11-785 PROJECT REPORT: TURNING DAY INTO NIGHT

Chi Liu
chil1@andrew.cmu.edu

Raphael Olivier
rolivier@andrew.cmu.edu

Teven Le Scao
tlescao@andrew.cmu.edu

Jean-Baptiste Lamare
jlamare@andrew.cmu.edu

ABSTRACT

The photography and cinema industries rely heavily on techniques to turn day pictures into night, called day-to night tricks. These techniques usually required special shooting conditions, and/or an image editing software that could not work without some manual intervention. The recent improvement of machine learning based computer vision techniques has allowed the development of algorithms that could turn day into night automatically.

1 INTRODUCTION

Given an outdoor picture, our objective is to change the perceived time of the day of the picture. Many movies, such as Mad Max : Fury Road (2015) (figure 1), use such techniques in order to shoot with the optimal lighting conditions of day-time but still achieve the desred night-time mood.

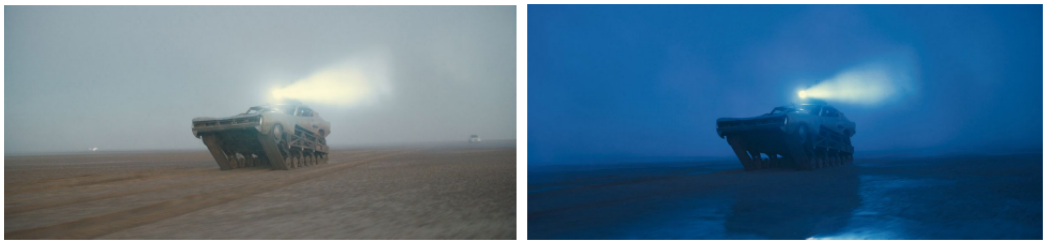

Figure 1: Original image from the shooting of of Mad Max : Fury Road (2015) and corresponding shot in the movie

In this report we focus on turning day into night, although a similar pipeline can be applied to the opposite translation, or many more.

2 RELATED WORK

Works on style transfer started a few decades ago with non-neural techniques. Efros & Freeman (2001) introduced "image quilting" which synthesizes a new image by stitching together small patches of other existing images. Neumann & Neumann (2005) transfers color style through Hue-Lightness-Saturation histogram matching. Shih et al. (2013) focus on a continuous day-night matching objective that is very close to ours, and proposes an edge aware locally affine RGB mapping. We try here to follow the same objective but with a deep learning-based model.

Neural style transfer arose around 3 years ago. Gatys et al. (2015) introduces a deep learning model based on Convolutional Neural Networks (CNNs) to capture features defining a painter's style and transfer it to a picture. It creates a new picture by jointly minimizing a per-pixel content loss function (compared to the original picture) and a style loss function (compared to the artwork) from a white noise image. If this works well in an artistic context, creating realistic pictures is much more challenging. In particular, the per-pixel loss functions do not seem appropriate as they do not represent how realistic the picture is. Johnson et al. (2016) proposes perceptual losses for style transfer, using high level features in the loss function instead of per-pixel results. These functions are supposed to better represent whether the synthesized image is able to fool human perception.

The last recent evolution in the field of style transfer came with the development of Generative Adversarial Networks (Goodfellow et al. (2014)). The issue of finding the appropriate loss function is partly solved by these GANs who lean their own loss function. Isola et al. (2017) proposes the "pix2pix" system for image-to-image translation, that uses conditional GANs (Mirza & Osindero (2014)) conditioned on the input image. Since GANs learn their own loss function, they also present their software as a general-purposed model that can be trained on many different datasets, including a day-night matching dataset. However, one of the main problems with this kind of tasks being the lack of paired training data (especially when the result of the generation is a new artistic result), Zhu et al. (2017) present cycle-consistent GANs based on pix2pix to train on unpaired data. A classic GAN would not work efficiently on unpaired data, because the mapping would be highly under-constrained. To fix this problem, in addition to the standard generator that works as a translator, the authors trained a second inverse translator, and added in the loss function that the composite functions of both translators should be close to the identity. Liu et al. (2017) chooses a different strategy to deal with the under-constrained model. They assume there exists a shared latent space both classes can be mapped to and train a Variational Auto Encoder (VAE) - GAN to learn this representation and generate from it.

