Implementation of Multimodal Unsupervised Image to Image Translation

Kiyarash Aminfar kaminfar@gmu.edu
Pratik Mohare pmohare@gmu.edu
Yuqing Yang yyang47@gmu.edu

Abstract

This paper is dedicated to investigating multimodal image-to-image translation using unpaired image sets with an aim to generate a wide distribution of images in different domains from the one given as an input image without supervision or reinforcement input. Employing two generative adversarial networks, which is based on variational auto-encoders, we are able to extract the style and content code of two images in different domain and map the content code into a shared space. Then, in reconstruction time, any content code can be combined with multiple style codes to synthesize various target images in the goal domain. Several investigations have been performed to improve and evaluate the effect and contribution of various parameters, such as hyper-parameters, network architecture, loss function, optimization, and data augmentation. In addition to the less challenging image datasets, we implemented the method on more sophisticated ones, such as summer2winter, day2night, and undamaged2damaged. The application of this method could be in multiple fields, including autonomous vehicle, so an attempt to enhance visibility (by translating night-to-day image) have been made on videos which further proves the capability of this framework.

1. Introduction

Ever since the success of Neural Style Transfer using relatively simple Convolutional Neural Network models, a lot of interest has been developed in translating one image into the style of another. There have been considerable efforts to implement different approaches in the last few years to generate more realistic images as well as to retain maximum content information from the original images [10][5]. Many approaches have shown to be successful in certain areas but have failed in others. The major challenges come in 2 scenarios; lack of paired and labeled data, and generating wide distribution of images from a single style code.

After introducing the UNsupervised Image-to-image Translation (UNIT) [7], which assumes a shared latent space such that corresponding images in two domains are mapped to the same content, Multi-modal-UNIT (MUNIT) uses a combination of a VAE and GAN with an additional constraint of enforcing disentanglement in the learned latent space in the training process. This generates a wide range of transformations that cannot be achieved using prior approaches like simply adding noise. The implementation uses the idea of a Generative Adversarial Network to extract the content and style out of an image and cross-combine with the style and content from a second image, thereby developing two models simultaneously; model to convert from domain A to domain B and likewise another model to convert from domain B to domain A.

Finally, the learned weights from one of the models can be used on a new unseen image from a similar distribution to one of the domains set, along with the style code from the other domain, to regenerate the scene in the desired style. The MUNIT framework is also applicable in style-transfer and competing well with other similar papers focused on this task [6][2].

2. Related works

Generative adversarial networks (GANs). Generative adversarial networks (GANs) are a type of neural network architecture that is used for generating new data samples that are similar to a given training dataset. This is often used in image generation tasks, where the goal is to create new images that are indistinguishable from real ones. One specific application of GANs is in the area of image-to-image translation, where the goal is to translate an input image from one domain to another. This has many potential applications, such as colorizing black and white photos, converting photos from low-resolution to high-resolution, and transferring the style of an image to another while retaining its content. GANs have been shown to be effective at this task, and continue to be an active area of research in the field of computer vision.

Image-to-image translation. There are two main settings in the training of image-to-image translation: paired supervised samples and unpaired samples. In paired setting, the images of different domains are matched, meaning that for each input image, there is a corresponding output image that shows the same scene or object but in a different

domain (e.g., gray-scale to color). This allows the GAN to learn the direct mapping between the two domains and can produce very high-quality results. On the other hand, unpaired training involves using input and output images that are not directly matched. This means that the GAN must learn the overall distribution of the two domains rather than a specific one-to-one mapping. This can be more challenging, but can also be more flexible, as it allows the GAN to translate images that were not included in the training dataset. In general, unpaired training is more commonly used in practice, as it is often difficult or impractical to obtain a large dataset of paired images. However, the quality of the results may be lower compared to paired training.

Unsupervised Image-to-image translation (UNIT). The unsupervised Image-to-image translation (UNIT) framework [7] is a method for solving the image-to-image translation problem using a combination of two neural network models: a generative adversarial network (GAN) and a variational autoencoder (VAE). The VAE is used to learn a continuous latent space that captures the underlying structure of the data, while the GAN is used to generate high-quality images in the target domain. In the UNIT framework, the VAE and GAN are trained jointly, with the VAE providing the continuous latent space and the GAN providing the image generation capabilities. This allows the model to take advantage of the strengths of both models, resulting in better image quality and more flexible image translation capabilities. One key aspect of the UNIT framework is its ability to handle unpaired training data, which is often the case in real-world applications. This is made possible by the continuous latent space learned by the VAE, which allows the model to interpolate between different data points and generate novel images that were not seen during training.

