Video Content Swapping Using GAN

Tingfung Lau (tingfunl) [1] Sailun Xu (sailunx) [1] Xinze Wang (xinzew) [1]

1. Introduction

Video generation is an interesting problem in computer vision. It is quite popular for data augmentation, movie special effects, AR/VR and so on. With the advances of deep learning, many deep generative models have been proposed to solve this task. These deep generative models provide a way to utilize all the unlabeled images and videos online, since it can learn deep feature representations with unsupervised manner. And these models can also generate different kinds of images, which have great value for visual application. However, video generation still has long way to go. Directly generating a video from scratch could be very challenging since we need to model not only the appearances of objects in the video but also their temporal motion.

There are three main challenges for video generation (Tulyakov et al., 2018). Firstly, the system needs to build appearance models and physical motion models to get both the time and space information. Any model that doesn't work well will lead to the generated video containing objects with physically impossible motion. Secondly, even when objects perform some basic motion, there are so many variations considering time dimension, such as speech. Thirdly, motion artifacts are particularly perceptible for human beings.

In this project, we focus on a particular sub-task of video generation: video content swapping. Since visual signals in a video can be divided into content and motion, which represent objects and their dynamics respectively. The goal of our project is to swap the appearance object in an video but keep the motion being the same. For example, if we have a video clip of a professional dancer performing ballet, we would like to generate video of ourselves performing a fantastic ballet show, with the same pose as the professional dancer in the original video. Users without professional knowledge can also create video with the content as they want. All they need is an original video, and a photo including the content that they want to replace. We want to develop such a system based on some conditional generative models.

This project requires to solve three major problems. First of all, we need to train a constrained generative model to make sure the output is corresponding to the input objects. It is not enough to randomly generate a normal video. Secondly, the system should recognize between foreground and background images of the video. And it can automatically adjust the screen according to the interactions between objects and context, such as occlusion. Thirdly, with the increase of the number of people in the image, the system can still maintain real-time performance. (Cao et al., 2018)

A wide range of generative methods having been proposed for image generation, including Variational Autoencoders (VAE) (Kingma & Welling, 2013), Generative Adversarial Networks (GAN) (Goodfellow et al., 2014), and Conditional Generative Adversarial Networks (CGAN) (Mirza & Osindero, 2014). For video generation, there are several approaches using GAN, such as Temporal GANs conditioning on Captions (TGANs-C) (Pan et al., 2017), Motion and Content decomposed Generative Adversarial Network (MoCoGAN) (Tulyakov et al., 2018), and Semantic Consistent Generative Adversarial Network (SCGAN) (Yang et al., 2018). (Denton et al., 2017) propose an alternative way using disentangled auto encoder.

As we focus on video content swapping, our video generation should be conditioned on content input and pose in a reference video. We first extract the pose information from a video using a pre-trained human pose detection (Cao et al., 2018) trained on large scale data set. Then we will use a generative model to synthesize a video using the extracted pose and content image. This could be done using a disentangled auto encoder that learns to reconstruct an image based on content code and pose code or using a cGAN that conditioned on pose and content image. We will do experiments with these two methods on 3 data sets to find out the best method for solving this task. We plan to make up a tool for user to create their own synthesized video based on our method.

2. Related Works

2.1. Video Content Swapping.

The swapping problem in videos, specifically human faces, has been studied via various approaches. Traditional methods such as (Thies et al., 2018) and (Dale et al., 2011) deliver impressive results, and the former one even offers real-time video reenactment. However, these methods rely heavily on domain-specific knowledge and fine-tuned features specifically designed for face, and thus could not be generalized for non-face tasks.

A recent unpublished work based on deep network called DeepFake takes on this problem by training a shared encoder, and decoding using the target-specific decoder. This approach also gives impressive results and does not require domain-specific knowledge. However, it requires large number of training samples of the target face. Even worse, we need to train a different encoder for every different target face that we want to map to.

2.2. Generative Model

A huge amount of deep generative models have been proposed for image generation and video generation. There are two basic structures for these models, which is VAE and GAN. Kingma & Welling (2013) proposed a stochastic variational inference and learning algorithm, which can be applied to large datasets and intractable cases efficiently, (Rezende et al., 2014) introduced a recognition model to represent an approximate posterior distribution and uses this for optimisation of a variational lower bound. (Tulyakov et al., 2017) presented a principled framework, which is Hybrid VAE (H-VAE), to capitalize on unlabeled data. Goodfellow et al. (2014) proposed GAN to solve image generation by learning a generator to fool a discriminator for classifying whether image is real/fake.

