Unsupervised Domain Adaptation for Human Movement

Alexander Tsvetkov, Amro Abu-Saleh
Supervised By: Tom Zahavy, Oron Anchel
Electrical Engineering Department
The Technion - Israel Institute of Technology
Haifa 32000, Israel

Abstract

We present a novel and effective approach for video to video translation through generative adversarial networks. Given an input video of a person and a target video, our model generates a video of that same person mimicking the motion of the person in the target video. Our model can be used for various tasks, from "teaching" someone to dance like a skilled dancer to the generation of realistic videos of celebrities performing our biddings. Motion transfer between people in the video to video and image to image translation problems is a challenging task. Existing implementations such as CycleGan [Zhu et al. (2017a)] perform well when they only have to transfer the style of one domain to another, but they fail to do so when they have to handle geometrical and structural changes between the domains , specifically when it comes to human images. Our approach aims to tackle this obstacle by introducing a skeleton consistency loss to the training process in order to preserve the human body structural information during the transformation between the domains. Our results are shown as proof of concept for our approach in which we demonstrate our superiority over the existing implementations in the task of human pose transferring.

Introduction

Imagine you could film a short video of yourself, and make yourself move as any other character you can see on TV. Alternatively you could make a video of yourself dancing, and generate a character of your choice to mimic your moves, and it's only the beginning.

We propose a method to transfer motion between two human subjects, realistic or even avatars, in two different videos. The first video is the video of our subject, for instance a video of ourselves or of a celebrity. The second video would be the target video in which there is some motion we would like to mimic e.g. a video of a dancer or a martial arts master. Our method enables us to generate images and videos of the subject from the first video mimicking the moves of the target from the second video i.e. transferring the moves of the dancer from the second video to the person in the first video.

The implications of this developments are important due to the versatile usage possibilities of this method. It can be used for forgery by creating realistic videos of people which later could be posted online or used as fake evidence in court. It also could be used as a simulation tool for trying new clothes (Zhu et al., 2017b), or simply used as an image/video generator for various mediums(news agencies for A.I anchors, youtube content and etc). This solution could surely lead to a change in our definition for "real or fake" when it comes to images and videos.

The mapping of images is challenging, because the neural networks have to preserve some latent invariant information of the input and output domains. Moreover these invariants have to be maintained sufficiently during the training process of the networks in which all of the parameters could be changed. The CycleGAN model achieves decent results when there are enough similarities between the domains, for example their famous horse-to-zebra transformation. Though more problems arise when there are fundamental structural differences between the domains, namely when we try to map between two different people with different poses and distinct body structures. The results of these problems are often seen as the generation of inconsistent images with undefined shapes or people with modified/extra limbs . It seems that the core of this issue is the inability to preserve the geometrical structure of general images or the anatomical structure in human images.

In our approach we present a possible solution to the loss of structural information. We do it with the use of Realtime Multi-Person Pose Estimation network (Cao et al., 2017; Wei et al., 2016) which encodes the skeleton structures in the images into a heat map. This heat map contains a differentiable information of the person's key points, which we later induce as a novel loss to the training problem, constraining the generation to focus on body movement.

Similar but different approach for solving this problem has been shown by the CyCADA framework (Hoffman et al., 2017). It adapts representations at both the pixel-level and feature-level, while enforcing local and global structural consistency through pixel cycle-consistency and semantic losses, to overcome the loss of structural information during the mapping.

Our approach relies on the extraction of the skeletons heat maps by the Realtime Multi-Person Pose Estimation network, and that is why our training images must contain humans with clear and vivid poses. Moreover, We have also limited our method to deal with only one person in an image so there would be clear definition of the source and target objects (even though the pose estimation network can extract

multiple skeletons at once).

Our method works as follows; Given two videos we create two datasets by sampling images from each video, which are later used during the training phase of our model. The model receives two images at a time and extracts the skeleton heat map vectors from the images. An L1 norm is calculated between both of the heat maps and is added to the overall loss of the model. The whole process is repeated throughout the training loop. We've experimented on various data sets which included: our videos, ballerina videos , avatar videos from fortnite and dancers videos. We've applied multiple settings on the training process to achieve the best results.