Multimodal Unsupervised Image-to-image Translation (MUNIT). The Multimodal Unsupervised Image-to-image Translation (MUNIT) framework is an extension of the UNIT framework for image-to-image translation. Like UNIT, MUNIT uses a combination of a VAE and a GAN but adds an additional constraint to the training process to enforce disentanglement in the learned latent space. Disentanglement refers to the idea of learning a latent space where each dimension corresponds to a specific generative factor, such as the pose of an object or the color of an image. This allows the model to more easily control the generation of images, as it can manipulate specific factors independently of each other. In MUNIT, this is achieved by using a shared content latent space for the VAE and GAN, but with different encoders and decoders for each modality (i.e., each domain of the input and output images). This allows the model to learn a disentangled latent space that is shared across both domains, resulting in more controllable and interpretable image generation.

There are many scientific gaps in the field of image-to-image translation, including the need for more robust and versatile models, the development of new training strategies and loss functions, and the exploration of new applications and use cases. One of the main challenges in this field is to develop models that are able to generate high-quality images that are faithful to the original input, while also being flexible and able to handle a wide variety of input and output styles and content. Another important area of research is to develop new training strategies that can handle large amounts of data and complex image distributions, and that can improve the performance and generalizability of image-to-image translation models. The MUNIT paper addresses some of the scientific gaps in the field of image-to-image translation by proposing a new model architecture and training strategy that can handle multiple input and output styles and content. The authors present a multimodal unsupervised image-to-image translation model, which is able to perform image-to-image translation without the need for paired training data. It allows the model to learn to translate images between different styles and content in an unsupervised manner.

More specifically, [3] discusses the limitations of existing image-to-image translation techniques, which often assume a deterministic or unimodal mapping between the source and target domains. This means that these methods are not able to capture the full range of possible outputs for a given input, and may produce results that are overly simplistic or unrealistic. Additionally, even if the model is made stochastic by injecting noise, the network may learn to ignore it, which can further limit the model's ability to capture the diversity of possible outputs. This can be a significant limitation, as real-world data is often complex and varied, and a model that is unable to capture this complexity may not be able to produce high-quality results. The MUNIT framework addresses this limitation by assuming that the latent space of images can be decomposed into a content space and a style space, and images in different domains share a common content space but not the style space.

3. Methodology

With the main objective of the conditional distribution of corresponding images in the target domain, without seeing any examples of corresponding image pairs, the model needs to be designed to be able to evaluate the performance and update the parameters without supervision or labels. To achieve this, the idea of GAN is extended to generate two models, which each generate new images and try to recreate the original images. Thus through gradient descent, the model would ideally start converging to Nash equilibrium. But more practically, each pass gradually improves the image division into content and style, which can then be freely used for other images. To understand the working, we need to look at the architecture, and image interpolation with the

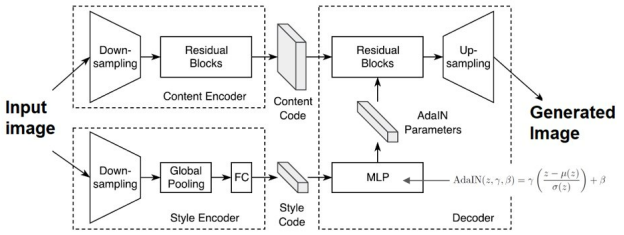
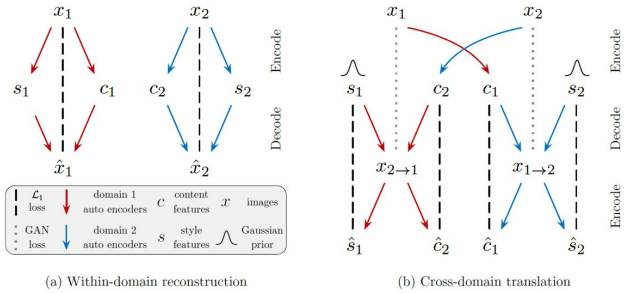


Figure 1. Auto-encoder architecture



(a) Within-domain reconstruction (b) Cross-domain translation

Figure 2. Image interpolation with respect to domain A and B, considering style and content code

loss function to train the model and hyper-parameters to define the importance of each of the loss components. Also, being a multimodal approach, selecting good datasets is an important step. Being unsupervised, the model needs to have pair of images from different domains but from similar distribution.

