An Unsupervised Deep Learning Method for IR to Visible Conversion

Alon Shavit

School of Electrical Engineering, Faculty of Engineering, Tel-Aviv University, Ramat Aviv 69978, Israel.
alonshavit@mail.tau.ac.il

Infra-red (IR) cameras can produce images in total darkness or low light conditions when regular cameras are severely impaired. Hence, they have great advantages on many applications in the surveillance and automotive industries. Yet, IR cameras produce greyscale images which represent the relative temperatures of the scene. As a result, thermal images are harder to understand for an untrained human or for an algorithm which is designed for day images. In this work I suggest an unsupervised deep learning method which transforms IR images into realistic visible output. The method is based on improvements which are applied over an existing framework, to make it suitable for handling IR to visible conversion. I show that the proposed solution produces better results than the existing framework. The results are also compared against a state-of-the-art supervised deep learning technique. I show that during daytime the proposed method produces comparable results to the supervised alternative, while having an advantage in complete darkness or low light conditions. All the results are analyzed both qualitatively and quantitatively on a publicly available dataset.

***Index Terms*—IR cameras, Spectral Conversion, GANs, Deep Learning.**

I. INTRODUCTION

Seeing in the dark is the ultimate dream. For decades, humans have been trying to overcome their physical limitation, which forces us to have a light source in order to use our vision system – which translates photons into shapes, colors and objects. Still, it has a very basic physical limitation- it does not operate in any spectral band different from 400-700 nm, usually called the visible spectrum. This capability is very important for a lot of real-life problems. Many practical challenges in the security and the automotive industries could be solved by seeing beyond our native spectral range.

Indeed, a lot of effort has been made throughout the years to solve this problem. While different types of imaging systems have been developed, they all share one common physical principle – converting a given spectral band to the visible spectrum, allowing us to reveal the unseen by converting it to a picture which we can see. However, all those techniques share some notable disadvantages; The conversion process creates an image in the visible band, which is difficult to understand to non-trained individuals, due to the non-native spectral input source. In addition, the produced images are missing the chromatic information, which is an important part of the human vision system.

Thermal imaging (TI) systems are good example of spectral domain translation. Thermal radiation emission is proportional to the temperature of the object. Hence, hot object produces more thermal radiation than a colder one. The wavelengths of IR are longer than those of visible light, so IR is invisible to humans [3]. A TI system converts the thermal radiation intensity into a greyscale image which represents the temperatures differences of the scene. Hence, a thermal image is essentially a relative heatmap, and has very low in common with the luminance of the visible spectrum. Moreover, thermal image has no chromatic information at all. As a result, converting a thermal image into a visible image is a great challenge.

Thermal to visible conversion has a unique importance in the autonomous car industry. Thermal cameras can operate in complete darkness or low light conditions, when regular cameras have poor performance. However, due to the different spectral range many of the road marks, which are painted using color which is designed for the visible spectrum, are barely seen. In addition, many algorithms which were designed to operate during day-time, are not applicable on IR images. Furthermore, converting IR images into visible spectrum could enlarge the training data base for artificial intelligence algorithms, which are designed for day images.

Deep convolutional networks can be used to perform the IR to visible conversion. However, to perform the network training, the majority of existing techniques require large data base consists of aligned visible-IR pairs. This demand has several limitations; Firstly, the image pairs should be photographed at the same time to ensure good pixel to pixel alignment. Secondly, the thermal and visible cameras should have the same field of view and aspect ratio. Moreover, to train the network to produce day images, all the pairs should be taken during daytime. Another big gap is the data availability - While public data sets of visible or IR images are common, data sets of IR-VIS pairs are rare [2].

This paper proposes a method which transforms IR images into visible images. It is based on optimizations to the network suggested on [1] by Huang at el. The proposed method uses an unsupervised translation network, which enables training without using any aligned image pairs. The method is evaluated, both qualitatively and quantitatively, on a publicly available data set consists of IR and VIS images which were taken from a moving car. The results are compared against (I) visible ground truth, (II) supervised translation network which was suggested by Berg at el in [2] and (III) the original implementation of [1] without any improvements.