3 DATASET

There are a few existing day-night matching datasets. However, they are not all created for deep learning and are often too small or not exploitable in our case.

The one we use is a vehicle dataset for vision-based place recognition from Milford & Wyeth (2012). This dataset is used in SeqSLAM for visual route based navigation in day and night with manually grounded truth frame correspondences. This dataset contains pairs of: sunny day condition of the route and rainy night of same route at suburb of Alderley, Queensland (figure 2). There are 14,607 pairs of day-night images. However, many image pairs are redundant (since they are extracted from a video, two pairs of images in a row are very similar). We thus decide to focus on 1/50th of the images (we keep one every 50 images), and focus on making our model robust against a more challenging dataset : indeed, as the night-time videos were shot in rainy conditions, they incorporate a lot of noise and hard-to-predict reflections.

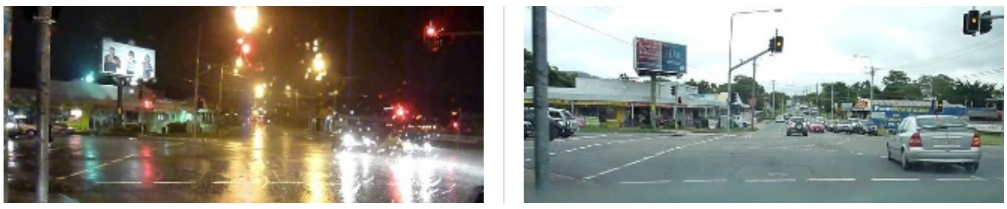

Figure 2: Dataset from SeqSLAM Milford & Wyeth (2012)

4 BASELINE MODEL

4.1 PRINCIPLE OF THE IMAGE-TO-IMAGE GENERATION MODEL

We are focusing on the dataset from Milford & Wyeth (2012) and therefore on a supervised/paired approach. Our reference is Isola et al. (2017). We describe the key elements of this approach below.

The network is a conditional GAN (CGAN), with a few important differences. CGANs are usually used for generation, under the weak supervision of a label ; for example we can apply them on MNIST, to learn how to generate images of a specific digit : for a pair $(x_{real}, y) \in [0,1]^{3\times28\times28} \times \{0, ..., 9\}$, the generator takes y as input and tries to generate an x_{fake}, using both y and a randomly sampled latent variable. The discriminator takes y and x (either generated from y or real with label y) and tries to determine if x is real or fake.

Here the process is a little bit different as it is an image-to-image translation task : the pair (x_{real}, y) consists of a night image x_{real} and the corresponding image during the day y (figure 3). In other words we generate night images under (very strong) supervision by a day image. Here sampling a latent variable for generation seems irrelevant, as the information needed for night generation is contained in the day image. However to learn a network we still need randomness in the generation process (otherwise we would not learn a distribution but just the association of the training pairs). Like Isola et al. (2017) we used randomness in the form of strong dropout after most layers (figure 4).

Figure 3: Source day and night images

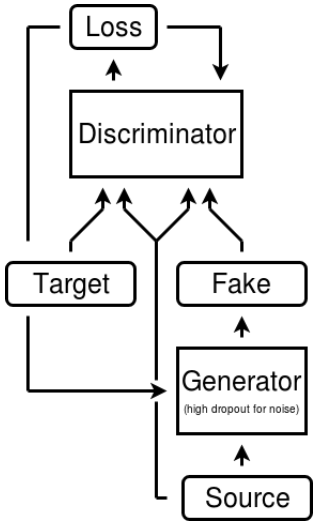

Figure 4: CGAN architecture

4.2 BASIC CONVOLUTIONAL DECODER

Our first model was a standard conditional GAN (CGAN). Starting from a classic CGAN code, we implemented the no-latent-sample generator described above as an encoder (several Convolution-Dropout modules, adding channels and reducing the image dimensions), followed by a decoder (Deconvolution-dropout modules restoring an image of correct size). The discriminator is a similar encoder ending with two linear layers. The activation used is LeakyReLU. We follow Isola et al. (2017) by using instance normalization, suited for style-transfer tasks, instead of traditional batch normalization.

