competitive baselines and fools more than 50% workers in a Turing test. It can support people to conveniently and creatively make expression GIFs, which are popular in online messaging and social networks nowadays. [1]

2 RELATED WORK

Image-to-Image Translation. Image-to-image translation aims to translate images from one domain to another domain. Due to the power of GANs [10] in image generation, most recent approaches apply conditional GANs for image-to-image translation, where they condition on an input image and generating a corresponding output image [11]. Furthermore, the cycle consistency is considered in the translation between unpaired domains [4, 31]. Benefit from cGANs and cycle consistency, the image-to-image translation has been successfully applied to image synthesis [27], style-aggregated face generation [5], and video-to-video translations [28]. Our task can be seen as consequent image-to-image translations, but it is more challenging to model the dynamics and consistency.

Temporal-based Video Prediction. Inherently, frames are temporally related in videos. Therefore, most previous approaches for video prediction capture the temporal correlations based on 3D convolutional neural networks [13, 19, 26, 30] or recurrent neural networks [3, 8, 9, 22, 24, 25, 29, 30]. For example, Li *et al.* [13] propose a two-stage framework: the first phase predicts multiple time step optical flows and the second phase synthesis future frames from the first frame and the predicted flows. In [25], the authors adopt LSTM to observe several consecutive poses and predict future poses. However, for image-to-video translation, as only a single image is observed, temporal-based methods are hard to get enough sequence information. Also, limited by 3D CNN and LSTM, temporal-based methods could only generate certain numbers of frames. In contrast, our approach models the intensity changes instead of the temporal changes, so we can control the number of predicted frames and the intensity for each of them.

Controllable Facial Expression Synthesis. Recently, researchers have shown enthusiasm for controllable manipulation of facial expressions. Simple solutions like DFI [23] traverses the latent feature space and making linear interpolation between the average attributes of the source and target set in the latent space. However, it ignores the intensity of expressions and requires both sets to be similar enough to the test image. To tackle this challenge, on the one hand, researchers turn to fine-grained manual representations. G2-GAN [21] utilizes fiducial points (landmarks) as the controllable condition to guide facial expression synthesis. GANimation [17], a GAN conditioning scheme, aims to control the change of skin and muscles based on Action Units (AUs). However, the landmarks or AUs are hard to represent all the potential expressions and need to be annotated by unsatisfactory external tools. On the other hand, some researchers model the intensities of facial expression images explicitly. CAFP-GAN [15] uses controllable labels (expression and intensity) for expression synthesis. In [7], the authors proposed

[1] Code released in https://github.com/sunlightsgy/AffineGAN

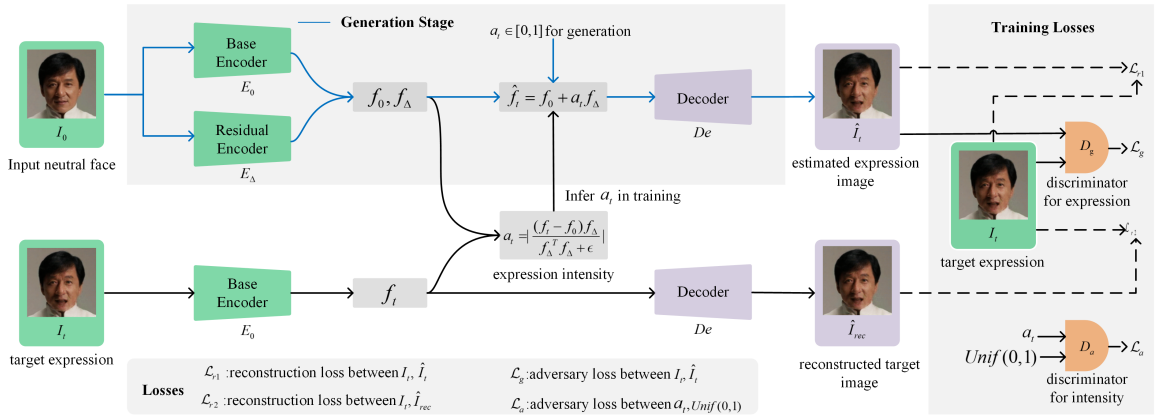

Figure 2: Framework of the proposed AffineGAN.

a user-controllable approach to generate videos clips of variable lengths from a single face image. However, these work manually label the expression intensities by their temporal positions in a training video. It incurs errors when the expression is incomplete or periodic and is a weak proxy for intensity as it assumes the speed of expression change is constant over time.

Comparing with the works above, our approach can infer the intensity automatically for the training frames and requires no large datasets, external tools, or extra annotations. Therefore, our approach is more flexible and robust, which can be trained with even incomplete or unordered frames in small datasets.