3.1. Network Architecture

The general idea is to use an image as an input to two correlated encode to get the style and content code separately and further use those and combine them to decode and create an output. This output result is imported to the discriminator along with another image which is from the target domain, and the discriminator decides whether it is fake or real. The architecture of the generator includes down-sampling, residual blocks in addition to convolutional layers. The "Base" network Architecture can be expressed as follows:

- Generator architecture:
• Content encoder: c7s1-64, d128, d256, R256, R256, R256, R256
• Style encoder: c7s1-64, d128, d256, d256, d256, GAP, fc8
• Decoder: R256, R256, R256, R256, u128, u64, c7s1-3
- Discriminator architecture: d64, d128, d256, d512

3.2. Hyper-parameters

During initial trials, the first purpose was to replicate the source paper. However, there were many challenges in terms of producing the results, the authors reported in their paper, such as tuning hyperparameters and training epochs. Hence, we decided to perform a wide parametrical study on different parameters, including loss's weight and down-sampling, residual blocks, and data augmentation. We were able to replicate similar qualitative results in the edge2shoes dataset. Later, using other datasets, as presented in section 3.5, the influence of the L1 loss function, Mean Square Error (MSE) loss function, and Kullback-Leibler divergence loss. Furthermore, it was observed that depending on the dataset and variability that is required, we can tune the loss weights to obtain more diverse output, in other words, get a multi-modal set of results.

3.3. Image Interpolation

With the architecture set, we need to train two models; one to translate images from domain A to domain B and another to translate images from domain B to domain A. To achieve this, the model needs to learn the mapping, which will separate the content part of the image, which is scene-specific, from the style part and determines the theme or the genre. In an unsupervised approach, the encoder cannot be trained directly; calculating the loss is the major challenge. The implementation aims at calculating losses in 2 phases: within-domain reconstruction and cross-domain translation.

The prior aims to calculate the loss from splitting the image into content and style and then using a decoder to interpolate the two into a new image to compare with the original input. The latter aims to use the content and style of the first image and interpolate with content and style from another image to generate a new pair of images. These images are ideally retaining the content within each of them and use the style mapping from the other. Using the same encoder decoder pair, the new images are separated into new contents and styles and cross-interpolated in an attempt to regenerate the original pair of input images. These differences between the final generated images with the original images in both situations contribute to the loss. This is further explained in the loss function.

3.4. Loss Function

As shown in related research articles, [9][11] [12], there should be a correlation between generator and discriminator losses. In the case of transferring images from/to the same space, other terms should be considered as well, such as:

$$\min_{E_1,E_2,G_1,G_2} \max_{D_1,D_2} \mathcal{L}(E_1,E_2,G_1,G_2,D_1,D_2) = \mathcal{L}_{\text{GAN}}^{x_1} + \mathcal{L}_{\text{GAN}}^{x_2} + \lambda_x(\mathcal{L}_{\text{recon}}^{x_1} + \mathcal{L}_{\text{recon}}^{x_2}) + \lambda_c(\mathcal{L}_{\text{recon}}^{c_1} + \mathcal{L}_{\text{recon}}^{c_2}) + \lambda_s(\mathcal{L}_{\text{recon}}^{s_1} + \mathcal{L}_{\text{recon}}^{s_2})$$

where λ_x, λ_c, λ_s are weights that control the importance of reconstruction terms.

The model is jointly trained on the entire set of encoders,

decoders, and discriminators to generate close to optimal images. These generations should, in theory, be indistinguishable from real images. So the final loss is calculated as a weighted sum of the adversarial loss and bidirectional reconstruction loss. Adversarial loss focuses on distinguishing real from translated images and trains the generator to generate better images which can fool the discriminator to term the generated images as real, while the discriminator, in parallel, learns to distinguish real from generated images in the target domain. Bidirectional reconstruction loss, as explained earlier, focuses on training the pairs of encoder and decoder to learn the inverses, which focus on the objective of direct image reconstruction, which is within the domain, and latent reconstruction, which is cross-domain. Further image translation can be controlled and tuned using the lambda multipliers. These parameters determine the importance of each individual loss type and train the model in the needed goal direction. Using higher lambda for style encourages diverse output given different style codes. Lambda for content encourages the translated images to preserve the semantics from the input image.