Figure 5 shows the output of our CGAN, trained on one tenth of the dataset (as they follow each other in a video, close pictures temporally tend to be very similar, so trimming the dataset isn't a big loss of information). The first row shows the first three generated results of our network. The second row shows the 200th-202nd (67th epoch) generated images. The third row shows the 749th-751st (250th epoch) generated images. As we can see, on the second row, the GAN learns to generate the lights and road although very blurry. On the third row, our GAN learns to only create light at certain locations : street lamp and road with water reflecting light.

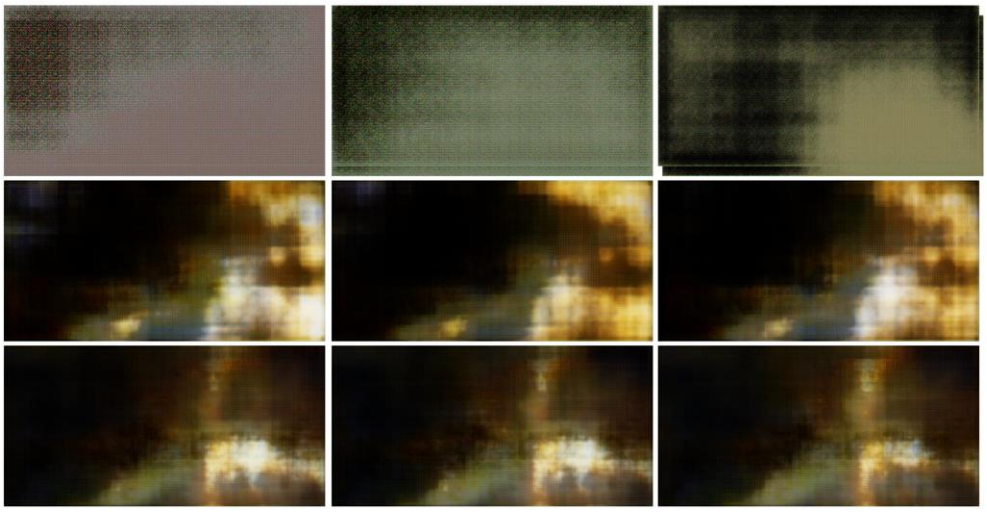

Figure 5: CGAN image generation (without skip connections)

However, the images remain very blurry. The main issue with the classical architecture is that the input layer is downsampled until a bottleneck layer, and can only learn high-level features, as the gradient typically doesn't flow to the the very first layers that encode the lower-level ones.

5 GENERATOR IMPROVEMENTS

5.1 U-NET DECODER

In our second model, to give the generator a chance to get low level features as well, we add skip connections between layers i and $n - i$ where n is the total number of layers. This model follows Isola et al. (2017) more closely, and is based on the U-Net from Ronneberger et al. (2015). Figure 6 shows the GAN generator in the two architectures.

As expected, the U-net generator produces images that respect the low-level features of the input picture ; as you can see from figure 7, smaller-scale elements of the image tend to transfer from the input to the output. Those sharper details, especially the edges, are a considerable improvement for human perception as they allow shapes like the streetlight, the shape of the headlights, or the car ahead to be recognized.