2.3. Video Generations

Denton et al. (2017) proposes a different idea by taking into account the temporal coherence of a video, albeit not for the face-swapping task but for video prediction task. It disentangles the video representation into two components–*content*, which remains unchanged throughout a single clip or a duration of time, and *pose*, which captures the dynamic aspects of the clip and thus varies over time. Concretely, for the pose constraint, they introduce the adversarial loss for a discriminator network C and pose encoder E_p, and the i-th clip at t's frame x_i^t:

$$\begin{aligned}\mathcal{L}_{adversarial}(C) &= -\log(C(E_p(x_i^t), E_p(x_i^{t+k}))) \\ &\quad - \log(C(E_p(x_i^t), E_p(x_j^{t+k}))) \\ \mathcal{L}_{adversarial}(E_p) &= -\log(C(E_p(x_i^t), E_p(x_i^{t+k}))) \\ &\quad - \log(C(E_p(x_i^t), E_p(x_i^{t+k})))\end{aligned}$$

Effectively, the loss encourages the pose encoder to produce pose embedding that is indistinguishable for a discriminator.

Yang et al. (2018) divides the human video generation process into two steps, the first step is to synthesize the human pose sequence from an initial pose at the first frame, the second step is to generate the video frames from an human pose sequences and initial frame of the video. They train their pose generator and video generator using the GAN loss to classify whether the generated pose sequences and video are real or fake.

Tulyakov et al. (2018) also proposes the idea of replacing the video entity by swapping the content embedding. However, their content embedding is sampled once and assumed fixed for the entirety of the clip, which is overly constrained as there might be object occulation and rotation in the video, which might change the content embedding. Moreover, they could not specify a target content as they do not have the encoder part, as they are only relying on the unconditional distribution of the latent content embedding.

2.4. Pose Extraction

Pose extraction is an important problem for a wide range of scenarios, and people have proposed and optimized different kinds of models for different situation. There are two mainstream human body posture detection method, which is Top-down and Down-top. For Top-down method, it firstly detect a person and then estimate the person's posture. Therefore, the calculation time increases with the number of people in the picture increases. And if the people detection is wrong, the extracted pose must be wrong.

The Down-top method firstly detect the key points, and then determine the posture of each person through inference. The main challenge is how to deal with the interactions between the various key points.

OpenPose (Cao et al., 2018) is one kind of down-top method. It takes a color image as input and produces the 2D locations of anatomical keypoints for each person in the image. First of all, the system predicts 2D confidence maps for body part locations and 2D vector fields for part affinities using feedforward network. Then the confidence maps and affinity fields are parsed by greedy inference to output the 2D keypoints for all people in the image. The main logic is shown in the Figure.1

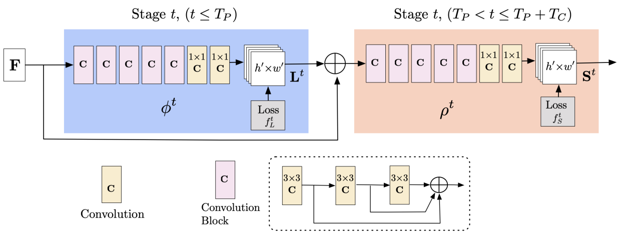

Figure 1. Architecture of the multi-stage CNN for OpenPose

3. Methods

Inspired by Denton et al. (2017), we want to solve this problem by learning a disentangled representation of video. The idea is that we use a auto-encoder like model to encode a video $\{\mathcal{I}\}_{t=1}^{T}$ to a content code $\{\mathcal{C}\}_{t=1}^{T}$ and a pose code $\{\mathcal{Z}\}_{t=1}^{T}$ and generate the video from the hidden code. The model have a pose encoder F_{pose} and a content encoder $F_{content}$ and a generator G, the hidden code and reconstruction is computed as

$$\begin{aligned}\mathcal{C}_i^t &= F_{content}(\mathcal{I}_i^t)\\ \mathcal{Z}_i^t &= F_{pose}(\mathcal{I}_i^t)\\ \hat{\mathcal{I}}_i^t &= G(\mathcal{C}_i^t, \mathcal{Z}_i^t).\end{aligned}$$

for each time frame $\mathcal{I}_i^t$ in a video sequence.