3 AFFINEGAN

Our goal is to generate a sequence of video frames $\boldsymbol{V} := \{\boldsymbol{I}_t\}_{t=0}^{T}$, given as input a neutral face image $\boldsymbol{I}_0 \in \mathbb{R}^{H \times W \times 3}$, where H and W are respectively the height and width of the image. This sequence unfolds an expression change beginning from the neutral face. We model each frame $\boldsymbol{I}_t$ at time t as a function of the input face $\boldsymbol{I}_0$ and an expression intensity $a_t \geq 0$, namely, $\boldsymbol{I}_t = g(\boldsymbol{I}_0, a_t)$. As this paper is concerned with the videos of facial expressions, we may cast some immediate constraints over the function $g(\boldsymbol{I}_0, a_t)$ to be learned. First, it should enable the *self-reconstruction* of the input image when $a_0 = 0$, *i.e.*, $\boldsymbol{I}_0 = g(\boldsymbol{I}_0, 0)$. Second, we should design the function to make this procedure *"monotonic"* with respect to the expression intensity a_t. The larger a_t is, the further the generated expression $\boldsymbol{I}_t$ is from the neutral image $\boldsymbol{I}_0$ and the closer it is to the peak state of the expression (e.g., laugh loudly). Finally, the output frame $g(\boldsymbol{I}_0, a_t)$ achieves at the peak expression state when $a_t \approx 1$. The last constraint ensures that the range of the expression intensity is close to $[0, 1]$, so that users can refer to this range when they specify the expressions for our model to predict.

The above problem formalism is advantageous over the existing video prediction works as it allows users to control the frame rate to depict the procedure of an expression change. The downside, however, is that the function $g(\boldsymbol{I}_0, a_t)$ depends on the expression intensity a_t, which is non-trivial to determine for the training frames. Fan *et al.*'s work [7] is the closest to ours. The authors suggest to bound a_t by 1 and then set it to a frame's normalized temporal position in a training video. However, this labeling method incurs errors when the training video is incomplete (an expression does not reach the peak state) or periodic (a person smiles again and again). Moreover, the temporal position could be a weak proxy to the expression intensity, as it basically assumes the speed of the expression change is constant over time. Finally, as we discussed earlier, people are expressive in distinct ways, making it hard to manually label the peak state across different subjects.

In order to tackle the challenges, we propose to automatically infer the expression intensity of a training video frame by imposing a mild assumption about the facial expressions changed linearly in a latent space. Thus, the mere annotation required for our approach is to select a frame per training video of a neutral face, which is often the first frame of the video.

3.1 Framework Design

We first describe a generator which approximates the frame prediction function $g(\boldsymbol{I}_0, a_t)$, followed by the method to derive the intensity a_t. We denote by $\widehat{\boldsymbol{I}}_t$ the estimated output of $g(\boldsymbol{I}_0, a_t)$ to distinguish it from the ground-truth frame $\boldsymbol{I}_t$ in the training set.

Generator. We make use of two encoders E_0 and E_Δ — one is called the basic encoder and the other called the residual encoder — to obtain the latent codes $\boldsymbol{f}_0$ and $\boldsymbol{f}_\Delta$ in the same latent space,

$$\boldsymbol{f}_0 = E_0(\boldsymbol{I}_0), \quad \boldsymbol{f}_\Delta = E_\Delta(\boldsymbol{I}_0), \tag{1}$$

where, by the residual vector $\boldsymbol{f}_\Delta$, we attempt to capture the direction of the expression change originated from the neutral state $\boldsymbol{f}_0$. The latent code of the target image is constructed through an affine transformation of the latent code $\boldsymbol{f}_0$ of the neutral face, the expression change direction $\boldsymbol{f}_\Delta$, and the expression intensity a_t,

$$\widehat{\boldsymbol{f}}_t = \boldsymbol{f}_0 + a_t \boldsymbol{f}_\Delta. \tag{2}$$

Despite its simplicity, this affine formulation can fulfill the three constraints over the frame prediction function $g(\boldsymbol{I}_0, a_t)$ laid out in Section 3. Indeed, the self-reconstruction property is naturally satisfied, $\widehat{\boldsymbol{f}}_0 = \boldsymbol{f}_0$; the monotone is guaranteed since $a_t \geq 0$; and we will show how to regularize a_t by a uniform distribution over $[0, 1]$ such that the peak expression is reached at about $a_t \approx 1$.

Finally, we generate the target image by feeding the latent code $\widehat{\boldsymbol{f}}_t$ to a decoder $De(\cdot)$,

$$\begin{aligned} g(\boldsymbol{I}_0, a_t) \approx \widehat{\boldsymbol{I}}_t &= De(\widehat{\boldsymbol{f}}_t) \\ &= De(E_0(\boldsymbol{I}_0) + a_t E_\Delta(\boldsymbol{I}_0)). \end{aligned} \tag{3}$$

It is interesting to read from the equation above that we may train the encoders and the decoder using pairs of images (one neutral face and the other an expression face). This gives us the flexibility of using a large batch size by sampling pairs from more than one training videos.

Thus far, we still have not addressed how to infer the expression intensity a_t. We present our solution next.

Inferring a_t. Our approach to inferring the expression intensity a_t hinges on the affine equation Eq. (2), from which we can calculate a_t inversely if we know the other quantities. Concretely, the calculation is as follows, and the propositions are proved in the supplementary materials.