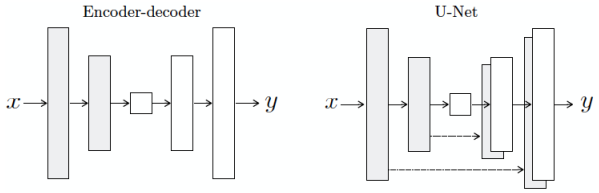

Figure 6: The two GAN generator architectures Isola et al. (2017)

Figure 7: Images generated at epoch 1, 10, 20 (with skip connections)

5.2 U-NET DECODER WITH L1 LOSS PENALTY

The details and the overall luminosity choices of this last model tend to be correct ; however, the colors are not the ones human perception would expect, and the edges come through with a lot of contrast, in a way that evokes a drawing rather than a photographic representation. In order to solve those issues, we tried adding a global regularization over the image to the training. The penalty we used was an L1 loss between the hallucinated output and the image it has to replicate.

Figure 8: Images generated at epoch 1, 10, 20 (with skip connections L1 ratio=100)

Figure 9: Images generated at epoch 1, 10, 20 (with skip connections L1 ratio=1)

Figure 10: Images generated at epoch 1, 10, 20 (with skip connections L1 ratio=0.01)

We have tried varying ratios between the L1 penalty and the loss induced by the discriminator (figure 8, 9, 10) ; a ratio of 0.01 means the L1 penalty was 100 times less important than the discriminator loss. From the experiments, the L1 penalty is indeed effective at forcing coherence in a global colour scheme, but completely overwhelms the finer details, resulting in an unsatisfactory, very blurry image even at low ratios. This might be because it is easier for the optimizer to cater to the fixed penalty rather than handle both the generator and discriminator at the same time.

6 DISCRIMINATOR IMPROVEMENTS

6.1 OTHER DISCRIMINATOR ARCHITECTURES

In our third model, we design a Markovian discriminator(PatchGAN) described in Isola et al. (2017), which assumes independence between pixels outside the patch diameter. As we can see in previous experiments, the L1 term are sufficient to ensure the correctness at the low frequencies feature(outline of the picture). However, for high frequency part (like edges), we need to focus our attention on small patches. Therefore, we use a patchGAN discriminator that tries to classify if each smaller patch in the images is real or fake seperately, and then average the loss function. In fact, a patchGAN is just a fully convolutional GAN which processes each image patch independently and identically. The patch size is typically much smaller than the full size of images : for example, ours is designed to be 32 times smaller. If the patch size is 1, then it could be named pixelGAN. Figure 11 is the structure of pixelGAN, patchGAN and original full size imageGAN respectively.

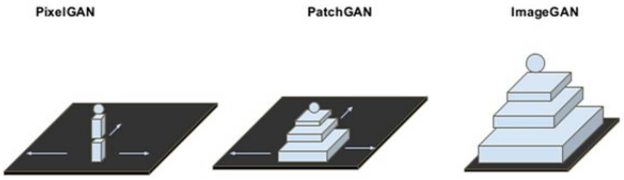

Figure 11: PatchGAN structure (Isola et al. (2017))

6.2 MULTIPLE-DISCRIMINATOR GAN

Faced with the variety of architectures we find in the literature, the one to choose is one given problem is not an obvious choice. These discriminators have various strengths and weaknesses: some of them are better for locally features(with artificial colors), while some of them are better for globally features (but blurry). We then try to train our generator against multiple discriminators at the same time, with the hope that it will learn to respect the image structure at all of their spatial scales at the same time.

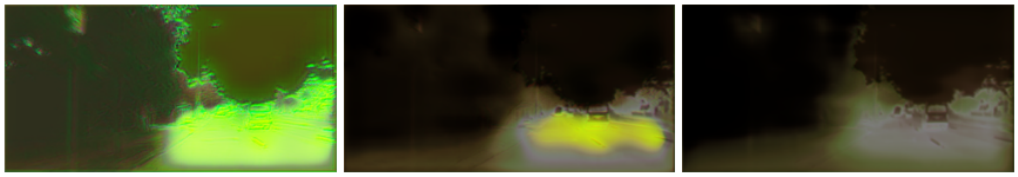

Figure 12: Combining all-CNN and patchGAN, with L1 Ratio = e-4)

Figure 13: Combining all-CNN and patchGAN, with L1 Ratio = e-2)

Figure 14: Combining pixelGAN and patchGAN, with L1 Ratio = 1)

Figure 12 shows the result of combining the all-CNN discriminator with a patchGAN. The L1 ratio is e-4.