II. RELATED WORK

The infrared (IR) spectrum is often divided into a reflection dominated band and an emission dominated band. The reflection band consists of near IR (NIR, 0.74–1 μm) and short-

wave IR (SWIR, 1–3 μm) regions, while the emission band consist of mid-wave IR (MWIR, 3–5 μm) and long-wave IR (LWIR, 8–14 μm) regions [4]. Several approaches have been suggested to enable spectral conversion from IR image into visible image.

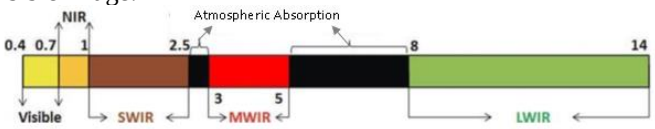

Fig. 1. Electromagnetic spectrum, with IR and visible notations. Wavelengths are given in μm [5]

Some articles, like Hu *et* al [6] showed that IR and visible images share common features, which can be extracted using mathematical models. Riggan *et* al [7] suggested to use a convolutional network in order to perform feature extraction out of a thermal image. In those approaches, the visible image can be estimated using the extracted features which are taken from the IR image. This technique is limited to very specific applications, like face recognition, and does not deal with the general problem of spectrum domain conversion.

In the recent years, convolutional neural networks (CNNs) have become the major method which enables good image prediction, including translation from one spectral domain to another. In all the implementations, the task definition remains the same - "translating" an input image into its corresponding output. This output can be forced to be in a certain domain or style.

The major breakthrough in this field has happened thanks to the use of conditional GANs (Generative Adversarial Networks) [8, 9, 10]. One of the widely used approach was suggested by Isola *et* al [11], which showed some impressive results using the Pix2Pix network architecture. This DCNN implementation takes an input image and turns it into different domain. The idea is to train a conditional generative adversarial network which is able to "fool" domain X classifier by using an image which was created from domain Y image. Later, it was shown by on Zhang *et* al [12] that GANs can be used to transform thermal face images into their matched visible domain images. Suaretz *et* al [13] showed that triplet DCGAN [17] architecture can be used to colorize near infrared images. Berg *et* al [2] showed that LWIR images can be translated into visible output more accurately, by changing the color space and the cost function of the original PIX2PIX network. Paper [2] used the same data base which I used for this work. Hence, in the evaluation section I shall compare my results against the final results of [2].

Although those articles have presented impressive results, they suffer from one basic limitation- they are all based on *supervised* learning. Hence, the data base which is used for training those networks is based on image **pairs**. It means that for each IR image, there must be a corresponding visible image. The two images must be taken at the very same time to ensure good pixel to pixel corresponding. In addition, the visible camera field of view (FOV) should be the same as the thermal camera FOV. Those limitations are a deal breaker for many real-life applications when IR-visible image pairs data set is not available.

To deal with this gap, an unsupervised method should be used. Unsupervised image to image translation is trained by using two independent sets of images. One set consist of images from domain A, while the other consists of images from domain B. this attitude solves the corresponding pairs problem which was mentioned before. The main challenge in this case is learning the joint distribution of images in the two different domains. In [18] a conditional generative adversarial network was proposed to translate rendering images into real image. Liu et al [14] suggested the UNIT architecture, and showed that the joint distribution could be learned using generative adversarial network, based on the assumption that the two domains share the same latent space. This assumption states that images from different domains could be mapped to the same latent representation in a shared latent space. To ensure a good domain translation, this method is based on the cycle consistency constraint assumption [15, 16]. This assumption states that an image in a source domain can be mapped into an image in the target domain and then it can be mapped back to the original image in the source domain.

