3D-Aware Talking-Head Video Motion Transfer

Haomiao Ni\textsuperscript{1} Jiachen Liu\textsuperscript{1} Yuan Xue\textsuperscript{2} Sharon X. Huang\textsuperscript{1}
\textsuperscript{1}The Pennsylvania State University, University Park, PA, USA
\textsuperscript{2}The Ohio State University, Columbus, OH, USA
\textsuperscript{1}\{hfn5052, jzl6493, suh972\}@psu.edu \textsuperscript{2}Yuan.Xue@osumc.edu

Abstract

Motion transfer of talking-head videos involves generating a new video with the appearance of a subject video and the motion pattern of a driving video. Current methodologies primarily depend on a limited number of subject images and 2D representations, thereby neglecting to fully utilize the multi-view appearance features inherent in the subject video. In this paper, we propose a novel 3D-aware talking-head video motion transfer network, Head3D, which fully exploits the subject appearance information by generating a visually-interpretable 3D canonical head from the 2D subject frames with a recurrent network. A key component of our approach is a self-supervised 3D head geometry learning module, designed to predict head poses and depth maps from 2D subject video frames. This module facilitates the estimation of a 3D head in canonical space, which can then be transformed to align with driving video frames. Additionally, we employ an attention-based fusion network to combine the background and other details from subject frames with the 3D subject head to produce the synthetic target video. Our extensive experiments on two public talking-head video datasets demonstrate that Head3D outperforms both 2D and 3D prior arts in the practical cross-identity setting, with evidence showing it can be readily adapted to the pose-controllable novel view synthesis task.

1. Introduction

The task of transferring motion between talking-head videos, while maintaining the identity of the target subject, is a compelling research area with broad applications in special effects, entertainment, and video editing. Despite the significant progress in guided image-to-image synthesis, such as person image generation [1, 25, 30] and facial expression generation [4, 21, 31], the challenge of capturing the temporal dynamics of motion in video-to-video transfer remains unsolved [5, 29]. Most current methods for talking-head video motion transfer use one subject image [35, 36, 38] or a simple combination of a few subject images [13, 28, 44, 45, 48] with 2D representations. These approaches may struggle to fully leverage the multi-view appearance information inherent in the subject video.

In this paper, we introduce Head3D, a novel 3D-aware framework for transferring motion between talking-head videos. This framework operates in a self-supervised, non-adversarial training manner, and is capable of recovering 3D structural information (i.e., head pose and depth) from each 2D video frame through self-supervised 3D head geometry learning, without the need for a 3D graphical model of the human head. By mapping each selected subject video frame to a 3D canonical space, Head3D further estimates a 3D subject canonical head using a recurrent network. To synthesize the final video frames, Head3D employs an attention-based fusion mechanism to combine appearance features from the 3D subject head with background and other details (e.g., facial expression, shoulder) from the subject. Unlike previous 3D-based methods that operate on the canonical feature space [6, 37, 46], Head3D offers visual interpretability by explicitly modeling the 3D canonical head. Compared with NeRF-based methods [7, 15, 26], Head3D shows better generalization ability without the need to retrain the model on unseen faces. Moreover, with the generated 3D subject head, Head3D can effectively handle large pose changes or extreme poses and achieve novel view syn-
thesis with user-provided pose inputs, as demonstrated in Fig. 1. Our contributions are summarized as follows:

- We introduce Head3D, a 3D-aware generative network for talking-head video motion transfer, which explicitly estimates a 3D canonical head without the need for any 3D shape priors.
- We propose a self-supervised 3D head geometry learning module with a recurrent network to generate a 3D visually-interpretable canonical head from the 2D subject video.
- Comprehensive experiments demonstrate that our proposed Head3D outperforms other 2D- and 3D-based methods in the practical cross-identity motion transfer setting. Our model can also be easily adapted to pose-controllable novel view synthesis.

2. Related Work

According to whether 3D information is utilized during generation, talking-head video motion transfer methods can be categorized into 2D- or 3D-based frameworks.

2D-based talking-head video motion transfer. Based on whether to use multiple frames from the subject video, 2D-based methods can be further classified into one-shot [35, 36, 38, 41, 47, 54] and few-shot methods [13, 28, 44, 45, 48, 51, 52]. One-shot 2D methods, also known as image animation, focus on generating videos based on one given subject image and one driving video. Siarohin et al. [36] proposed a general self-supervised first-order-motion framework (FOMM) to predict dense motion flow for animating arbitrary objects with learned keypoints and local affine transformations. In [38], the authors further improved their network by modeling object movement through unsupervised region detection. Tao et al. [41] improved FOMM by introducing a deformable anchor model (DAM) to ensure that the object structure is well captured and preserved. However, these one-shot 2D methods are limited to using a single subject image, which makes it hard for them to utilize the multi-view appearance features of the subject when the subject video is available.

