Domain Translation of Images Using Auto Encoders and Coupled GANs with Unsupervised Learning

Vigneshkumar Arunachalam¹, Harikumar Rajaguru ², Vinubama Krishnamoorthy³, Viswanathan Varadaraj ⁴, Pavithra Ravikumar⁵
Department of Computer Science and Engineering
¹,²,³,⁴,⁵Bannari Amman Institute of Technology, Sathyamangalam, Erode - 638401
vigneshkumara@bitsathy.ac.in, harikumarr@bitsathy.ac.in, vinubama.cs15@bitsathy.ac.in, viswanathanv@bitsathy.ac.in, pavithrardm@bitsathy.ac.in

Abstract. Domain translation of images is nothing but learning from marginal distribution across different to obtain a joint distribution from the listed domains. But there exist a limitless number of joint distributions, but we couldn’t infer that without additional assumptions. It is nothing but changing the entire scenario of the given input image by translating and reconstructing into an entirely new image but with the key features of the input image. Here for the domain translation of images from different domains in video, we are using Coupled-GAN’s for domain translation with making use of shared latent space.

Keywords- Domain Translation; video; Coupled-GAN’s;

INTRODUCTION

The model uses images from two different marginal domains, in which each domain is contrastingly different from one another. The main goal of this model is to recreate and translate the given image into another domain image [1]. Initially it was tested with digits in MNIST dataset for generating fake images, but recently proposed domain transformation network has led a huge path for the domain translation of images from which images could be generated from various domains as needed by humans. Since there is lot of things needs to be considered for domain translation. Learning an unique joint distribution from marginal distributions is little difficult, since we tried to localize the particular joint distribution from marginal distribution[2-4].Since there exists multiples of joint distributions, we are in need to find the particular joint distribution from marginal distributions, so here comes the cycle consistency constraint[5].

The cycle consistency constraint comes into role by because of the shared space assumption here we used. By using shared latent space assumption, the images in each frame could be either in each of one domain so that it could be transferred to the respective another domain [6]. By using GANs simply called as Generative Adversarial Networks which is emerged as a great deal in machine learning field. The GANs usually try to replicate the original image which is the given input image with some noise in that. The created image is similar to original image but not the exact original image [7]. Here GANs learn from the marginal distribution and tries to replicate the original image and tries to create the translated image of another domain [8]. The architecture of Domain Translation of frames in video is making use of Variation Auto Encoders which is used for domain translation and Coupled GANs for image reconstruction. This model is making use of the model VAE-GAN architecture [9,10]. The variation auto encoders relate the translated images with the input images in one domain.

The proposed model could be used for both supervised and unsupervised translation, where in supervised learning a pair of marginal distribution is given as an input and for unsupervised learning two different marginal distributional which is entirely different from one another is mapped together.
to form one complete reconstructed image with the features of another domain[11]. The model that we propose transfers a video in one domain to a video in another domain in which it has trained. Basically a video is nothing but a collection of images in each frames which are merged together to construct a video.

1. ARCHITECTURE DESIGN

The implementation of shared space (noted by S, here for our assumption) we has two domains. Let's assume that domains as I1 and I2, where each domains correspond to two marginal distributions. Here for video translation mostly the things in video are not similar, so it comes under unsupervised learning. For making joint distribution from marginal distribution we could infer lot of additional assumptions, so these assumptions are in the shared space(S).

![Figure 1. Layout of the Framework with Shared Space and Coupled-GANs](image)

Here the $E_1$ and $E_2$ are the domain image encoders, $G_1$ and $G_2$ are the domain image generators similarly $D_1$ and $D_2$ are the domain image discriminators. The generators \{G_1, G_2\} and encoders \{E_1, E_2\} together make the Variation Auto Encoders(VAE) configuration likewise generator\{G_1, G_2\} and discriminator\{D_1, D_2\} together make Generative Adversarial Networks(GAN). For any two combination of images let's assume \{i_1, i_2\} there is something called a shared code(i.e, the assumptions for unique features in one domain) s in a shared space S. The encoders \{E_1, E_2\} and generators \{G_1, G_2\} such that, for any pair of images (i_1, i_2) from the marginal distribution, we have $S = E_1(i_1) = E_2(i_2)$, that is in the shared space we have the unique features from both the distributions and inversely the generator again reconstructs the image $i_1 = G_1(s)$ and $i_2 = G_2(s)$.