Proposition 1. *If $\widehat{\boldsymbol{f}}_t = \boldsymbol{f}_0 + a_t \boldsymbol{f}_\Delta$, then we definitely have $a_t = (\widehat{\boldsymbol{f}}_t - \boldsymbol{f}_0)^{\mathrm{T}} \boldsymbol{f}_\Delta \,/\, \boldsymbol{f}_\Delta^{\mathrm{T}} \boldsymbol{f}_\Delta$.*

Of course, we do not know $\widehat{\boldsymbol{f}}_t$ unless through Eq. (3), leading to a chicken-and-egg issue. However, we do have access to the groundtruth video frame $\boldsymbol{I}_t$ in the training stage, so we can encode it through the basic encoder $\boldsymbol{f}_t = E_0(\boldsymbol{I}_t)$. Since $\widehat{\boldsymbol{f}}_t$ is an approximation of $\boldsymbol{f}_t$ in the latent space by design, we instead use the features $\boldsymbol{f}_t$ of the groundtruth video frame to compute a_t,

$$a_t \approx (\boldsymbol{f}_t - \boldsymbol{f}_0)^{\mathrm{T}} \boldsymbol{f}_\Delta \,/\, \boldsymbol{f}_\Delta^{\mathrm{T}} \boldsymbol{f}_\Delta. \tag{4}$$

We can also justify Eq. (4) by the proposition below.

Proposition 2. *If $\widehat{\boldsymbol{f}}_t = \boldsymbol{f}_0 + a_t \boldsymbol{f}_\Delta$, then the optimal solution to $\arg\min_{a_t} \|\boldsymbol{f}_t - \widehat{\boldsymbol{f}}_t\|_2$ is given by Eq. (4), where $\|\cdot\|_2$ is the ℓ_2 norm.*

In other words, we acknowledge by the proposition that replacing $\widehat{\boldsymbol{f}}_t$ with $\boldsymbol{f}_t$ incurs inconsistency when $\widehat{\boldsymbol{f}}_t \neq \boldsymbol{f}_t$. However, Proposition 2 tells that updating a_t via Eq. (4) is still the optimal choice in terms of the mean-squared error since it brings the two features close as much as possible. In a sense, it also ensures the generated image from $\widehat{\boldsymbol{f}}_t$ close to the real one. Another advantage of Eq. (4) is that it involves the term $\boldsymbol{f}_t - \boldsymbol{f}_0$ that naturally correlates the value of a_t with the difference between the target and input features: if $\boldsymbol{f}_t = \boldsymbol{f}_0$, then $a_t = 0$; if $\boldsymbol{f}_t$ is far away from $\boldsymbol{f}_0$, then a_t is large too. The offset term $\boldsymbol{f}_t - \boldsymbol{f}_0$ is projected to the residual direction $\boldsymbol{f}_\Delta$, so what survives is only the information relevant to the facial expression change.

We apply some straightforward enhancements to Eq. (4):

$$a_t = p(\boldsymbol{I}_0, \boldsymbol{I}_t) := \left| \frac{(\boldsymbol{f}_t - \boldsymbol{f}_0)^{\mathrm{T}} \boldsymbol{f}_\Delta}{\boldsymbol{f}_\Delta^{\mathrm{T}} \boldsymbol{f}_\Delta + \epsilon} \right|, \tag{5}$$

where a small value $\epsilon > 0$ is added to the denominator to prevent it from the ill-defined situation. Additionally, the absolute operation ensures that a_t is non-negative.

The overall encoder-decoder. Instantiating the intensity in Eq. (3) with Eq. (5) gives the overall structure,

$$\widehat{\boldsymbol{I}}_t = De(E_0(\boldsymbol{I}_0) + p(\boldsymbol{I}_0, \boldsymbol{I}_t) E_\Delta(\boldsymbol{I}_0)), \tag{6}$$

which is also depicted in Fig. 2. For generation, we choose a series of linearly increasing expression intensities $\{a_t\}$ from $[0, 1]$ and produce the corresponding images $\{\widehat{\boldsymbol{I}}_t\}$. We train models for expressions respectively, but it is easy to make a unified model by spatially replicating and concating domain labels, as well as adding a domain classification loss as in StarGAN [4].

3.2 Loss Functions

Given a training video clip of length T, we assume that the first frame $\boldsymbol{I}_0$ is a neutral expression. We do **not** require the T-th frame to be at the peak of the expression. Since our model automatically determines the intensity for each frame, it can justify which frame is at the crested point. As below, we present the training losses.

Adversarial losses on images. Recall that $\widehat{\boldsymbol{I}}_t$ is a frame predicted by applying Eq. (6). The first loss we use to train the proposed AffineGAN, which consists of the encoders E_0 and E_Δ and the decoder De, is an adversarial loss,

$$\mathcal{L}_g \quad := \quad -\log(1 - D_g(\widehat{\boldsymbol{I}}_t)) - \log D_g(\boldsymbol{I}_t)), \tag{7}$$

where D_g is a global discriminator which takes as input an image. Following [7], we can also employ an adversarial loss on some informative local regions of the frames optionally,

$$\mathcal{L}_l := -\log(1 - D_l\left(\widehat{\boldsymbol{I}}_t \circ \boldsymbol{M}_t\right)) - \log D_l\left(\boldsymbol{I}_t \circ \boldsymbol{M}_t\right), \tag{8}$$

where D_l is the local discriminator, $\boldsymbol{M}_t$ is the mask to crop out the local patch of interest, and $\circ$ denotes element-wise multiplication.