Background

GANs

Generative Adversarial Networks are a class of artificial intelligence algorithms, implemented by a system of two neural networks contesting with each other in a zero-sum game framework.

We aim to learn the generators distribution p_g over the data. In order to do so we define a prior on input noise variables p_z(z), which is later used as an input to a mapping to data space as G(z;θ_g), where G is a representation of the generator function by a multi layer perceptron with parameters θ_g. Similarly we define the discriminator function by another multilayer perceptron $D(x;\theta_d)$ that outputs a single scalar. $D(x)$ represents the probability that x came from the real data rather than generated data distribution p_g. The discriminator D is trained to maximize the probability of correctly labeling the training samples and the generated samples from G. Simultaneously the generator G is trained to minimize $\log(1 - D(G(z)))$. The process may also be described as a minimax game with the two players being G,D and a value function of $V(G, D)$ which is denoted as:

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{data}}[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}[\log 1 - D(G(z))] \tag{1}$$

as shown in (Goodfellow et al., 2014).

These two networks are trained simultaneously with a mini-batch SGD algorithm in the context of the equations above which are used as loss functions respectively for each network. In theory the process stops when a Nash's equilibrium is reached in the minimax game, but in practice we decide on a fixed number of iterations for the SGD algorithm (usually the training is stopped when the results look good visually). Later on we discard the discriminator network altogether and use only the generator network to generate our images.

CycleGAN

CycleGAN is one of many variations of the GANs framework. The idea behind its implementation is to translate an image from a source domain X to a target domain Y in the absence of paired examples. This problem arises due to the lack of paired data(image pairs) which could be used to solve the problem in the supervised setting, which is inherently simpler than the unsupervised setting.

Let us denote two domains of images as X and Y (an example for these domains could be set as the domain of horse images X and zebra images Y). Basically the goal is to find a mapping(translator function) $G : X \rightarrow Y$ between the two domains of images X and Y such as the output $\hat{y} = G(x)|x \in X$ is indistinguishable from images $y \in Y$. In other words,it assumed there is some underlying relationship between the domains (for example, that they are two different renderings of the same underlying scene) and we seek to learn that relationship. We may train such mapping as described above by an adversarial network trained distinguish $\hat{y}$ from y ,thus inducing an output distribution over $\hat{y}$ that matches the empirical distribution $p_{data}y$. The optimal result would be a mapping G that translates the domain X to a domain $\hat{Y}$ which is distributed identically to Y.

Unfortunately such mapping does not guarantee a meaningful pairing between input x and output y because there is an infinite amount of such mappings. Moreover in practice this procedure often leads to a problem of mode-collapse ,where the all of the input images are mapped to the same output image.

To overcome this issues cycle consistency is added to the objective, which can be denoted as the condition of the existence of an inverse mapping F which is subject to $F(G(x)) = x$ and the subjection of G to $G(F(y)) = y$ (the mappings should be inverse of each other). The logic behind that is the same as in linguistic translation, namely if we translate a word from Hebrew to English and then back to Hebrew, we should get the original word.

This structural assumption is applied by training both G and F simultaneously with the addition of the cycle consistency $L1$ loss to enforce $F(G(x)) \approx x$, $G(F(y)) \approx y$ .The addition of this loss to the original adversarial losses of the vanilla GAN algorithm formulates the full objective on unpaired image to image translation in an unsupervised setting problem. As the extent of that two pairs of generators and discriminators are trained(one for G and one for F) with the addition of the cycle consistency loss :

$$\begin{aligned}\mathcal{L}_{\text{cyc}}(G,F) = \mathbb{E}_{x\sim p_{\text{data}}(x)}\left[\|F(G(x)) - x\|_1\right] \\ + \mathbb{E}_{y\sim p_{\text{data}}(y)}\left[\|G(F(y)) - y\|_1\right]\end{aligned} \tag{2}$$