Figure 13 shows the result of combining the all-CNN discriminator with a patchGAN. The L1 ratio is e-2.

Figure 14 shows the result of combining the pixelGAN discriminator with a patchGAN. The L1 ratio is 1.

We compute a global loss achieved by averaging several discriminator losses, as well as the L1 penalty. Learning the weights for this averaging would be desirable in future work ; for now, in order to prevent the model from oscillating between following discriminators one after each other, we give a higher weight, in order, to the all-CNN, patchGAN and pixelGAN discriminators. Some parameter fine-tuning is required for this model, but the final all-CNN and patchGAN we choose has the same sense of structure as the U-net and all-CNN GAN with less unnatural colours artifacts.

7 OUT-OF-DOMAIN GENERATION

In order to check how robust our model is and to highlight what the model is learning, we applied it on an out-of-domain picture of the CMU campus. The result is reported in figure 15. The model tries to identify typical elements of the training images : it recognizes trees well, but tries to light the background up without a good reason here, as the dataset images typically feature streelights in the distance, with heavy reflections on the rain of the windshield. An interesting point is how all of the finer details get painted over with very bright colours : indeed, in the training data, those tend to be objects that light up at night, like streetlights, car lights, or bus stops. The result could not fool human perception, but still learns structures at different scales.

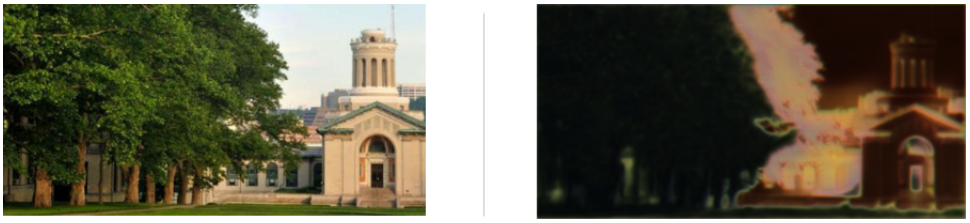

Figure 15: The Carnegie Mellon University Campus

8 CONCLUSION

We applied conditional GANs to the day-night image-to-image translation problem, on a more challenging dataset than is usual for the task. We have explored different architectures and performed error analysis, showing the influence and limits of tricks such as skip connections in CNNs and L1 regularization in hard generation tasks. We also explored an original way to combine models in generation tasks, by training a generator with several discriminators simultaneously, and showed that this method can yield improvements over simpler models. Although the quality of the generated images at this time is still far from movie editing standards, we hope that this work and analysis can contribute to future progress in image translation tasks.

REFERENCES

Alexei A Efros and William T Freeman. Image quilting for texture synthesis and transfer. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pp. 341–346. ACM, 2001.

Leon A Gatys, Alexander S Ecker, and Matthias Bethge. A neural algorithm of artistic style. *arXiv preprint arXiv:1508.06576*, 2015.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pp. 2672–2680, 2014.

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. *arXiv preprint*, 2017.

Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision*, pp. 694–711. Springer, 2016.

Ming-Yu Liu, Thomas Breuel, and Jan Kautz. Unsupervised image-to-image translation networks. In *Advances in Neural Information Processing Systems*, pp. 700–708, 2017.

Michael J Milford and Gordon F Wyeth. Seqslam: Visual route-based navigation for sunny summer days and stormy winter nights. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pp. 1643–1649. IEEE, 2012.

Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

László Neumann and Attila Neumann. Color style transfer techniques using hue, lightness and saturation histogram matching. In *Computational Aesthetics*, pp. 111–122. Citeseer, 2005.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241. Springer, 2015.

Yichang Shih, Sylvain Paris, Frédo Durand, and William T Freeman. Data-driven hallucination of different times of day from a single outdoor photo. *ACM Transactions on Graphics (TOG)*, 32 (6):200, 2013.

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. *arXiv preprint arXiv:1703.10593*, 2017.