We want to ensure the learned content code and the pose code captures exactly the appearance and motion features respectively. For the pose code, since we focus on video with a human as its subject, so a natural idea is to utilize the off-the-shelf deep human pose estimators pretrained on large scale image data sets (Cao et al., 2018), as already mentioned in section 2.4. Therefore the F_{pose} would be a fixed pre-trained model throughout the training process. We could first extract $\mathcal{Z}_i^t$ in one pass in the preprocessing step and use it for later training.

For the content code, intuitively the content code should not vary much throughout the consecutive frames. So we use the following consistency loss to ensure the content encoding of frame t and $t + k$ for k random sampled from some time-window $[-w, w]$ is consistent(Denton et al., 2017).

$$\mathcal{L}_{consist} = \|F_{content}(\mathcal{I}_i^{t+k}) - F_{content}(\mathcal{I}_i^t)\|_2^2 \quad (1)$$

If we use the content code in time $t+k$ and the pose code in time t to reconstruct the image, the image should be close to frame t, therefore we impose the following temporal-shifted reconstruction loss on our encoder-decoder network as in Denton et al. (2017).

$$\mathcal{L}_{rec} = \|\mathcal{I}_i^t - G(F_{content}(\mathcal{I}_i^{t+k}), \mathcal{Z}_i^t))\|_2^2 \quad (2)$$

The whole model is shown in Figure 2. It could be trained on human video data set to learn the disentangled representation.

During inference time, we first compute the pose codes $\{\mathcal{Z}_i\}_{t=1}^{T}$ for a video using the pretrained human pose encoder F_{pose} and then compute the content code $\tilde{\mathcal{C}}$ for some an reference image for content swapping. Then we use the generator to produce images sequence $\tilde{\mathcal{I}}^t = G(\tilde{\mathcal{C}}, \mathcal{Z}_i^t)$ as the edited video. Since the encode disentangle the content and pose, so the resulting system will be able to achieve the goal of video content swap.

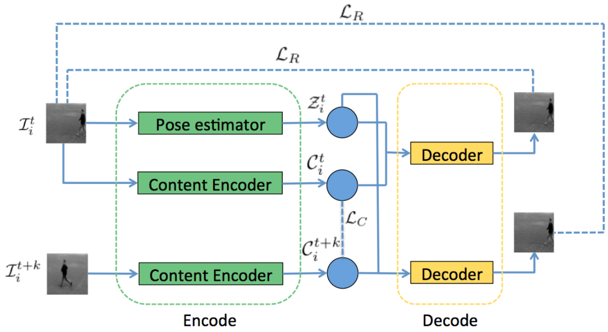

Figure 2. The whole pipeline of disentangled video representation learning.

The content encoder in the original work is a shallow 5 layer convolutional network to produce features at different resolution. The decoder uses 5 layer transposed convolution combine hidden code at different resolution to generate a image. We plan to use a deeper network to enhance the capacity of the encoder and decoder so that we can handle video with different content and background. For example, we can use the generative network with deep residual learning (Zhu et al., 2017).

Conditional GAN Using conditional GAN is another method to solve the video content swapping problem. In cGAN framework, we will train a generator G that takes an content image I and a human pose code Z. We can apply the pretrained human pose encoder F_{pose} on the faked output to ensure it generate the image at the desired poss. This is our first objective,

$$\mathcal{L}_{pose} = \|F_{pose}(G(I, Z)) - Z\|_2^2 \quad (3)$$

We also train a discriminator $D_{content}$ to ensure the output looks like real image and is conditioned on the two input. The $D_{content}$ will take a pair of image, and distinguish whether two real frames are in the same video, or a generated output and a conditional image. If the generator uses the content information, then the output should looks like in the same video.

$$\mathcal{L}_{GAN} = \log D(\mathcal{I}_i^t, \mathcal{I}_i^{t'}) + \log(1 - D(G(\mathcal{I}_i^t, Z), \mathcal{I}_i^{t'})) \quad (4)$$

where the pose code could be sampled either in the same video i or in some other video. This will make the generator handle multiple poses. GAN is known to generate image with high visual quality. It is interesting to see how GAN performs in this task compared to the auto encoder method.

4. Experiment

4.1. Data

The most important point for the dataset is that the videos and images should contain people in motion. Since in encoding process, pose estimator would extract the pose of people based on pretrained human pose frame. It will be difficult to extract the pose of other moving objects. Therefore, we searched for several datasets containing moving people to evaluate our model from different aspects.