Reconstruction losses. Following [11], we further augment the adversarial loss with the reconstruction error between the estimated expression $\widehat{\boldsymbol{I}}_t$ and ground-truth $\boldsymbol{I}_t$, *i.e.*,

$$\mathcal{L}_{r1} = \|\boldsymbol{I}_t - \widehat{\boldsymbol{I}}_t\|_1. \tag{9}$$

In addition to this, all target images $\boldsymbol{I}_t$ should also reconstruct themselves after passing through the encoder E_0 and the decoder De. Hence, we define the second reconstruction loss by

$$\mathcal{L}_{r2} = \|\boldsymbol{I}_t - De(E_0(\boldsymbol{I}_t))\|_1. \tag{10}$$

Adversarial loss on a_t. As Eq. (5) shows, the expression intensity a_t is not naturally restrained within any particular range. We apply another adversarial loss to regularize its value so that it does not deviate away from the range $[0, 1]$. To do so, we define

$$\mathcal{L}_a \quad := \quad -\log(1 - D_a(a_t)) - \log D_a(x)), \tag{11}$$

where D_a is the discriminator and x is sampled from the uniform distribution over $[0, 1]$, *i.e.*, $x \sim Unif(0, 1)$.

Combining the above losses together, we alternately optimize the AffineGAN model and the discriminators by

$$\min_{D_g, D_l, D_a} \max_{G} \sum\nolimits_{t=1}^{T} \mathcal{L}_g + \mathcal{L}_l + \mathcal{L}_{r1} + \mathcal{L}_{r2} + \mathcal{L}_a. \tag{12}$$

where G refers to the generator. We empirically set the weights using validations. For a better understanding of our model, we summarize the entire procedure (Fig. 2) as follows:

- Training: E_0 encodes a neutral face $\boldsymbol{I}_0$ and a target face $\boldsymbol{I}_t$ to latent codes $\boldsymbol{f}_0$ and $\boldsymbol{f}_t$, and E_Δ encodes the direction $\boldsymbol{f}_\Delta$. Thus, the intensity a_t can be inferred through Eq. (5), and we can reconstruct target face $\widehat{\boldsymbol{I}}_t$ by Eq. (6). Then, the generators and discriminators can be optimized following Eq. (12).
- Generation: given a neutral face $\boldsymbol{I}_0$, $\boldsymbol{f}_0$ and $\boldsymbol{f}_\Delta$ can be encoded by E_0 and E_Δ. Thus, when intensity sequence $\{a_t\}$ comes, the target expressions $\{\widehat{\boldsymbol{I}}_t\}$ are generated by Eq. (3).

4 EXPERIMENTS

Datasets. As we require each training video contains one single expression of different intensities behaved by one single person, we construct CK-Mixed and Cheeks&Eyes datasets as follows:

CK-Mixed. The Cohn-Kanade (CK+) [16] dataset is prevalent in the facial expression analysis, where each expression is unfolded from the initial neutral state monotonically. It contains 593 videos of 8 emotion categories, in which most of the videos are in gray-scale except for the "contempt" class. To better analyze the emotions in RGB-scale, the CK++ [7] dataset adds 558 RGB-scale videos to the "happy", "angry", and "surprise" categories acted by 65 volunteers. We train and validate our approach on the *CK-Mixed* dataset that collects RGB-scale videos of the categories "happy", "angry", and "surprise" from CK++ and "contempt" from CK+.

Cheeks&Eyes. As a complement to the *CK-Mixed* dataset, we build another facial expression dataset focusing more on the eyes and cheeks. We asked 50 volunteers to each act three expressions before mobile phone cameras: closing eyes, raising eyes, and drumming cheeks. Then, we remove the extremely blurred frames and crop the frames to make the faces roughly centered. Unlike CK-Mixed, we retain some periodic frames in the videos.

For testing, we collect wild neutral faces from the Internet under various conditions of facial proportions, photo styles, *etc.*

Implementation Details. Our encoders and decoder are designed as 8-layer neural networks with skip connections [20], allowing low-level information to shortcut across the network. Both the local and global discriminators are designed as three-layer convolutional neural networks, and the discriminator regularizing the intensity a_t is a three-layer perceptron. For the experiments on CK-Mixed, we leave out 8 video clips (4 in gray-scale and 4 in RGB-scale) per class for validation. For each category in Cheeks&Eyes, we use 10 video clips (5 males and 5 females) for validation. All other video clips are used for training. For each video, we use the first frame as the neural face and pair it with five randomly sampled frames in training. We adopt instance normalization and set the video batch size to 1 following [11]. Adam [12] is the chosen optimizer with a learning rate 0.0002, beta1 0.5, and beta2 0.999. The weight coefficients for the loss terms in Eq.12 are set to 1, 1, 100, 10, 100, respectively. For all the expressions of CK-Mixed and drumming cheeks, we crop out the mouth region as the local patch of interest in Eq. 7. Regarding raising eyes and closing eyes, we disregard this loss as the subjects' mouths almost keep stationary for them.