3.5. Datasets

Edge2shoes/handbags. The proposed dataset in the source paper was edge2shoes which is obtained from reference [4].

Summer2Winter-Yosemite. We used summer2winter-Yosemite [3] containing 3253 summer photos and 2385 winter photos. Thanks to data augmentation and addition, we were able to equalize the number of photos of them and have a total of 8k images. We had to manually remove some outliers and some photos that were taken from unrelated subjects (such as humans) or were taken during nighttime.

Night2Day. We use the dataset provided by [1], which contains 17.8k images of night and day scenes of different places taken at different times of the day and in different lighting. We train a model for the night-to-day using unpaired images.

Damaged2Undamaged. We then used images taken from [8] from the damaged and undamaged surfaces. It contains a total of 40k images divided into two classes.

4. Experiment

The first trial on the edge2shoes dataset, after 200 epochs, with the proposed architecture aligned with the original paper. Having that completed, with the goal of comparing different parameters, we tried to observe and understand the behavior and influence of each one on the output. The next step was to apply and tune the network on the summer2winter dataset. Figure 3 shows the final output of synthesized images, which transfer from summer2winter. Here, it is worth mentioning that although the original dataset had images from different angles, subjects, and scenes (including mountain, and jungle), there were some outliers and some unusable data for our task in the dataset, which was removed manually. For the sake of enhancing the dataset we have, we performed common data augmentation, including rotation, and vertical flip.

Figure 3. Result from Summer2Winter

5. Results

The investigation on various network-dependent parameters such as loss function, optimizer, and network layers was tested on the summer2winter dataset. The best combination is achieved by assigning 1, 10, 1, 10, and 10 to losses' weight: lambda gan, lambda id, lambda style, lambda cont, and lambda cyc, respectively.It was observed that increasing residual layer numbers is not useful and leads to a decrease in the overall quality of reconstructed photos. On the other hand, a similar approach was applied to the undamaged2damaged dataset and tuning the network based on that. The results of generated photos are depicted in Figure 3 and 4.

Regarding the network-dependent parameters, as it was

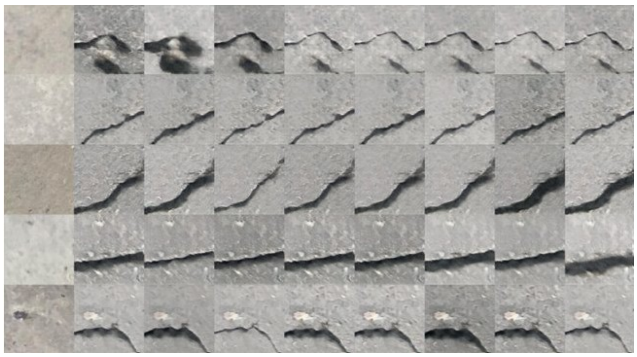


Figure 4. Result from DamagedUndamaged

Figure 5. Example results of Night2Day translation

anticipated, increasing the size of the input image is beneficial. However, residual layers should not be more than three layers, although it leads to additional training costs/time. A comprehensive evaluation of various loss formulations revealed that MSE has a higher priority compared to the L1 loss function. Nevertheless, the KLDiv Loss function, having low generator and discriminator losses, did not produce meaningful results.

We also performed experiments on the night-to-day translation dataset. The example translation results are shown in Figure 5. It can be seen that the model is able to capture the full range of possible outputs for a given input image. The training dataset contains images taken at different times of day, in different lighting conditions, and during different seasons, which contributes to the model's ability to generate a wide range of styles. Notice that the translated images show a diversity of details, such as the leaves on a tree, the shape of the clouds in the sky, and the changing colors and light that represent different times of the day. Overall, the model demonstrates a level of accuracy and flexibility in its translations from the night domain to the day domain.

Given the success of the summer2winter dataset, to add more variations, we merged the summer2winter dataset with the Night2Day dataset and retrained the model. The results can be found in Figure 6.

6. Conclusion

We looked into and expanded the multimodal unsupervised image-to-image translation (MUNIT) framework and studied various network-dependent parameters utilizing various challenging datasets. The reconfigured network produced distinctive and high-quality images by fusing together two conceptually related but stylistically distinct domains.