Few-shot 2D methods instead utilize the subject video more effectively by synthesizing a video based on a number of subject video frames. Wang et al. [45] proposed a video-to-video synthesis approach (vid2vid) under the generative adversarial learning framework [12], which produces one new video frame based on several previously generated images and the corresponding landmarks of the driving frame. In [44], they further proposed a few-shot vid2vid framework to learn how to synthesize videos of unseen subjects by leveraging a few example images of the target at test time. Ha et al. [13] proposed a few-shot face reenactment framework, MarioNETTe, which employed image attention block, target feature alignment, and landmark transformer to address unseen identity and large-pose gaps. While few-shot methods have shown promising performance by utilizing appearance information from multiple frames, they operate only on 2D features and thus fail to fully exploit the multi-view information available in subject videos.

3D-based talking-head video motion transfer. 3D-based models have shown substantial progress in recent years. Some recent methods [3, 9, 10, 19, 24, 40] incorporate predefined shape models (e.g., 3DMM [2] or FLAME [23]) to model 3D face for face manipulation. Liu et al. [24] proposed 3D-FM GAN for 3D-controllable face manipulation by encoding both the input face image and a physically-based rendering of 3D edits into the latent space of StyleGAN [18]. However, these methods depend on predefined 3D graphical models that may have limitations in modeling the unique shape details of different subjects. Other recent methods [7, 8, 15] instead used Neural Radiance Fields (NeRFs) [26] as a 3D representation of the human head. Gafni et al. [7] proposed dynamic neural fields for modeling the appearance and dynamics of a human face tracked by 3DMM [2]. However, it can be hard for these NeRF-based models to generalize to unseen subject videos and they require fine-tuning or retraining when applied to new subjects. Some other methods [6, 14, 42, 46] are based on 3D geometrical transformation. Wang et al. [46] proposed a one-shot free-view neural talking-head video synthesis model which represents a video using a sparse 3D keypoint representation. Hong et al. [14] introduced a self-supervised geometry learning method to automatically recover depth from face videos and leverage them to estimate sparse facial keypoints for talking head generation. Though also using 3D geometrical transformation, our proposed Head3D is different from these methods by explicitly modeling and visualizing the 3D canonical head estimated from the 2D subject video, thus providing an easily interpretable representation of the subject’s head.

3. Methodology

Figure 2 shows the training framework of our Head3D. In general, Head3D is trained in an unsupervised manner, using self-reconstruction loss to restore one video frame with several randomly sampled frames from the same video. This training process neither requires any human annotation nor involves adversarial training. The training of Head3D consists of three stages: (1) 3D head geometry learning, (2) recurrent canonical head generation, and (3) attention-based fusion mechanism. To ease the training, we train the modules in these three stages separately. Given a set of randomly sampled $N$ reference frames $S_{\text{ref}} = \{s_1, s_2, \ldots, s_N\}$ and a driving frame $s_{\text{dr}}$ from the same training video $S$, in the first stage, we utilize a self-supervised 3D head geom-
etry learning framework to train a depth network \( F_D \) and a pose network \( F_P \) for predicting the head pose and depth of each 2D video frame. During the second stage, we use a recurrent canonical head generation network that leverages ConvLSTM-based feature aggregation to create a 3D canonical head \( \hat{x}^c \) incorporating warped reference frame features. Finally, in stage three, we employ an attention-based fusion mechanism to synthesize each final output frame \( s_{\text{dr}} \) by combining head appearance from the canonical head \( \hat{x}^c \), the background and other appearance details (e.g., neck and shoulder) from one randomly selected subject frame \( s_{\text{ref}} \), and motion and expression information from the driving frame \( s_{\text{dr}} \). Details of each component of our proposed framework are introduced as follows. More implementation details can be found in Sec 4.2.