4.1 Comparison Results

4.1.1 State-of-the-art methods. We compare AffineGAN with several state-of-the-arts: ConvLSTM [29], VideoGAN [26], the flow-grounded spatial-temporal (FGST) method [13], and GANimation [17]. For fair comparison, in ConvLSTM, we enhance the CovNet with U-net [20] to improve the performance. For VideoGAN, we adopt the conditional version of their official implementation by generating videos conditioning on the first frame. As the LSTM and 3D convolution implementations require a fixed-length sequence as inputs, we uniformly sample/up-sample 9 frames from each original video clips for ConvLSTM, VideoGAN, and FGST. In order to apply GANimation to our setting of video prediction from a single image, we pre-process the videos in the following way. We first align the faces and then extract their AUs via OpenFace [2]. Furthermore, in the prediction stage, we select the most expressive AU from the training set as the label map of each target expression.

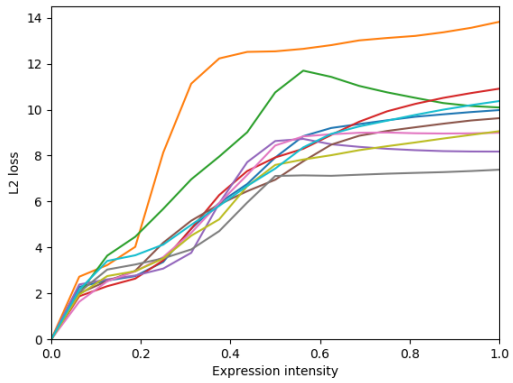

Figure 3: L2 loss between the generated frames and the neutral face on the test set of "closing eyes'. Different lines represent videos generated from different test images.

4.1.2 Qualitative Results. Fig. 4 compares our method with the state-of-the-arts on the validation set of the "happy" category. It is clear that the proposed AffineGAN produces the most realistic, smooth, and consistent frame sequence. Some of the frames generated by ConvLSTM are blurry. VideoGAN performs the worst and produces faces of a different person, probably because it overfits to the training set and fails to generalize to previously unseen subjects. FGST's results are better than VideoGAN's due to the 3D convolution over multiple optical flows, but it generates distorted images because the flows extracted by SPyNet [18] are probably not fine-grained enough to capture facial changes (maybe other methods like [6] can achieve better results). GANimation produces images most comparable to ours, but the facial change is within a much smaller range (*e.g.*, the teeth cannot be observed).

To further evaluate the generalization capacity of our approach, we try to generate facial expressions for images in test set crawled from the Internet. As shown in Fig. 5, our approach still generates satisfactory expressions. However, ConvLSTM, VideoGAN, and FGST perform much worse due to the domain gap between the training set and these Internet images. GANimation worked well on CK-Mixed but herein fails to produce semantically meaningful videos of "closing eyes". This is because GANimation relies on AUs extracted by Openface which, however, does not compute the AUs for eyes movements. In sharp contrast to GANimation, our AffineGAN is self-contained — without resorting to any additional tools or annotations — and can automatically derive the expression intensity. Fig. 6 shows that AffineGAN delivers consistent performances on the Internet images in the test set for all the expressions, from an extreme expression like "happy" to a modest one like "contempt". When a is close to 0 or 1, the generated frames may look a little similar. However, Fig. 3 shows that for 10 images in "closing eyes", the l_2 losses between the generated frames and the corresponding neutral face increase as the intensity move from 0 to 1, indicating that the frames are inherently different and monotonic.

Figure 4: Qualitative comparison with baselines for "happy" on the validation set.

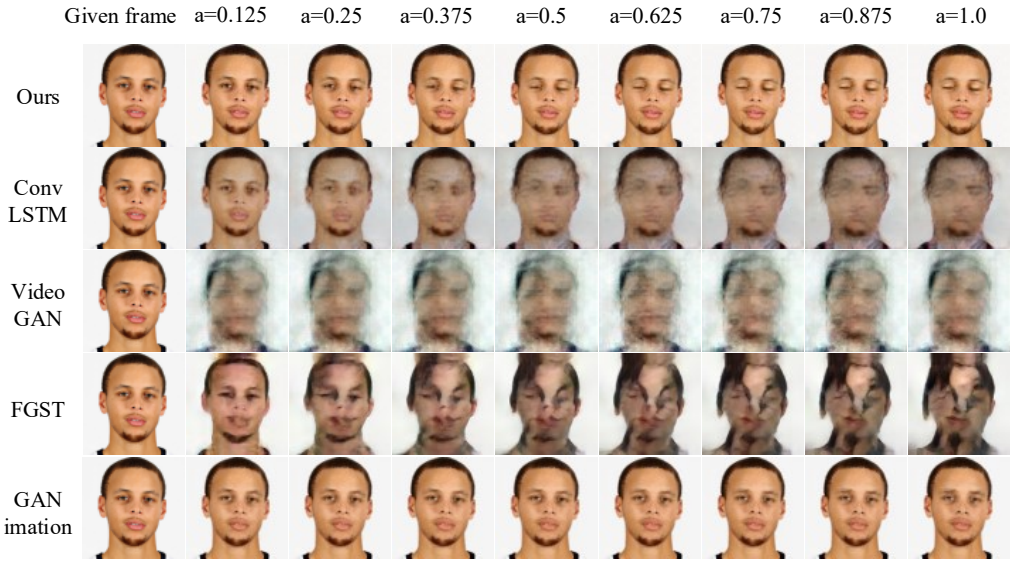

Figure 5: Qualitative comparison with baselines for "closing eyes" on the test set.