While the MUNIT model has been shown to be effective at solving the image-to-image translation problem, there are some limitations to its approach. One potential limitation is that the disentanglement constraint added in MUNIT may not always be appropriate or beneficial for all tasks. For example, in some cases, it may be desirable for the model to learn a more complex and interconnected latent space, rather than a disentangled one.

Another potential limitation is that MUNIT, like other GAN-based models, can be difficult to train and may not always converge to a satisfactory solution. GANs are known to be sensitive to hyper-parameters and may require careful tuning and regularization in order to work well.

Additionally, MUNIT and other GAN-based models can suffer from mode collapse, where the model only learns to generate a limited subset of the data distribution, resulting in poor diversity of the generated samples. This can be addressed using techniques such as minibatch discrimination, but it remains a challenge in GAN-based models.

7. Future Work

In future work, it may be interesting to experiment with different network architectures used in the model. One potential approach could be to use larger filters for the convolutional layers, based on the size of the input images. This could help capture more detailed information from the images and potentially improve the performance of the network. Another potential approach could be to use max pooling instead of average pooling. This could help retain more information from the images and allow the network to learn more robust features. Additionally, using the tangent hyperbolic activation function in the network could improve its ability to capture nonlinear relationships in the data. Overall, these modifications to the network architecture could help improve its performance on the dataset and provide more accurate results.

Another possible direction for future works is to explore the use of different loss functions, such as the feature matching loss used in the Few-Shot Unsupervised Image-to-Image Translation (FUNIT). In FUNIT, the feature matching loss is used in the training of the generator

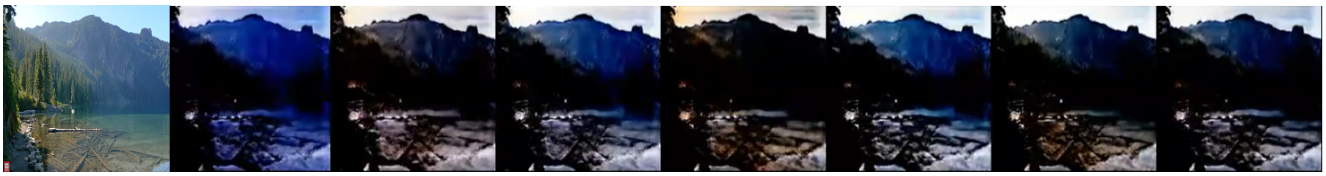


Figure 6. Result from Summer2Winter and Night2Day merged

network, measuring the similarity between the features extracted from the generated images and the features extracted from the real images. The goal of the feature matching loss in FUNIT is to encourage the generator to produce images that are similar to the real images in terms of the high-level features they contain, which can help to improve the overall quality and realism of the generated images.

Finally, style data from multiple images could be used to train a multi-layer perceptron (MLP) on the style data from multiple images. The MLP could be trained to learn the common features and patterns in the style data and to generate a new style matrix that combines these features in a way that preserves the individual styles of the input images. This new style matrix could then be used in the model to generate images with a more diverse range of styles. This approach could potentially improve the quality of the generated images by allowing them to have a more dynamic and varied style.

References

[1] Night2day, 2022. Data uploaded by tamirpuzanov. https://www.kaggle.com/datasets/tamirpuzanov/night2day.

[2] X. Huang and S. Belongie. Arbitrary style transfer in real-time with adaptive instance normalization, 2017.

[3] X. Huang, M.-Y. Liu, S. Belongie, and J. Kautz. Multimodal unsupervised image-to-image translation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 172–189, 2018.

[4] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks, 2016.

[5] T. Kim, M. Cha, H. Kim, J. K. Lee, and J. Kim. Learning to discover cross-domain relations with generative adversarial networks, 2017.

[6] Y. Li, M.-Y. Liu, X. Li, M.-H. Yang, and J. Kautz. A closed-form solution to photorealistic image stylization, 2018.

[7] M.-Y. Liu, T. Breuel, and J. Kautz. Unsupervised image-to-image translation networks. *Advances in neural information processing systems*, 30, 2017.

[8] F. Özgenel. Concrete crack images for classification. 2019.

[9] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, and B. Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans, 2017.

[10] Z. Yi, H. Zhang, P. Tan, and M. Gong. Dualgan: Unsupervised dual learning for image-to-image translation, 2017.

[11] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks, 2017.

[12] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.