### 3.1. 3D Head Geometry Learning

Given a talking-head video \( \mathcal{S} \), we first randomly sample a set of \( N \) reference frames \( \mathcal{S}_{\text{ref}} = \{ s_1, s_2, \ldots, s_N \} \) and one driving frame \( s_{\text{dr}} \) from \( \mathcal{S} \). To recover the 3D geometry of the subject’s head from a 2D talking-head video, we assume that videos are captured with a static perspective camera and that the subject’s head can be treated as a rigid object. Our motivation is, by estimating a 3D head in canonical space, \textit{i.e.}, a 3D canonical head, the head region of each target video frame can be reconstructed by transforming the points of the 3D canonical head using a rigid pose transformation \( \mathbf{P} = \{ \mathbf{R}, \mathbf{t} \} \in \text{SE}(3) \). To only reconstruct the head part in subject video frames, we employ a pretrained face parsing network [50] to extract the facial and hair regions. This results in a set of reference head images \( \mathcal{X}_{\text{ref}} = \{ x_1, x_2, \ldots, x_N \} \) and a driving head image \( x_{\text{dr}} \).

As shown in Fig 2, after head extraction, we apply a depth estimation network \( F_D \) and a head pose prediction network \( F_P \) to each frame in \( \mathcal{X}_{\text{ref}} \) and \( x_{\text{dr}} \) for estimating their depth maps \( D_{\text{ref}} = \{ d_1, d_2, \ldots, d_N \} \) and \( d_{\text{dr}} \), and their head poses \( \mathbf{P}_{\text{ref}} = \{ \mathbf{P}_1, \mathbf{P}_2, \ldots, \mathbf{P}_N \} \), \( \mathbf{P}_{\text{dr}} \). For each reference frame \( x_i \) in \( \mathcal{X}_{\text{ref}} \), where \( i = 1, \ldots, N \), we compute the corresponding canonical frame \( x_i^c \) based on the image formation model in [49]. Let pixel \( q = (u, v, 1) \) be the homogeneous coordinate of one pixel in the reference frame \( x_i \), and pixel \( \hat{q} = (u^c, v^c, 1) \) be the corresponding pixel in the canonical frame \( x_i^c \). We can transform each pixel \( q \) to \( \hat{q} \) to generate canonical frame \( x_i^c \) by:

\[
q^c \propto K (\mathbf{R}^T (d[u, v] \cdot K^{-1} q - t)) , \tag{1}
\]

where \( d[u, v] \) is the depth value of pixel \((u, v)\) in the depth map \( d \), \( \{ \mathbf{R}, \mathbf{t} \} \) is the head pose of frame \( x_i \), and \( K \) is the camera intrinsic matrix, which can be computed by:

\[
K = \begin{pmatrix}
f & 0 & c_u \\
0 & f & c_v \\
0 & 0 & 1
\end{pmatrix}, \quad \begin{cases}
c_u = \frac{W - 1}{2} \\
c_v = \frac{H - 1}{2} \\
f = \frac{W - 1}{2 \tan \frac{FOV}{2}}
\end{cases}, \tag{2}
\]

where \( H \) and \( W \) are the height and width of image, \( \theta_{\text{FOV}} \) is the field of view of the perspective camera. Following [49], we assume \( \theta_{\text{FOV}} \approx 10^\circ \) and the nominal distance of the subject from the camera is about 1m. To simplify the training, we take the average of all the warped canonical frames \( x_i^c \) to produce the final canonical frame \( \bar{x}^c \). We also apply \( F_D \) to obtain its depth map \( d^c \). Similar to Eq. (1), we can transform each pixel \( q^c \) in the canonical frame to the pixel \( q \) in the target frame by:

\[
q \propto K (d^c[u^c, v^c] \cdot \mathbf{R} K^{-1} q^c + \mathbf{t}) , \tag{3}
\]

where \( q = (u, v, 1) \) is the homogeneous coordinate of one pixel in the target frame. By applying \( F_P \) to driving head frame \( x_{\text{dr}} \) to estimate the head pose \( \mathbf{P}_{\text{dr}} = \{ \mathbf{R}_{\text{dr}}, \mathbf{t}_{\text{dr}} \} \), using Eq. (3), we can transform the canonical frame \( \bar{x}^c \) to frame \( x_{\text{dr}} \) with \( \mathbf{P}_{\text{dr}} \). Then we can train depth network \( F_D \) and pose network \( F_P \) with the following head reconstruction loss function:

\[
I_{\text{geo}} = ||\bar{x}_{\text{dr}} - x_{\text{dr}}||_1 + \lambda_{\text{sym}} \mathcal{L}_{\text{sym}}(\bar{x}^c) + \lambda_{\text{D}} \mathcal{L}_{\text{D}}(d^c) , \tag{4}
\]
where $\mathcal{L}_{\text{sym}}$ is designed to ensure the estimated 3D head under the canonical pose by imposing symmetry constraint. Here $\mathcal{L}_{\text{sym}} = ||\hat{x}^c - \bar{x}^c||_1$, where $\bar{x}^c$ is the horizontally-flipped version of $\hat{x}^c$. $\mathcal{L}_D$ is the depth smoothness loss used in [11]. $\lambda_{\text{sym}}$ and $\lambda_D$ are balancing coefficients.