4.1.3 Quantitative Results. We also report quantitative results by the Average Content Distance (ACD) [30] and user studies. ACD-I measures the quality of facial identities by calculating the average distance between predicted frames and the original input, while ACD-C measures the content consistency by computing the average distance of all possible pairs of frames in a video. To better evaluate the expression changes, we further propose ACD-G metric by computing the average frame-to-frame distance between the generated frames and the corresponding ground-truth ones. We use OpenFace [1] to produce 128-dimensional feature vectors for the video frames. All the ACD scores are calculated using the ℓ_2 distance upon the feature vectors. When OpenFace cannot recognize any face from a generated frame — implying a bad prediction, we sample a feature vector from the normal distribution $N(0, 1)$.

Table 2 shows the overall comparison results on the validation set. We also compute all ACD scores for the ground-truth videos for reference. Note that the ACD scores are the lower, the better. We can see that AffineGAN gives rise to much lower ACD scores

Figure 6: The generated videos for all the seven categories on a previously unseen Internet image.

Table 1: Comparisons of AMT results between AffineGAN and baselines on CK-Mixed and Cheeks&Eyes.

Q1: "Which video represents # expression better?"	Happy	Anger	Surprise	Contempt	Raise eyes	Drum cheeks	Close eyes	Mean
Preference ours over ConvLSTM	75.5%	73.7%	73.8%	65.6%	78.2%	74.8%	82.7%	74.9%
Preference ours over VideoGAN	77.8%	76.1%	81.6%	72.8%	75.4%	79.8%	80.2%	77.7%
Preference ours over FGST	72.4%	74.1%	77.8%	68.3%	88.8%	90.5%	85.8%	79.7%
Preference ours over GANimation	72.1%	66.2%	78.9%	57.9%	71.0%	74.2%	68.8%	69.9%
Q2: "Is this video real or fake?"								
Mark real video as real video	81.5%	78.3%	80.8%	76.5%	83.3%	80.9%	84.0%	80.8%
Mark fake video as real video	50.6%	51.7%	43.5%	44.4%	69.6%	58.2%	61.5%	54.2%

Table 2: Generation quality comparison.

Methods	ACD-I	ACD-C	ACD-G
ConvLSTM	1.31	1.38	1.33
VideoGAN	1.64	1.14	1.64
FGST	0.99	1.10	0.98
GANimation	0.35	**0.25**	0.47
Ours	**0.31**	0.29	**0.39**
Reference	0.29	0.28	0

than ConvLSTM, VideoGAN, and FGST and slightly higher scores than the ground-truth reference. GANimation produces very low ACD-C because it predicts almost constant frames for each video.

In addition to the ACD scores, we also run two user studies on Amazon Mechanical Turk (AMT). In the first, we sample 50 predicted videos per expression for each method. We each time display a video per method (five videos in total) of the same expression in random order on the screen. Then, we ask 50 workers to choose which one of the five videos best represents the expressions. As shown in Table 1, the workers prefer our solution over all baselines, especially on Cheeks&Eyes. For the second study, we conduct a Turing test for each expression by randomly choosing 10 real videos from the training set and 10 generated videos of the testing faces. We shuffle these videos and assign each worker with 50 videos. Workers are asked to decide whether the displayed video is real or not. In Table 1, about 85% of the real videos are labeled as "real", verifying the reliability of the workers. As for the videos predicted by AffineGAN, 54.2% of them are marked as real on average and, in particular, 70% of the "raising eyes" videos fool the workers successfully. Both studies demonstrate that AffineGAN can generate more realistic and expressive videos for all these facial expressions.

4.2 Ablation Studies

On inferring the expression intensity by $p(I_0, I_t)$. We have derived an automatic approach to inferring the intensity a_t in Section 3.1. Instead of using the particular formulation $p(I_0, I_t)$, one

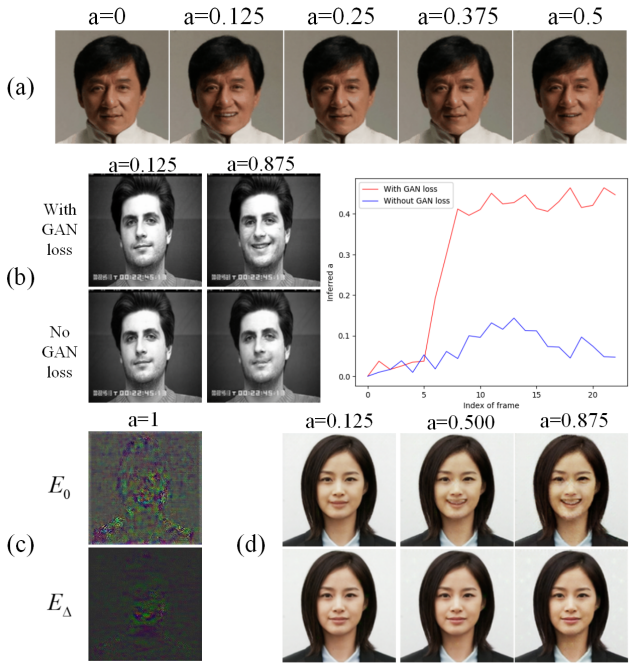

Figure 7: (a) Plain MLP is not as effective as the derived network design (Eq. 7) for inferring the expression intensity. (b) Comparison for results with/without adversary loss for a. Left: qualitative results; right: the inferred a of groundtruth frames. (c) Visualizations of gradients in the input layer for E_0 and E_Δ. (d) Generated expressions when only two frames are remained in the training set. Upper: Ours; lower: FAN.