Thus resulting in the overall objective function:

$$\begin{aligned}\mathcal{L}(G,F,D_X,D_Y) = \mathcal{L}_{\text{GAN}}(G,D_Y,X,Y) \\ + \mathcal{L}_{\text{GAN}}(F,D_X,Y,X) \\ + \lambda\mathcal{L}_{\text{cyc}}(G,F)\end{aligned} \tag{3}$$

Where $\mathcal{L}_{\text{GAN}}$ is the adversarial loss as seen in the GANs section;

$$\begin{aligned}\mathcal{L}_{\text{GAN}}(G,D_Y,X,Y) = \mathbb{E}_{y\sim p_{\text{data}}(y)}\left[\log D_Y(y)\right] \\ + \mathbb{E}_{x\sim p_{\text{data}}(x)}\left[\log\left(1 - D_Y(G(x))\right)\right]\end{aligned} \tag{4}$$

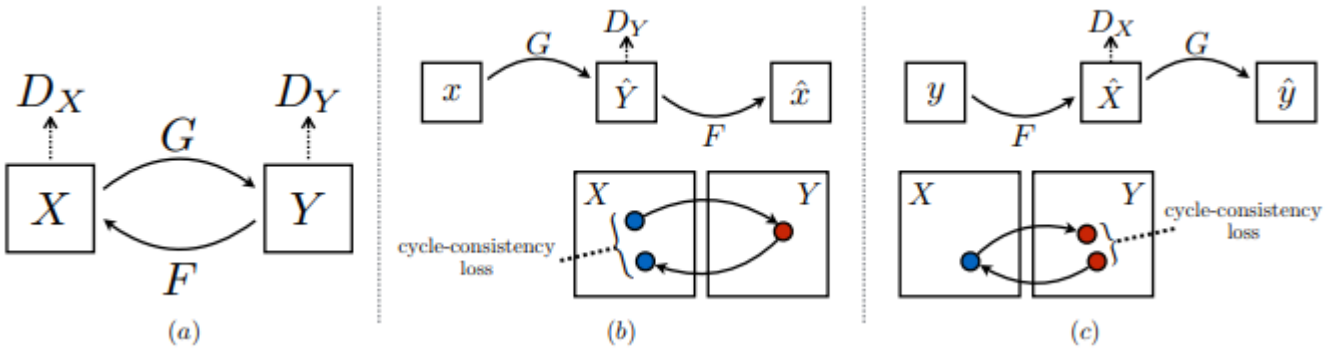


Figure 1: Our model contains two mapping functions $G : X \rightarrow Y$ and $F : Y \rightarrow X$, and associated adversarial discriminators D_Y and D_X. D_Y encourages G to translate X into outputs indistinguishable from domain Y , and vice versa for D_X and F. To further regularize the mappings, we introduce two cycle consistency losses that capture the intuition that if we translate from one domain to the other and back again we should arrive at where we started: (b) forward cycle-consistency loss: $x \rightarrow G(x) \rightarrow F(G(x)) \approx x$, and (c) backward cycle-consistency loss: $y \rightarrow (y) \rightarrow G(F(y)) \approx y$ (Zhu et al., 2017a)

A diagram for the CycleGAN model can be seen in figure 1

Realtime Multi-Person Pose Estimation

The Human pose estimation problem is the task of localizing anatomical key points and parts for a given input video or image which may consist of groups or individuals .This problem has its' set of unique challenges especially when there are multiple interacting individuals in the same spacial environment. The Realtime Multi-Person Pose Estimation framework which is used in our environment works as follows; The framework receives as an input a color image sized $w \times h$, and outputs the two dimensional locations of the anatomical keypoints for each person in the image. A feed forward network simultaneously predicts a set of two dimensional confidence maps of body part locations and a set of two dimensional vector fields of part affinities, which encode the degree of association between parts . The confidence maps and the affinity fields are parsed by a greedy inference to output the two dimensional keypoints for all people in the image (which are later used in our method). (Cao et al., 2017)