With this model, there is a function which transfers style like $i_2 = f_{1\rightarrow2}(i_1)$ that maps from I_1 domain to I_2 domain that could be represented by the formulation $f_{1\rightarrow2}(i_1) = G_2(E_1(i_1))$. Likewise, $i_1= f_{2\rightarrow1}(i_2) = G_1(E_2(i_2))$. This then increases a learning problem $f_{1\rightarrow2}$ and $f_{2\rightarrow1}$ which is the real domain translation of images takes place within this functions. An important constraint for domain translation $f_{1\rightarrow2}$ and $f_{2\rightarrow1}$ to take place is the cycle-consistency constraint which has been discussed over few years and this is the combination for again reconstruction the original image from the input image $i_1 = f_{2\rightarrow1}(f_{1\rightarrow2}(i_1))$ and $i_2 = f_{1\rightarrow2}(f_{2\rightarrow1}(i_2))$. Preferentially, the framework proposed with shared space configuration and along with the cycle-consistency constraint the image translation is taking place.
2. FRAMEWORK

Our framework is the work on Coupled Generative Adversarial Networks (so called as coupled GANs) and Variation Auto Encoders (so called as VAEs) as it consists of several sub networks for domain image encoders \{E1, E2\}, domain image generator \{G1, G2\}, domain image discriminators \{D1, D2\}.

2.1 VAE GAN

The encoders and generators together \{E1, G1\} constitutes an auto encoder for I1 distribution and it is termed as VAE1, similarly \{E2, G2\} constitutes an auto encoder for I2 distribution is termed as VAE2. Let’s consider an input image i1 in I1 distribution, VAE1 first marks the input image i1 to the shared code s in shared space S through the encoder E1 and decodes some random unique features from the image to save for the purpose of domain translation for another domain and reconstructing the real input image with some noisy data through the generator G1. In a similar way does the VAE2. The components in shared space are conditionally independent and Gaussian unit variance [12]. The output of the encoder is a mean vector \(E\mu,1(i1)\) with some layers of encoders are connected for the constraint of weight sharing and the notation for the shared code s1 is given by \(q1(s1|i1) ≡ \mathcal{N}(s1 | E\mu,1(i1), I)\) where I is not a domain it is an Identity matrix.

The I1 domain reconstructed image is noted by the notation \(i1→1 = G1(s1 ∼ q1(s1|i1))\). Make a note that the notation used for the domain I(i.e, distribution) is \(q1(s1|i1)\) and since this is a random sample vector of \(\mathcal{N}(E\mu,1(i1), I)\) and the features are taken out from it. Similar way, the output of the encoder E2 is a mean vector \(E\mu,2(i2)\) with some layer are interconnected for the purpose of weight sharing and the notation for the shared code s2 is noted down by \(q2(s2|i2) ≡ \mathcal{N}(s2 | E\mu,2(i2), I)\). The I2 domain reconstructed image is \(i2→2 = G2 (s2 ∼ q2(s2|i2))\). The reparameterization metric for training of VAEs uses the back propagation method [13]. Let’s say η be some random vector with a Gaussian distribution for shared space: \(η ∼ \mathcal{N}(\eta |0, I)\). The sampling techniques for reconstruction of images are \(s1 ∼ q1(s1|i1)\) and \(s2 ∼ q2(s2|i2)\) which could be implemented as \(s1 = E\mu,1(i1) + \eta\) and \(s2 = E\mu,2(i2) + \eta\).

The framework has two Generative Adversarial Networks consisting of \{G1, D1\} is noted as GAN1 and \{G2, D2\} is noted as GAN2. In GAN1, for original images (\(i1\)) sampled from the I1 domain, the discriminator D1 should return the output as true, similarly the images which is generated by the generator (G1) for domain translation (\(i1→2\)), it should return the output as false. Generators are trained to create two kinds of images: 1) the images which is generated by the generator which is with some noisy data then that stream of reconstruction is noted as \(i1→1 = G1 (s1 ∼ q1(s1|i1))\) and 2) the images which is generated by the generator but with translated pixel values inside the image then that stream of translation is noted down by \(i2→1 = G1(s2 ∼ q2(s2|i2))\). Since the reconstruction of image process is a supervised training, it is enough to go with adversarial training for the translation step of image processing \{\(i2→1, i2→1\)\}.