### 3.2. Recurrent Canonical Head Generation

The canonical head image $\bar{x}^c$ is computed by averaging each transformed reference head frame in $\mathcal{X}_{\text{ref}}$. Thus $\bar{x}^c$ is often blurry and not ready for the subsequent target frame generation. To produce a high-quality fine-grained canonical head, we propose a novel recurrent canonical head generation network to combine transformed reference frames. As shown in Fig. 2, for each reference head frame $x_i$, we utilize head image encoder $E_H$ to encode $x_i$ as feature $h_i$ and also use its corresponding depth $d_i$ and pose $P_i$ to compute backward optical flow $f_{R_{\text{dec}}}^{i}$ (i.e., warping from canonical head $x^c$ to reference head $x_i$) by:

$$f_{R_{\text{dec}}}^{i}(u, v) = (u, v) - (u^c, v^c)$$

where $u^c, v^c$ is one pixel in $x^c$ and $(u, v)$ is the corresponding warped pixel in $x_i$ by applying Eq. (3) to each $(u^c, v^c)$ with $P_i$. Here we adopt the backward warping operation because it can be implemented efficiently in a differentiable manner using bilinear sampling [16].

We later apply flow $f_{R_{\text{dec}}}^{i}$ to warp reference head feature $h_i$ to $h^c_i$. Then we employ a Convolutional LSTM [34] module $\Lambda$ to aggregate all $h^c_i$ to generate the final canonical head feature $h^c$. A head image decoder $\Omega_H$ is finally used to decode feature $h^c$ to be canonical head $\hat{x}^c$. Then we can apply $F_D$ to $\hat{x}^c$ and combine the estimated depth $\hat{d}^c$ with $\hat{x}^c$ to form a 3D canonical head. This 3D head helps to fully utilize the multi-view appearance information provided by different reference frames. By applying Eq. (3) to $\hat{x}^c$ using the driving head pose $P_{\text{dri}} = \{R_{\text{dri}}, t_{\text{dri}}\}$, we transform $\hat{x}^c$ to the estimated driving head frame $\hat{x}_{\text{dri}}$. So we can train head image encoder $E_H$, image decoder $\Omega_H$, and ConvLSTM $\Lambda$ by the following head reconstruction loss function:

$$l_{\text{head}} = ||\hat{x}_{\text{dri}} - x_{\text{dri}}||_1$$

### 3.3. Attention-based Fusion Mechanism

Because of modeling with rigid transformation, the estimated canonical head $\hat{x}^c$ can only describe movements of the whole head. To model facial expressions as well as the appearance and motion of non-head regions such as the shoulder and background, we propose an attention-based fusion mechanism to combine $\hat{x}^c$, a randomly selected reference frame $s_{\text{ref}}$, and the motion and expression from driving frame $s_{\text{dri}}$ to produce the final target video frame $s_{\text{dri}}$. As Fig. 2 shows, we first transform the estimated canonical head $\hat{x}^c$ to driving head $\hat{x}_{\text{dri}}$ using the driving head pose $P_{\text{dri}} = \{R_{\text{dri}}, t_{\text{dri}}\}$. We then employ a frame encoder $E_F$ to represent $\hat{x}_{\text{dri}}$ and $s_{\text{ref}}$ as features $\hat{x}_{\text{dri}}$ and $s_{\text{ref}}$. We also design a flow and attention map predictor $\Phi$, to which $\hat{x}_{\text{dri}}, s_{\text{ref}}$ and $s_{\text{dri}}$ are fed, in order to estimate two backward warping feature flows $f_{\hat{x}_{\text{dri}} \leftarrow s_{\text{dri}}}$ and $f_{s_{\text{ref}} \leftarrow s_{\text{dri}}}$, and three attention maps $a_{\hat{x}_{\text{dri}}}, a_{s_{\text{ref}}}$ and $a_{s_{\text{dri}}}$. Then the combined feature $e_{\text{out}}$ can be computed by:

$$e_{\text{out}} = a_{\hat{x}_{\text{dri}}} \odot W(e_{\text{dri}}, f_{\hat{x}_{\text{dri}} \leftarrow s_{\text{dri}}}) + a_{s_{\text{ref}}} \odot W(e_{\text{ref}}, f_{s_{\text{ref}} \leftarrow s_{\text{dri}}})$$

$$+ a_{s_{\text{dri}}} \odot e_{\text{dec}}$$

where $W(\cdot, \cdot)$ is backward warping, and $f_{\hat{x}_{\text{dri}} \leftarrow s_{\text{dri}}}$ is used for warping estimated head $\hat{x}_{\text{dri}}$ to $s_{\text{dri}}$ for adding facial expression to $\hat{x}_{\text{dri}}$. $f_{s_{\text{ref}} \leftarrow s_{\text{dri}}}$ is used for warping reference frame $s_{\text{ref}}$ to $s_{\text{dri}}$ for providing the background and other appearance details. $e_{\text{dec}}$ is the intermediate feature from decoder for synthesizing unseen novel regions. Attention maps $a_{\hat{x}_{\text{dri}}}, a_{s_{\text{ref}}}$ and $a_{s_{\text{dri}}}$ are designed to indicate which parts in the feature maps can be kept and which parts should be masked out. The sum of the attention weights for corresponding pixels in the three attention maps should be equal to 1. Finally we employ a frame decoder $\Omega_f$ to decode feature $e_{\text{out}}$ to target frame $s_{\text{dri}}$. $s_{\text{dri}}$ should be identical to $s_{\text{dri}}$ and thus we can train the frame encoder $E_F$, flow and attention map predictor $\Phi$, the frame decoder $\Omega_f$ using the following frame reconstruction loss:

$$l_{\text{frame}} = L_{\text{rec}}(s_{\text{dri}}, s_{\text{dri}})$$

where $L_{\text{rec}}$ is the loss measuring the difference between reconstructed frame $s_{\text{dri}}$ and ground truth frame $s_{\text{dri}}$. Per [36, 38], we implement $L_{\text{rec}}$ using the perceptual loss [17] based on pretrained VGG network [39] and also add the equivariance loss [36] to stabilize the training.

### 3.4. Inference

During testing, given one subject video $S$ and one driving video $Y = \{y_1, y_2, \ldots, y_M\}$, we first randomly sample one reference image $s_{\text{ref}}$ and a set of reference frames $S_{\text{ref}}$ from video $S$, and estimate 3D canonical head $\hat{x}^c$ from $S_{\text{ref}}$ through our proposed recurrent canonical head generation framework. Then for each driving frame $y_i$ in $Y$, we adopt our attention-based fusion mechanism to combine $\hat{x}^c$, $s_{\text{ref}}$, and $y_i$ to generate the corresponding novel frame $\hat{s}_i$. The final target video $S = \{\hat{s}_1, \hat{s}_2, \ldots, \hat{s}_M\}$ is generated in a frame-by-frame manner.

**Pose-controllable novel view synthesis.** Our proposed Head3D can be easily adapted to the pose-controllable novel view synthesis task by manually inputting the desired pose transformation, $P_{\text{dri}} = \{R_{\text{dri}}, t_{\text{dri}}\}$, to generate the novel view $\hat{x}_{\text{dri}}$ from the canonical head representation, $\hat{x}^c$, rather than obtaining $P_{\text{dri}}$ from the driving frame $s_{\text{dri}}$. Then instead of inputting a $s_{\text{dri}}$ to the flow and attention map predictor $\Phi$, we input $\hat{x}_{\text{dri}}$ to the predictor, and the final output frame $\hat{s}$ will have the pose $P_{\text{dri}}$.

### 4. Experiments

#### 4.1. Datasets and Metrics

**Datasets.** We conduct comprehensive experiments on two public datasets: *VoxCeleb* dataset [27] and *FaceForensics* dataset [32]. The VoxCeleb dataset includes 22,496
videos downloaded from Youtube. To simplify the training, we only keep 7,500 videos for training and 400 videos for testing. The FaceForensics dataset contains 1,004 videos of news briefings from different reporters. We find that models trained on the VoxCeleb dataset can be generalized well to this new dataset. So we only randomly choose 150 videos for testing without any additional training. Following the preprocessing approach in [36], we crop videos in these datasets to mainly keep the head regions and resize all video frames to 128 $\times$ 128. Since the original videos in these datasets are long, we randomly select a short segment of 40 continuous frames from each video and use these selected short videos in our experiments.