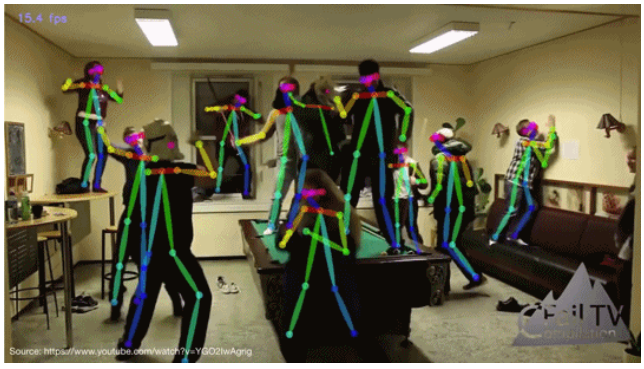


Figure 2: An example of application of pose estimation on an image

Method

Our framework is built on the CycleGAN implementation hence we are using the same network architecture: We have 2 generative networks; Each network contains two stride-2 convolutions, several residual blocks, and two fractionally strided convolutions with stride of $\frac{1}{2}$. We use 6 blocks for 128×128 images and 9 blocks for 256×256 and higher resolution training images, instance normalization is used. For the 2 discriminator networks we use 70×70 PatchGANs. (Zhu et al., 2017a)

In order to preserve the pose structure of the input images we use the Realtime Pose Estimation network, which will be denoted as the mapping $H : X, Y \rightarrow Z$ whereas X and Y are the input and output image domains and Z is the 2D key points domain. To preserve the pose structure of the people inside the frames during the transformation we augment our objective function by adding another condition: $H(x) = H(G(x)) \wedge H(y) = H(F(y))$. The new relation could be described by a block diagram as seen in figure 3. This condition is translated into the skeleton consistency loss:

$$\begin{aligned}\mathcal{L}_{\text{skel}}(G, F) = &\mathbb{E}_{x \sim p_{\text{data}}(x)}\left[\|H(G(x)) - H(x)\|_1\right] \\ &+ \mathbb{E}_{y \sim p_{\text{data}}(y)}\left[\|H(F(y)) - H(y)\|_1\right]\end{aligned} \tag{5}$$

Which is added to the objective function of the CycleGAN from equation 3 , thus resulting in:

$$\begin{aligned}\mathcal{L}(G, F, D_X, D_Y) = &\mathcal{L}_{\text{GAN}}(G, D_Y, X, Y) \\ &+ \mathcal{L}_{\text{GAN}}(F, D_X, Y, X) \\ &+ \lambda \mathcal{L}_{\text{cyc}}(G, F) + \lambda_{skel} \mathcal{L}_{\text{skel}}\end{aligned} \tag{6}$$

By adding this condition to our objective function we encourage the mappings G and F to maintain the structural integrity of the source image in the output image e.g. the people in the generated image would have the same skeleton

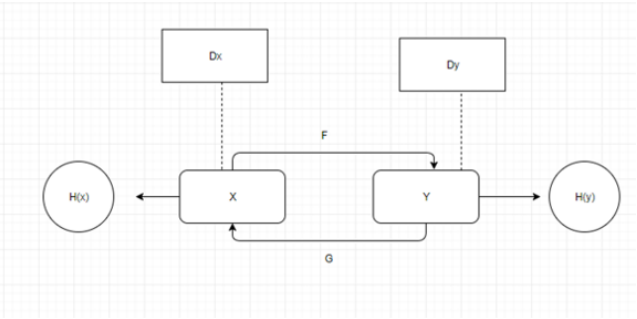


Figure 3: Block diagram

structure as in the original image. The logic behind that condition is that it would help us to pick a specific G and F out of all possible options , which will maintain the skeletons of the original images.

After the addition of this condition we train the networks to minimize the overall objective function as seen above. It is important to note that the Real time Pose estimation network is a pre-trained model, thus its' parameters H_θ are fixed during all of the training period as we relate to its' outputs as ground truth for our cause. We train the networks for 200 epochs with SGD(Adam optimizer) and a batch size of 1.