Besides of the existing datasets, we also plan to download some other videos of different themes from YouTube. In this way, we can evaluate the performance of our model when applied in different scenes.

- **Tai-Chi.** It is a gine-grained action dataset, which consists of unconstrained user-uploaded web videos from YouTube and other data sources. There are about 4,500 Tai Chi video clips. For every clip, we cropped it so that the performer can be in the center and scaled it to 64×64 pixels. After the preprocessing, we use a pretrained human pose estimator to extract the pose of human in the video.

- **Weizmann Action.** Weizmann Action database contains 81 videos of 9 people performing 9 actions, such as jumping-jack and waving-hands. For this dataset, all the videos are scaled to 96×96.

- **UCF101.** This database is commonly userd for video action recognition. It includes 13,220 videos of 101 different action categories, such as basketball shooting, biking/cycling, diving, golf swinging and so on. This data set is very challenging due to large variations in camera motion, object appearance and pose, object scale, viewpoint, cluttered background, illumination conditions, etc. For each category, the videos are grouped into 25 groups with more than 4 action clips in it. The video clips in the same group share some common features, such as the same actor, similar background, similar viewpoint, and so on.

For some datasets, the scaled videos are really small, so it will be difficult to conduct a quantitative evaluation. Therefore, we will provide visual results in this condition.

4.2. OpenPOSE: Pose Extraction

We preprocess each video with the OpenPose. The return format is a list of key-point's 2-D coordinates: $(x_i, y_i)_{i=1}^{M}$, where i denotes the i-th key-point, e.g: head, and there are M key-points in total per person per frame. For the sake of simplicity, we will only be dealing with video with single subject(human).

In figure 3, the top row shows some consecutive frames of the yoyo video from the UCF-101 dataset and the bottom row shows the corresponding pose overlay outputs of OpenPose.

4.3. Disentangle

We did a preliminary experiment on disentangle auto encoder. We use the original implementation[1] to train the network on weizmann action dataset. Notice that when doing this experiment the pretrained human pose encoder haven't been done yet. The pose encoder is learnt using a GAN objective as in Denton et al. (2017).

The image is resized to 128×128, we trained for 60 epochs using Adam with default learning rate 0.001. The result is visualized in Figure.4. We can see that the network sort of learn to a disentangled representation, and the reconstruction using pose image and content image from different frames in the same video look close to the expected goal. However, the quality of generated image is low and the image is a little bit blurry and lack details. The decoder is not always generated image according to pose code and sometimes it will generate image with wrong pose or even two human in an image.

In this experiment, the pose encoder is learnt using an unsupervised objective. We believe that using a accurate human pose encoder pretrained on a large scale data set could further improve the fidelity of conditionally generation.

5. Future Work

- **Encoder and Decoder.** For the complete architecture as in shown in Figure 2: As mentioned earlier, we are using a pre-trained, fixed pose encoder. We are planning on using a ResNet(He et al., 2015) architecture for the content encoder and transposed convolutional network as used in DCGAN(Alec Radford & Chintala, 2015) for the decoder.

- **CGAN** Currently, we are using the ℓ_2 loss as a good heuristic metric for the consistency loss as well as the reconstruction loss. However, the ℓ_2 loss tend to produce a blurry result. Using CGAN as the generative model could improve the quality of image and make the results look the photo-realistic. We will be implementing CGAN as described in previous section and compare the generation results with ones from current ℓ_2 heuristic loss approach.

In the final report, we will implement the whole system and do experiments of video content swapping on the data sets we have mentioned. The qualified and quantitative result