**Metrics.** Following [36], we compute metrics based on two testing settings, self-reconstruction and cross-identity transfer. For self-reconstruction, we segment a video of the same subject into two non-overlapping clips. We use one clip as the subject video and the other as the driving video. In this setting, the driving video serves as ground truth. Similar to [7, 47], we compute the normalized mean $L_2$ distance and Learned Perceptual Image Patch Similarity (LPIPS) [53] metrics between self-reconstructed results and driving videos. For cross-identity transfer, which is more practical in real-world applications, subject video and driving video are of different subjects in this setting. As there is no ground truth available, we conduct a paired user study to compare our model with state-of-the-art methods. Specifically, we generated 100 videos for each baseline method on each dataset and paired them with videos produced by our model. We then invited 10 human evaluators to make judgments regarding the better video in each pair, considering aspects such as visual realism, motion accuracy, and identity consistency.

### 4.2. Implementation

**Model Implementation.** We employ a public pretrained face parsing network\(^1\) to extract the head regions (face and hair) from each video frame. For 3D head geometry learning, we implement the depth network $F_D$ with a similar architecture used in [20]. To stabilize the training, we add instance normalization layer [43] to the decoder of $F_D$. We adopt a similar architecture to HopeNet [33] for the pose network $F_P$ in our implementation. The original HopeNet only predicts the yaw, pitch, and roll of the head (i.e., $R$). To enable estimation of the 3D head translation $t$, we modified the final layer of the network. To accelerate the training, we initialize most parameters in $F_D$ and $F_P$ with pretrained models provided in [20] and [33]. For the recurrent canonical head generation, we choose the architecture in [17] to implement the head image encoder $E_H$ and decoder $\Omega_H$ with 2 downsampling blocks. We employ a one-layer ConvLSTM [34] to implement $\Lambda$. In our attention-based fusion mechanism, we also construct the frame encoder $E_F$ and decoder $\Omega_F$ using the same architecture as $E_H$ and $\Omega_H$. The flow and attention map predictor $\Phi$ is built based on the

---

\(^1\)https://github.com/zllrunning/face-parsing. PyTorch
Figure 4. Examples of generated talking-head videos using our proposed Head3D. For each block, Head3D synthesizes the new video (3rd row) with the appearance from the subject video (1st row) and motions from the driving video (2nd row).

flow predictor in MRAA [38]. We slightly change its architecture to enable the prediction of three attention maps.

As mentioned in Sec. 3, the whole training process of Head3D includes three separate stages. In the first stage, we train the depth network $F_D$ and pose network $F_P$ through 3D head geometry learning. In the second stage, we train the head image encoder $E_H$, head decoder $\Omega_H$, and ConvLSTM $\Lambda$ for the recurrent canonical head generation. We finally train frame encoder $E_F$, frame decoder $\Omega_F$, and flow and attention map predictor $\Phi$ for the attention-based fusion mechanism in the third stage. We set batch size as 5 videos and use the Adam optimizer [22] with $(\beta_1, \beta_2) = (0.5, 0.999)$ during all three-staged training. Unless otherwise specified, the number of reference frames is set to 5. During 3D head geometry learning, we train $F_D$ and $F_P$ for 10 epochs. The learning rate of $F_D$ and $F_P$ is $2 \times 10^{-4}$ and $2 \times 10^{-5}$ and drops by 0.1 at epoch 5. The balancing parameter $\lambda_{\text{sym}}$ and $\lambda_D$ in Eq. 4 are all set to be 0.1. When training recurrent canonical head generation, we train $E_H$, $\Omega_H$ and $\Lambda$ for 20 epochs with the learning rate of $2 \times 10^{-4}$ and drop learning rate by 0.1 at epoch 10. We train the attention-based fusion modules ($E_F$, $\Omega_F$ and $\Phi$) for 50 epochs with a fixed learning rate of $2 \times 10^{-4}$.

**Baseline Implementation.** We compare the proposed Head3D with three state-of-the-art motion transfer baseline models: 2D-based methods FOMM [36] and MRAA [35], and 3D-based methods DAGAN [14] and FaceV2V [46]. We follow the default settings in the methods’ original implementations wherever possible and retrain all the baselines using the same training videos on the VoxCeleb dataset as ours with the same $128 \times 128$ resolution.