may wonder an alternative "black-box" neural network which takes as input the concatenation of f_0, f_t, and f_Δ. We contend that this naive solution does not necessarily cover the formulation $p(I_0, I_t)$ and, even if it does, it requires a massive dataset to learn. To verify this, we instead train an ablated AffineGAN which uses a plain MLP to estimate the intensity. Fig. 7(a) shows the results. Clearly, the generated video by this ablated AffineGAN is non-monotonic, indicating that the MLP fails to capture the expression intensity.

On the effect of the adversary loss for a_t. Recall that we apply an adversary loss to constrain the intensity a_t to be close to $[0, 1]$ (cf. Eq. 11). Here, we justify its importance by removing it from our overall objective function. Fig. 7(b) shows that the facial changes are much small without this adversary loss. As our method can infer the intensity itself, we further display the predicted values of a_t by our method in Fig. 7(b). Obviously, without the adversary loss, the inferred a_t is not monotonic and sometimes moves beyond the range of [0,1]. In contrast, the intensity a_t estimated by the full AffineGAN approach is reasonably within $[0, 1]$.

What do the encoders learn? Our generator contains a basic encoder E_0 and a residual encoder E_Δ, as introduced in Sec. 3.1. To gain insights about them, we visualize the gradients of the input layers of them when the expression intensity is set to 1. Fig. 7(c) shows a "happy" example. It shows that while the basic encoder E_0 mostly captures the outline of the face (*e.g.*, identity information), the residual encoder E_Δ focuses more on the active expression part (*e.g.*, the mouth area). These observations are consistent with our design by which we expect E_0 to model the neutral face and E_Δ to track the direction of expression change.

Incomplete, unordered, or periodic frames. As discussed in Section 3, the mere annotation required for our approach is to select a neutral face per training video. Indeed, AffineGAN can accept incomplete, unordered, or periodic frames of facial expressions. This property makes our method very flexible and more advantageous than the closely related approach (FAN) [7], where both the neutral and peak expressions are manually labeled and a linear change in between is assumed. We compare AffineGAN with FAN under the settings with these frames. First, for the incomplete setting, we randomly select only two frames from each video. The intensity of the second frame is labeled to 1 according to FAN. As shown in Fig. 7(d), FAN fails to make reasonable intensity change, but our approach still leads to satisfactory performance. Second, for the unordered and periodic setting, we shuffle the frames except the first for each video and also repeat each video for random times to construct unordered and periodic videos. FAN fails to generate meaningful expression as its assumption of linear intensity change becomes fundamentally wrong. As a result, FAN also performs badly on the Cheeks&Eyes dataset, which contains unordered training frames. Nevertheless, our method can well handle these videos.

5 CONCLUSION

This paper proposes AffineGAN to enable controllable facial expression generation from a single neural face. The key advantage of AffineGAN is that the expression intensity used for the expression generation can be inferred automatically by an inverse formulation to the affine transformation. We evaluate AffineGAN in two expression datasets: CK-Mixed and Cheeks&Eyes. Both the qualitative and quantitative results verify the effectiveness of our method. To further justify the generation ability of AffineGAN, we collect neural faces from the Web and construct a test set that exhibits a clear domain gap to our training faces. The generated expressions, beyond expectation, can fool more than 50% AMT workers in our Turing test. Considerable ablation studies have also be performed to reveal the robustness of our method. This method can be more powerful; for example, if we select happiness as the source expression and neutral as the target, we can go from happy to neutral, thus enabling face synthesis like happiness to anger.

Nowadays, GIF is a popular art form in online messaging and social networks, which improve users' experience dramatically. AffineGAN can support people to produce expression GIFs 1) conveniently, as only one neural face is required in generation; 2) creatively, as models for new expressions can be trained on easy-to-gather datasets. For the future work, we will explore to apply our method on other types of images, *e.g.*, the action data.

ACKNOWLEDGMENTS

This work was support by National Program on Key Basic Research Project No. 2015CB352300, National Natural Science Foundation of China Major Project No.U1611461 and Shenzhen Nanshan District Ling-Hang Team Grant under No.LHTD20170005. Prof. Tan was supported by Program for Guangdong Introducing Innovative and Enterpreneurial Teams 2017ZT07X183. Special thanks to Lijie Fan and the volunteers.

REFERENCES

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

[2] Tadas Baltrusaitis, Amir Zadeh, Yao Chong Lim, and Louis-Philippe Morency. 2018. OpenFace 2.0: Facial Behavior Analysis Toolkit. In *IEEE International Conference on Automatic Face and Gesture Recognition*. IEEE, 59–66.

[3] Haoye Cai, Chunyan Bai, Yu-Wing Tai, and Chi-Keung Tang. 2018. Deep Video Generation, Prediction and Completion of Human Action Sequences. In *The European Conference on Computer Vision (ECCV)*. 374–390.