Just as in the CycleGAN paper we change our adversarial loss from equation 4 to be:

$$\begin{aligned}\mathcal{L}_{\text{GAN}}(G, D, X, Y) = \mathbb{E}_{x \sim p_{\text{data}}(x)}\left[(D(G(x)) - 1)^2\right] \\ \mathbb{E}_{y \sim p_{\text{data}}(y)}\left[(D(y) - 1)^2\right] + \\ \mathbb{E}_{x \sim p_{\text{data}}(x)}\left[D(G(x))^2\right]\end{aligned} \tag{7}$$

Because this loss is more stable during training and generates higher quality results(Mao et al., 2016).

Experiments

As mentioned in the introduction we always train on two distinct datasets which are created from two videos.The videos are chosen by the criteria of having a single person in them which performs some motion. Each data set consists of approximately 1500 images which are sampled at 60 fps from a given video. The images are cropped to the size of 256X256 at the input to the network ,so we don't have a consistent image size for the datasets.

Our most prominent results have been on datasets which had clear background and similar body structures between the people in each dataset. Examples for results from such data sets can be seen in the figures 4 and 5:

Figure 4: An example good results due to good background

Figure 5: An example good results due similar body structure

full video result can be found at:
https://youtu.be/TwsOvMZEWkk

During our training sessions we have noticed that some key elements in the dataset might prove crucial for the quality of our results. For instance we have discovered (by trial and error) that datasets with neutral backgrounds without artifacts have proven to generate the best result. This can be explained by the added complexity to the generator networks needed to generate the background . An example for this issue can be seen in figure 6.

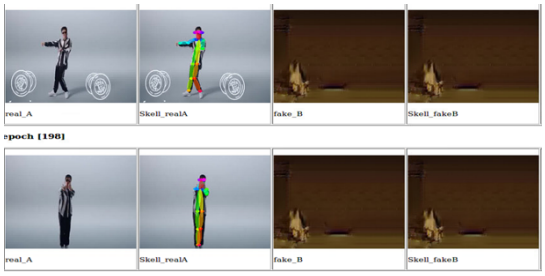
Another issue we've encountered is the phenomena of

Figure 6: An example of a result from a bad background with artifacts and vivid colors(batch size 1)

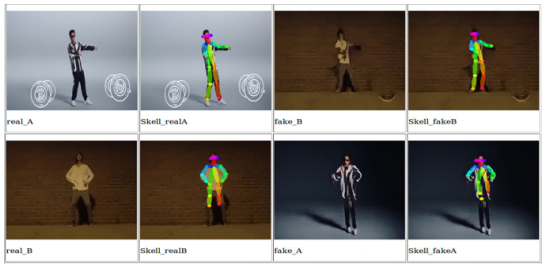
mode collapse. During various training sessions we have noticed that at some epoch our model "got stuck" on generating the same output image for every possible input. We suspect that one of the reasons for it was the mismatch between the scaling of the skeletons to the size of the picture e.g. one of the domains had people closer to the frame than the other, resulting in huge differences in the skeleton mapping. Same goes for instances where the different people had natural size differences in their structure. Results can be seen in figure 7.

Figure 7: An example of mode collapse due to bad skeleton scaling. The generator generated the same instance of the avatar image in the center under the fake label, even though the input images (under real label) have changed.

We've also tried experimenting with different batch sizes as well. The motivation behind this was the notion that increasing the batch size would improve our results(as seen in (Brock, Donahue, and Simonyan, 2018) . Though this might be true to general implementations of GANs, increasing the batch size for us resulted in poor results and has often lead to mode collapse. We assume that it has something to do with the structural differences between the images in the batch (Which could lead to a broad generalization of the skeleton reconstruction). Results for training in different batch sizes could be seen in figures 8a and 8b .



(a) Batch size 30(with mode collapse).