### 4.3. Result Analysis

**Comparison with state-of-the-art methods.** We compare our Head3D with state-of-the-art (SOTA) methods under the self-reconstruction setting in Table 1. As shown in Table 1, Head3D achieves comparable or better performance when compared with the SOTA methods. While MRAA [38] performs better in most metrics under the self-

---

4959

4959

---

Due to the lack of official implementation, we implement FaceV2V with the code from https://github.com/zhanglonghao1992/One-Shot_Free-View_Neural_Talking_Head_Synthesis.
Figure 5. Examples of pose-controllable novel view synthesis. Each column demonstrates changing of a 3D rotation or translation parameter.

Table 1. Comparison of proposed Head3D with state-of-the-art methods under the self-reconstruction setting on VoxCeleb and FaceForensics datasets.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Method</th>
<th>$L_2$ ↓</th>
<th>LPIPS ↓</th>
</tr>
</thead>
<tbody>
<tr>
<td>VoxCeleb</td>
<td>FOMM [36]</td>
<td>0.0114</td>
<td>0.0856</td>
</tr>
<tr>
<td></td>
<td>MRAA [38]</td>
<td>0.0108</td>
<td><strong>0.0830</strong></td>
</tr>
<tr>
<td></td>
<td>DAGAN [14]</td>
<td>0.0123</td>
<td>0.0885</td>
</tr>
<tr>
<td></td>
<td>FaceV2V [46]</td>
<td>0.0186</td>
<td>0.0994</td>
</tr>
<tr>
<td></td>
<td>Head3D (Ours)</td>
<td>0.0113</td>
<td>0.0855</td>
</tr>
<tr>
<td>FaceForensics</td>
<td>FOMM [36]</td>
<td>0.0102</td>
<td>0.0543</td>
</tr>
<tr>
<td></td>
<td>MRAA [38]</td>
<td><strong>0.0075</strong></td>
<td>0.0449</td>
</tr>
<tr>
<td></td>
<td>DAGAN [14]</td>
<td>0.0106</td>
<td>0.0490</td>
</tr>
<tr>
<td></td>
<td>FaceV2V [46]</td>
<td>0.0119</td>
<td>0.0509</td>
</tr>
<tr>
<td></td>
<td>Head3D (Ours)</td>
<td>0.0079</td>
<td><strong>0.0442</strong></td>
</tr>
</tbody>
</table>

Table 2. User preferences in the paired study: our approach vs. state-of-the-art methods under cross-identity setting on VoxCeleb and FaceForensics datasets.

<table>
<thead>
<tr>
<th>Methods</th>
<th>VoxCeleb (%)</th>
<th>FaceForensics (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ours/FOMM [36]</td>
<td>72/28</td>
<td>68/32</td>
</tr>
<tr>
<td>Ours/MRAA [38]</td>
<td>57/43</td>
<td>59/41</td>
</tr>
<tr>
<td>Ours/DAGAN [14]</td>
<td>80/20</td>
<td>86/14</td>
</tr>
<tr>
<td>Ours/FaceV2V [46]</td>
<td>53/47</td>
<td>54/46</td>
</tr>
</tbody>
</table>

reconstruction setting, our proposed Head3D outperforms it under the more practical cross-identity setting as shown in Table 2. Under the self-reconstruction setting, we speculate that the advantage of using the 3D canonical head in Head3D may not be apparent, as the head motion and pose changes are limited due to the subject and driving videos being clipped from the same original video. When applied to the cross-identity motion transfer task, which typically involves larger head movements, Head3D benefits from leveraging the multi-view appearance information from the 3D canonical head, as is also shown in Fig. 3 and Fig. 4. More importantly, different from 2D-based FOMM and MRAA, Head3D can be easily applied to pose-controllable novel view synthesis, as shown in Fig. 1 and Fig. 5. Additionally, unlike 3D-based DAGAN and FaceV2V, the canonical head representation in Head3D is visually interpretable, as shown in Fig. 1 and Fig. 6.