[1] https://github.com/edenton/drnet

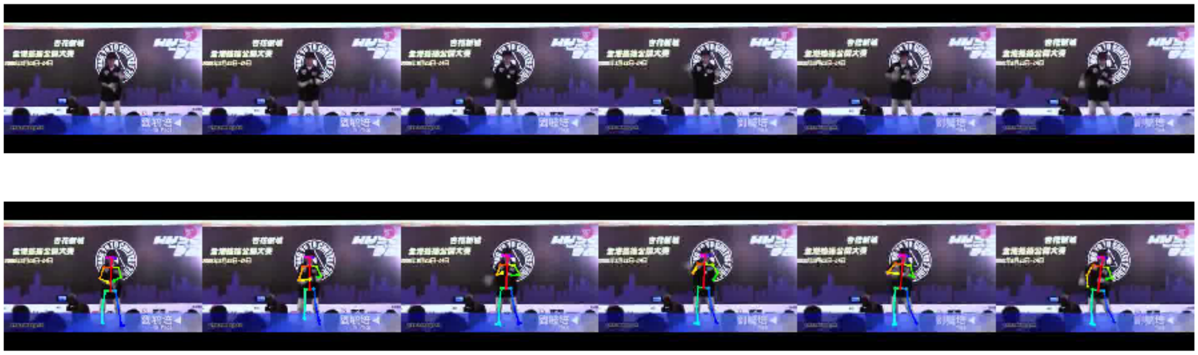

Figure 3. OpenPose output

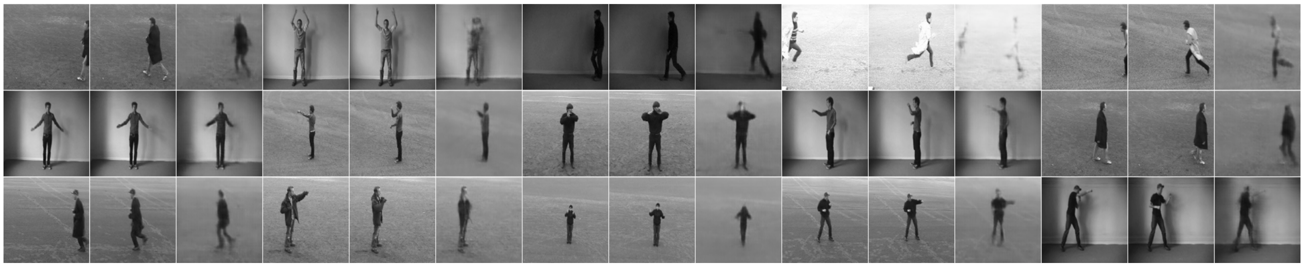

Figure 4. The results of disentangled auto encoder. In every 3 column by 1 row patch, the left image is used to compute the content code, the middle image is used to compute the pose code, and the right image is reconstructed from content code and pose code. It should look like the image in the middle by our content invariance assumption

will be presented in the final report.

References

Alec Radford, L. M. and Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

Cao, Z., Hidalgo, G., Simon, T., Wei, S.-E., and Sheikh, Y. OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields. In *arXiv preprint arXiv:1812.08008*, 2018.

Dale, K., Sunkavalli, K., Johnson, M. K., Vlasic, D., Matusik, W., and Pfister, H. Video face replacement. In *Proceedings of the 2011 SIGGRAPH Asia Conference*, volume 30, pp. 130:1–130:10, 2011.

Denton, E. L. et al. Unsupervised learning of disentangled representations from video. In *Advances in Neural Information Processing Systems*, pp. 4414–4423, 2017.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N. D., and Weinberger, K. Q. (eds.), *Advances in Neural Information Processing Systems 27*, pp. 2672–2680. Curran Associates, Inc., 2014. URL http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015.

Kingma, D. P. and Welling, M. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Mirza, M. and Osindero, S. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

Pan, Y., Qiu, Z., Yao, T., Li, H., and Mei, T. To create what you tell: Generating videos from captions. In *Proceedings of the 25th ACM international conference on Multimedia*, pp. 1789–1798. ACM, 2017.

Rezende, D. J., Mohamed, S., and Wierstra, D. Stochastic backpropagation and approximate inference in deep generative models. *arXiv preprint arXiv:1401.4082*, 2014.

Thies, J., Zollhöfer, M., Stamminger, M., Theobalt, C., and Niessner, M. Face2face: Real-time face capture and reenactment of rgb videos. *Commun. ACM*, 62(1):96–104, December 2018. ISSN 0001-0782. doi: 10.1145/3292039. URL http://doi.acm.org/10.1145/3292039.

Tulyakov, S., Fitzgibbon, A., and Nowozin, S. Hybrid vae: Improving deep generative models using partial observations. *arXiv preprint arXiv:1711.11566*, 2017.

Tulyakov, S., Liu, M.-Y., Yang, X., and Kautz, J. Mocogan: Decomposing motion and content for video generation. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1526–1535, 2018.

Yang, C., Wang, Z., Zhu, X., Huang, C., Shi, J., and Lin, D. Pose guided human video generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 201–216, 2018.

Zhu, J.-Y., Park, T., Isola, P., and Efros, A. A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2223–2232, 2017.