(b) Batch size 3

Figure 8: Results of training with batchsize 30 and batchsize 3 on the same dataset

We've came to the understanding that the value of the λ_{skel} hyper parameter had a grave impact on our results as well. When set too high it resulted in poor generated images due to its' dominance in the objective function (the cyclic and the adversarial losses were lesser by a sheer magnitude),which made the other losses ineffective. We've set this hyper parameter such value so our added skeleton consistency loss would be proportional to the other losses.

In most of our iterations we have noticed that it was difficult to reproduce the faces when generating images as seen in figure 6. We tried to tackle this issue by increasing the discriminator loss . The idea behind this was to force the generator to create more realistic images with clear faces, unfortunately we had no success in doing so(a possible solution could be seen in the form of a dedicated gan for faces only as seen in (Chan et al., 2018).

In conclusion we have came to the understanding that the best results are achieved by a combination of the choice of the right hyper parameters and a proper dataset.

Discussion

We've presented a possible solution to the human pose structural transfer in an unsupervised domain setting problem. Our approach has demonstrated superior results over the CycleGAN implementation when it came to the preservation of the skeleton structural integrity between input and output domains. Our results show that our method enables us to transfer the pose structure from the source to the target domain under certain limitations:

The choice if neutral and consistent background for the dataset is crucial to the success of our training process. When inconsistent background is used e.g. a patterned background or a background with artifacts our method is easily prone to failure. A possible solution could be achieved by the use of semantic segmentation masks to divide the foreground and the background as seen in (Wang et al., 2018).

Different choices of hyper parameters can lead to inconsistent results. Our model performs well when the batch size is set to 1 and the λ_{skel} parameter is set to a value which ensures the skeleton consistency loss to be proportional to all the other losses.

When combined into a video our generated frames induce noise and inconsistent positioning, thus resulting in a shaky video output. A possible solution would be to subject the generator to the creation of the next frame on the previous frame which would encourage a smooth transition between the generated frames.

Future directions

We plan to make our method more robust with the addition of several key elements in the future:
Frame consistency loss - With the use of sequential mini-batch sampling and the conditioning the generation of the next frame to the two previous frames, we can achieve smoothness and motion consistency in video generation.
Skeleton scaling - to counter the problem of the size difference between the skeletons and their proximity in the frames we aim to scale/normalize heatmap vector received from the Realtime Pose Estimation network as done in (Chan et al., 2018) .

References

Brock, A.; Donahue, J.; and Simonyan, K. 2018. Large scale gan training for high fidelity natural image synthesis.

Cao, Z.; Simon, T.; Wei, S.-E.; and Sheikh, Y. 2017. Realtime multi-person 2d pose estimation using part affinity fields. In *CVPR*.

Chan, C.; Ginosar, S.; Zhou, T.; and Efros, A. A. 2018. Everybody dance now.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. 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*. Curran Associates, Inc. 2672–2680.

Hoffman, J.; Tzeng, E.; Park, T.; Zhu, J.-Y.; Isola, P.; Saenko, K.; Efros, A. A.; and Darrell, T. 2017. Cycada: Cycle-consistent adversarial domain adaptation.

Mao, X.; Li, Q.; Xie, H.; Lau, R. Y. K.; and Wang, Z. 2016. Multi-class generative adversarial networks with the L2 loss function. *CoRR* abs/1611.04076.

Wang, T.-C.; Liu, M.-Y.; Zhu, J.-Y.; Liu, G.; Tao, A.; Kautz, J.; and Catanzaro, B. 2018. Video-to-video synthesis.

Wei, S.-E.; Ramakrishna, V.; Kanade, T.; and Sheikh, Y. 2016. Convolutional pose machines. In *CVPR*.

Zhu, J.; Park, T.; Isola, P.; and Efros, A. A. 2017a. Unpaired image-to-image translation using cycle-consistent adversarial networks. *CoRR* abs/1703.10593.

Zhu, S.; Fidler, S.; Urtasun, R.; Lin, D.; and Loy, C. C. 2017b. Be your own prada: Fashion synthesis with structural coherence. *CoRR* abs/1710.07346.