3. NETWORK ARCHITECTURE

The network model of our framework, the encoders consists of three convolutional layers which is the front end and four residual blocks as the backend layers and the generators consists of four residual blocks as the front end layers and three deconvolutional layers as the backend layers. The discriminator consists of a six layers of convolutions which are stacked together with rely as the activational layers for first five convolutional layers. The last convolutional layer for discriminator
consists of a sigmoid function as activational layer. The last convolutional network has only two neurons to which lead to binary classification to say whether the translated image is from another domain or not.

4. TRAINING PROCESS

The last few layers of encoders \{E_1, E_2\} are given to the shared space as inputs from that shared space initial layers of generators \{G_1, G_2\} are linked. The generators creates two types of images one image is the reconstructed image which has some noisy data with compared to the fed input image and this image is given as input to the discriminators \{D_1, D_2\}. The job of discriminator and it should output true as the image is similar to the given input image. Another image generated by the generator is the domain translated image for which it should actually return false as it is a domain translated image.

Random noise generation for reconstruction of original images follows Gaussian distribution likewise the auto encoders VAE_1 and VAE_2 share weights by sharing the features of some intermediate layers. The training for VAEs should ensure that the images generated by \(VAEs(i^{1\rightarrow 1}, i^{2\rightarrow 2})\) are similar to the original image\{i_1, i_2\}. The training for GANs ensure that the translated image \(i^{2\rightarrow 1}, i^{1\rightarrow 2}\) and the domain images\{i_1, i_2\} are similar to each other. The property for the translation of images from one domain with one distribution to an image in another domain with another distribution by understanding the unique features in the opposite domain should be cycle consistent, is the sense that we translate, for e.g., if we add 2 plus 3, and then their reverse is 3 plus 2, we should come up with the same result 5. Computing mathematically, we are having two translators \(T_1 : I_1 \rightarrow I_2\) and another translator \(T_2 : I_2 \rightarrow I_1\), then \(T_1\) and \(T_2\) should be the converse of each other. Training both the mappings of translators \(T_1\) and \(T_2\) at same time, and applying a cycle consistency regression loss for training. By combining this cycle consistency loss with adversarial loss on domains \(\{I_1, I_2\}\) yields the complete objective for unsupervised learning in the field of image to image translation [14].

For solving the problems of learning the VAE\_1GAN\_1 and VAE\_2GAN\_2 for the reconstruction of images and the image translations, the cycle reconstruction cycles are,

\[
\text{min}(E_1, E_2) \text{min}(G_1, G_2) \text{max}(D_1, D_2) L_{VAE1}(E_1, G_1) + L_{VAE2}(E_2, G_2) + L_{GAN1}(E_2, G_1, D_1) + L_{GAN2}(E_1, G_2, D_2) + L_{C1}(E_1, G_1, E_2, G_2) + L_{C2}(E_2, G_2, E_1, G_1)
\]

Auto Encoders training aims at minimizing a reconstruction loss for the reconstruction images,

\[
L_{VAE1} (E_1, G_1) = \lambda_1 \text{KLdiv}(q_i(s_i|i_1)|p_i(s)) - \lambda_2 E_{s_1} \sim q_i(s_i|i_1)[\log p_{G_1}(i_1|s_i)]
\]

\[
L_{VAE2} (E_2, G_2) = \lambda_1 \text{KLdiv}(q_i(s_i|i_2)|p_i(s)) - \lambda_2 E_{s_2} \sim q_i(s_i|i_2)[\log p_{G_2}(i_2|s_i)]
\]

where the hyperparameters \(\lambda_1\) and \(\lambda_2\) initially initialized as 0.1 and 100 which takes the control of the weights shared in between the layers of the encoders and the KL divergence loss term is used to find the diversion of the shared code \(s\) in shared space \(S\) from the real distribution to identify the unique features. The regularizarion here we used leakly rely activation function which allows to easily sample out the unique features or key features from the image from the shared space. We have modeled the generators \(\{G_2, G_1\}\) using Laplacian distributions \(\{p_{G_1}, p_{G_2}\}\). Hence, reducing the negative log is same as for reducing the distance between the reconstructed image and the original image. The GAN functions are represented by,

\[
L_{GAN1}(E_2, G_1, D_1) = \lambda_0 E_{s_1} \sim P_{crit}[\log D_1(i_1)] + \lambda_2 E_{s_2} \sim q_i(s_i|i_2)[\log(1 - D_1(G_1(s_2)))]
\]