[4] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. 2018. StarGAN: Unified Generative Adversarial Networks for Multi-Domain Image-to-Image Translation. In *CVPR*.

[5] Xuanyi Dong, Yan Yan, Wanli Ouyang, and Yi Yang. 2018. Style Aggregated Network for Facial Landmark Detection. In *CVPR*. 379–388.

[6] Lijie Fan, Wenbing Huang, Chuang Gan, Stefano Ermon, Boqing Gong, and Junzhou Huang. 2018. End-to-End Learning of Motion Representation for Video Understanding. In *CVPR*.

[7] Lijie Fan, Wenbing Huang, Chuang Gan, Junzhou Huang, and Boqing Gong. 2019. Controllable Image-to-Video Translation: A Case Study on Facial Expression Generation. In *AAAI*.

[8] Chuang Gan, Chen Sun, Lixin Duan, and Boqing Gong. 2016. Webly-supervised video recognition by mutually voting for relevant web images and web video frames. In *ECCV*. 849–866.

[9] Chuang Gan, Ting Yao, Kuiyuan Yang, Yi Yang, and Tao Mei. 2016. You lead, we exceed: Labor-free video concept learning by jointly exploiting web videos and images. In *CVPR*. 923–932.

[10] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *NIPS*. 2672–2680.

[11] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. 2017. Image-to-Image Translation with Conditional Adversarial Networks. In *CVPR*. 5967–5976.

[12] Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *ICLR*.

[13] Yijun Li, Chen Fang, Jimei Yang, Zhaowen Wang, Xin Lu, and Ming-Hsuan Yang. 2018. Flow-Grounded Spatial-Temporal Video Prediction from Still Images. In *The European Conference on Computer Vision (ECCV)*. 609–625.

[14] Yue Liu, Xin Wang, Yitian Yuan, and Wenwu Zhu. 2019. Cross-Modal Dual Learning for Sentence-to-Video Generation. In *Proceedings of the 27th ACM international conference on Multimedia*. ACM.

[15] Zhihe Lu, Tanhao Hu, Lingxiao Song, Zhaoxiang Zhang, and Ran He. 2018. Conditional Expression Synthesis with Face Parsing Transformation. In *2018 ACM Multimedia Conference*. 1083–1091.

[16] Patrick Lucey, Jeffrey F Cohn, Takeo Kanade, Jason Saragih, Zara Ambadar, and Iain Matthews. 2010. The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. In *CVPRW*. 94–101.

[17] A. Pumarola, A. Agudo, A.M. Martinez, A. Sanfeliu, and F. Moreno-Noguer. 2018. GANimation: Anatomically-aware Facial Animation from a Single Image. In *ECCV*. 835–851.

[18] Anurag Ranjan and Michael J. Black. 2017. Optical Flow Estimation Using a Spatial Pyramid Network. In *CVPR*. 2720–2729.

[19] Fitsum A. Reda, Guilin Liu, Kevin J. Shih, Robert Kirby, Jon Barker, David Tarjan, Andrew Tao, and Bryan Catanzaro. 2018. SDC-Net: Video prediction using spatially-displaced convolution. In *ECCV*.

[20] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*. 234–241.

[21] Lingxiao Song, Zhihe Lu, Ran He, Zhenan Sun, and Tieniu Tan. 2018. Geometry Guided Adversarial Facial Expression Synthesis. In *2018 ACM Multimedia Conference*. 627–635.

[22] Sergey Tulyakov, Ming-Yu Liu, Xiaodong Yang, and Jan Kautz. 2018. Mocogan: Decomposing motion and content for video generation. In *CVPR*. 1526–1535.

[23] Paul Upchurch, Jacob Gardner, Geoff Pleiss, Robert Pless, Noah Snavely, Kavita Bala, and Kilian Q. Weinberger. 2017. Deep Feature Interpolation For Image Content Changes. In *CVPR*. 6090–6099.

[24] Ruben Villegas, Jimei Yang, Seunghoon Hong, Xunyu Lin, and Honglak Lee. 2017. Decomposing Motion and Content for Natural Video Sequence Prediction. *ICLR* (2017).

[25] Ruben Villegas, Jimei Yang, Yuliang Zou, Sungryull Sohn, Xunyu Lin, and Honglak Lee. 2017. Learning to Generate Long-term Future via Hierarchical Prediction. In *ICML*.

[26] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. 2016. Generating Videos with Scene Dynamics. In *NIPS*. 613–621.

[27] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. 2018. High-Resolution Image Synthesis and Semantic Manipulation with Conditional GANs. In *CVPR*.

[28] Xingxing Wei, Jun Zhu, Sitong Feng, and Hang Su. 2018. Video-to-Video Translation with Global Temporal Consistency. In *2018 ACM Multimedia Conference on Multimedia Conference*. ACM, 18–25.

[29] Shi Xingjian, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo. 2015. Convolutional LSTM network: A machine learning approach for precipitation nowcasting. In *NIPS*. 802–810.

[30] Long Zhao, Xi Peng, Yu Tian, Mubbasir Kapadia, and Dimitris Metaxas. 2018. Learning to forecast and refine residual motion for image-to-video generation. In *ECCV*. 403–419.

[31] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. 2017. Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networkss. In *ICCV*.