Ablation Study. To analyze the effectiveness of each module in Head3D, we conduct an ablation study on the VoxCeleb dataset. Table 3 shows quantitative comparison results of the ablation study under the self-reconstruction setting. We first evaluate the effect of using different numbers of reference frames $N$. Since ConvLSTM can utilize different number of reference frames during training and testing, in our experiments, we specifically train a model with 5 reference frames and then evaluate its performance with different number of reference frames during testing. Compared with our final model with 5 reference frames, using fewer frames ($N = 1, 3$) generated worse results while increasing the number of frames ($N = 10$) can lead to better LPIPS but also longer inference time. So we choose $N = 5$ as our default setting. To evaluate the effectiveness of proposed recurrent canonical head generation, we compare our model with [Head3D w/ $\tilde{x}_c$], which employs the mean canonical head $\bar{x}_c$ instead of the $\hat{x}_c$ generated by the recurrent network to synthesize the final frames. One can observe that using mean canonical head $\bar{x}_c$ noticeably diminishes performance. The reason may be that $\bar{x}_c$ is generated by simply taking the average of all the canonical head images warped from reference frames, which makes it blurry and not capturing some important details. We also experiment with removing the canonical head input $\hat{x}_{dr}$ during attention-based fusion and evaluate this variant model [Head3D w/o $\hat{x}_{dr}$]. Without using the appearance informa-
Figure 6. Illustration of the effect of attention maps in our proposed attention-based fusion mechanism. The red regions indicate a higher degree of attention, while the blue regions suggest a lower degree of attention. “Deformed” refers to applying warping flow $f_{s\rightarrow d}$ to the subject frame and “Transformed” means applying driving pose $P_{dri}$ to the canonical head.

Table 3. Ablation Study under the self-reconstruction setting on VoxCeleb dataset.

<table>
<thead>
<tr>
<th>Methods</th>
<th>$L_2 \downarrow$</th>
<th>LPIPS $\downarrow$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Head3D ($N = 1$)</td>
<td>0.0117</td>
<td>0.0873</td>
</tr>
<tr>
<td>Head3D ($N = 3$)</td>
<td>0.0115</td>
<td>0.0880</td>
</tr>
<tr>
<td>Head3D ($N = 10$)</td>
<td>0.0116</td>
<td><strong>0.0839</strong></td>
</tr>
<tr>
<td>Head3D w/ $\hat{x}^{c}$</td>
<td>0.0117</td>
<td>0.0897</td>
</tr>
<tr>
<td>Head3D w/o $\hat{x}_{dri}$</td>
<td>0.0117</td>
<td>0.0872</td>
</tr>
<tr>
<td>Head3D ($N = 5$)</td>
<td><strong>0.0113</strong></td>
<td>0.0855</td>
</tr>
</tbody>
</table>

We also illustrate the effectiveness of our proposed attention-based fusion mechanism by visualizing some examples of attention maps in Fig. 6. As shown in Fig. 6, when a significant pose difference exists between the subject and driving frames, as in the first row, our model will assign higher attention values to the transformed canonical head to synthesize facial areas. In cases where the poses are more similar, such as in the second row, our model will combine information from both the subject frame and canonical head to generate the facial regions.

5. Limitation and Discussion

Head3D can achieve promising performance in most cases (see Fig. 4 and Supp. videos). However, it still suffers from several limitations. First, our current framework employs an off-the-shelf face parsing network to segment the head regions from video frames. Imprecise segmentation performed by the pretrained network may result in inconsistent or incorrect extraction of head regions, which could further adversely impact the estimation of the 3D canonical head (see the 1st row in Fig. 7). Second, when the subject video only provides a single side-view of the person, it can be challenging to generate a high-quality canonical head (see the 2nd row in Fig. 7). Currently, our proposed attention-based fusion mechanism can mitigate these limitations by assigning lower attention values to incorrect details of the canonical head, thereby reducing their influence on the final synthesized output. In future work, we will investigate the use of a more robust pretrained face parsing network or incorporate an unsupervised face parsing model into the current framework to enable end-to-end training. Recently, there has been a growing interest in high-resolution video generation [6]. We have provided a Supp. video at the $256 \times 256$ resolution, produced by our Head3D trained with different size parameters. In our subsequent research, we will also explore the video generation at the megapixel resolution such as $512 \times 512$.

6. Conclusion

In this paper, we present Head3D, a novel 3D-aware approach for transferring motion in talking-head videos. Head3D capitalizes on the multi-view appearance information inherent in a 2D subject video by estimating a 3D canonical head using a recurrent network. We introduce a self-supervised 3D geometry learning module to predict pose and depth map, and an attention-based fusion network to generate the final synthesized video. The explicit modeling of a 3D canonical head in Head3D allows for easy application to novel view synthesis tasks using user-provided pose inputs. Comprehensive experiments on two public talking-head datasets demonstrate the state-of-the-art video motion transfer capabilities of Head3D.
References


[26] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view syn-