\[
L_{GAN2}(E_1, G_2, D_2) = \lambda_0 E_{s_2} \sim P_{crit}[\log D_2(i_2)] + \lambda_0 E_{s_1} \sim q_i(s_i|i_1)[\log(1 - D_2(G_2(s_1)))]
\]
The main functions of GAN\(_1\) and GAN\(_2\) are for real translation involves \(L_{GAN_1}(E_2, G_1, D_1)\) for translating an image from domain \(I_1\) to domain \(I_2\) and \(L_{GAN_2}(E_1, G_2, D_2)\) is for translating images from domain \(I_1\) to domain \(I_2\) which is the key objective function of GANs. This is used to verify that the translated images and the target specific domain images are interrelated to each other. The hyper parameter \(\lambda_0\) which is initially set as 10 has the control over the impacts of GANs functions. The auto encoders has modelled to use the cycle consistency constraint in which the cycle consistency loss is inferred by

\[
L_{C1}(E_1, G_1, E_2, G_2) = \lambda_3 KL(q_1(s_1|i_1) \| p_\eta(s)) + \lambda_3 KL(q_2(s_2|i_1^{1\rightarrow2}) \| p_\eta(s)) - \lambda_4 E_{s_2 \sim q_2}(s_2|i_1^{1\rightarrow2}) \log p_G_1(i_1^{1\rightarrow2}|s_2))
\]

\[
L_{C2}(E_2, G_2, E_1, G_1) = \lambda_3 KL(q_2(s_2|i_2) \| p_\eta(s)) + \lambda_3 KL(q_1(s_1|i_2^{2\rightarrow1}) \| p_\eta(s)) - \lambda_4 E_{s_1 \sim q_1}(s_1|i_2^{2\rightarrow1}) \log p_G_2(i_2^{2\rightarrow1}|s_1))
\]

The negative logarithmic term is used to ensure that the reconstructed image resembles the input target specific domain and the translated image resembles the output target specific domain. The KL divergence loss which is most off similar to cross entropy loss function, is used to find the shared codes are how much deviating from the input distribution for finding cycle reconstruction loss[15]. The \(\lambda_3, \lambda_4\) are the hyper parameters which are used to have a control over the weights of two different terms of KL divergence produced. The proposed framework is most off like a zero-sum game where this game consists of two players, the first player is the \(E_1, E_2, G_1, G_2\) and the second player is the \(D_1, D_2\). In order to win in the game the encoders and generators has to minimize their losses especially the cycle consistency losses.

5. EXPERIMENTS

We have analysed the various things by changing the parameters to the framework.

5.1 PERFORMANCE ANALYSIS

The optimizer we had used is the Adagrad optimizer with the weight decay and learning rate was initialized with 0.0001 and the setted hyperparameters are the momentums, batch-size. The momentums (beta1 = 0.7 and beta2 = 0.9) and the batch-size was given as five for both the domains, so at a time five images gets processed parallel for training at a time.

We had used a train set consisting of two domains in four folders, two folders for each domain. Each domain consists of one train set and test set. Train set is for training purpose of the model and the test set is for validation purposes. This setup is common for training all set of translations. There is no ground truth for this model since this is unsupervised training model and the performance metric is analyzed by only the visual perception of the humans. The total discriminator loss for the model in summer to winter translation. At some point the loss becomes stagnant, after this it starts over fitting the model.

![Figure 2. Loss for Difference between Translated Image vs domain image.](image-url)
6. RESULTS AND DISCUSSION

For summer to winter translation, summer is considered as domain I₁ and winter is considered as domain I₂. Summer has a train set of totally 1231 images, test set consisting of 309 images and winter has a train set of totally 962 images and test set consisting of 238 images with the image dimensions of width is 256 and height is 256. We have got the checkpoints for the summer to winter translation in 10K iterations and we have got the pretty much checkpoints in 120K for summer to winter translation and for winter to summer translation we get the checkpoints in 90K iterations.

Figure 3. Test Images for Summer to Winter Translated Images

In this images in top layer are the given input image(i₁), the images in middle are the reconstructed images(i₁→₁) and the images in bottom layer are the domain translated images(i₁→₂). This image is for summer to winter translation of images.

Figure 4. Test Images for Winter to Summer Translation

In this images in top layer are the given input image(i₂), the images in middle are the reconstructed images(i₂→₂) and the images in bottom layer are the domain translated images(i₂→₁). This image is for winter to summer translation of images.

In another experiment with day to night conversion, the day train dataset consists of 978 images, test dataset consists of 474 images and the night train dataset consists of 765 images, test dataset consists of 102 images with the dimensions of width 1280 and height 720. We get a pretty much checkpoints for day to night conversion in 90K iterations and for night to day conversion in 190K iterations.

Figure 5. Test Images for Day to Night Translation
In this image in top layer are the given input image ($i_1$), the images in middle are the reconstructed images ($i_{1→1}$) and the images in bottom layer are the domain translated images ($i_{1→2}$). This image is for day to night translation of images.

**Figure 6. Test Images for Night to Day Translation**

In this images in top layer are the given input image ($i_2$), the images in middle are the reconstructed images ($i_{2→2}$) and the images in bottom layer are the domain translated images ($i_{2→1}$). This image is for night to day translation of images.

7. CONCLUSION AND FUTURE WORK

This is a general model from which by changing a little hyperparameters this model could be used for various domain translations like apple to orange translation and vice versa, donkey to zebra translation and vice versa, cat to tiger translation and vice versa etc. The issues in the given framework are that it uses Gaussian space assumption which is a limitation. While running the inference with the videos and the issue faced is the quality of the frames generated by the generators for domain translation is of low resolution while compared with the given input image. These limitations could be resolved in upcoming works and additionally going to create a pipeline for detection models to detect the objects in the domain translated images. The proposed framework could be used for high resolution images also. The weight sharing constraint between encoders and generators and cycle consistency constraint has made our framework a good domain translator.

Deep learning has achieved an incredible success from the day it has evolved. In recent days, the usages of GANs for image generation become much better in generating images according to human perspective. The framework with the combination of GANs and VAEs which is for domain translation in videos will be useful as a good data augmenter. The quality of the images produced by GANs has been improved a lot from the day of its introduction. There are various types of GANs like vanilla GAN, Deep Convolutional GAN, Conditional GAN, Laplacian pyramid GAN and so on.

REFERENCES

[1] Ming-Yu Liu, Thomas Breuel, Jan Kautz, "Unsupervised Image-to-Image Translation Networks" NIPS 2017 Spotlight, arXiv:1703.00848 2017

[2] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. “Image-to-image translation with conditional adversarial networks”. Conference on Computer Vision and Pattern Recognition, 2017

[3] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, et al. “Photo-realistic single image super-resolution using a generative adversarial network”. Conference on Computer Vision and Pattern Recognition, 2017
[4] Z. Yi, H. Zhang, T. Gong, Tan, and M. Gong. Dualgan: “Unsupervised dual learning for image-to-image translation”. In proceedings of the International Conference on Computer Vision, 2017

[5] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. “Unpaired image-to-image translation using cycle-consistent adversarial networks”. International Conference on Computer Vision, 2017

[6] X. Yan, J. Yang, K. Sohn, and H. Lee. “Attribute2image: Conditional image generation from visual attributes”. European Conference in Computer Vision, 2016

[7] T. Kim, M. Cha, H. Kim, J. Lee, and J. Kim. “Learning to discover cross-domain relations with generative adversarial networks”. International Conference on Machine Learning, 2017

[8] A. Radford, L. Metz, and S. Chintala. “Unsupervised representation learning with deep convolutional generative adversarial networks”. In ICLR, 2016

[9] M.-Y. Liu and O. Tuzel. “Coupled generative adversarial networks”. Advances in Neural Information Processing Systems, 2016

[10] D. P. Kingma and M. Welling. “Auto-encoding variational bayes”. International Conference on Learning Representations, 2014

[11] L. A. Gatys, A. S. Ecker, and M. Bethge. “Image style transfer using convolutional neural networks”. In Computer Vision and Pattern Recognition (CVPR), 2016

[12] E. L. Denton, S. Chintala, R. Fergus, et al. “Deep generative image models using a laplacian pyramid of adversarial networks”. Advances in Neural Information Processing Systems, 2015

[13] D. J. Rezende, S. Mohamed, and D. Wierstra. “Stochastic backpropagation and variational inference in deep latent gaussian models”. International Conference on Machine Learning, 2014

[14] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. Lopez. “The SYNTHIA Dataset: A large collection of synthetic images for semantic segmentation of urban scenes”. Conference on Computer Vision and Pattern Recognition, 2016

[15] M. Mathieu, C. Couprie, and Y. LeCun. “Deep multiscale video prediction beyond mean square error”. In ICLR, 2016