However, for a lot of real-life applications, those unsupervised methods are not good enough. They all make simplified assumption by modeling the problem into deterministic one-to-one mapping. As a result, they fail to generate diverse outputs from a given source domain image [1]. This capability is important if the translated image is aimed to be use for making decisions. For example, consider a traffic light, which has two possible states: green and red. A deterministic model would map an input image into green **or** red output, hence avoiding the user/algorithm to understand that another possible translation exists. To deal with this problem, Huang *et* al presented a multimodal unsupervised image to image translation framework (MUNIT [1]). This work assumes that an image consists of two latent domains: style and content. Hence, images of different scenes from the same domain share the same style code, while images from different domain of the same scene share a common content code. This work has yielded some impressive results.

The aim of my work is to show how the MUNIT method could be optimized to deal with IR to visible translation. In the original MUNIT article, only images from the visible spectrum were used. It makes the translation process easier, since there is a relation between the object appearance in one domain (i.e. a mountain during summer), to its appearance in other domain (i.e. a mountain during winter). IR and visible domain do not have this privilege. IR image represents totally different physical data than a visible image. Hence, there is no a direct relation between an object taken by IR camera to its visual appearance. On [2], this IR to visible translation task was found

hard to perform by using the network from [14], which was built for images in the visible spectrum. They showed that it created cartooning effects and unsmooth results. The main renewal of my approach is that by (a) adjusting the objective functions and (b) performing suitable preprocessing over the training set, I could archive smoother visible results compared to the network suggested on [1]. In addition, I show that the results are comparable and sometimes better than the supervised method suggested on [2].

III. DATA

The KAIST-MS traffic scene dataset [19] was chosen to be the source of images for this work. It consists of about 95,000 visible and thermal pairs, which were taken from a driving car. Fig. 2 shows the physical setup. Since I wanted to train the network to produce day-like images, all the images which were taken during night time were ignored. It left about 60,500 pairs.

The IR images were originally taken using a FLIR A35 LWIR camera with resolution of 320 x 256 pixels and vertical field of view of 39º. The visible images were taken using PointGrey Flea3 color camera with 640 x 480 resolution and vertical field of view of 103.6º. The frame rate of both cameras was 20 fps [19]. After alignment process, each image is 640 x 512 resolution, with bit depth of 24 bit (8 bits per each color channel).

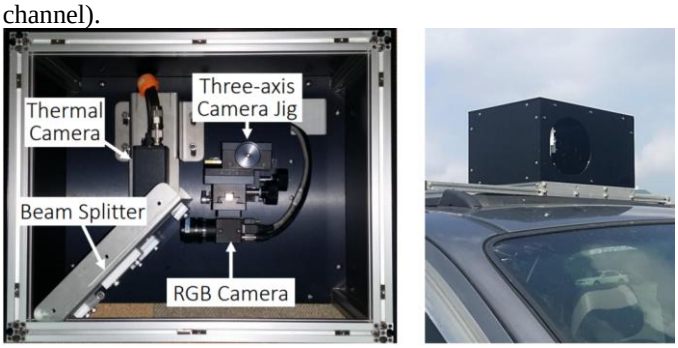

Fig. 2. The HW configuration for capturing VIS-IR images [19]

As a preprocessing stage, each picture was trimmed around the middle and was then down-sampled to have 128 x 128 pixels resolution. 4033 images from each domain were randomly chosen for the training process. Note that even though the data set consists of aligned pairs, for the training process each image was handled separately, without any connection the any image on the other domain, to ensure unsupervised learning process. In addition, thanks to the random choice of images, the training set did not include the original aligned pairs. The validation set consists of 30,247 VIS-IR pairs. It included all the data base, excluding night time images, when each second image is taken (due to the fact that the images were taken in 20 fps, there were duplicate images). The complete validation set was used for evaluation.

IV. METHODS

The baseline for this project was the MUNIT method which was proposed in [1]. I used the Keras implementation of the network from [20] as a baseline for my modulations. I used the instance normalization suggested on [22], and the implementation from [21] to normalize the activations of the precious layer at each training step. The MUNIT network is based on two auto encoders, one for each domain. To encode, an image is represented by 2 codes – content code *c*, and a style code *s*. The model is trained using GANs which ensure the translated images are similar to real images from the target domain. In addition, bidirectional reconstruction objectives reconstruct both images and latent codes [1].

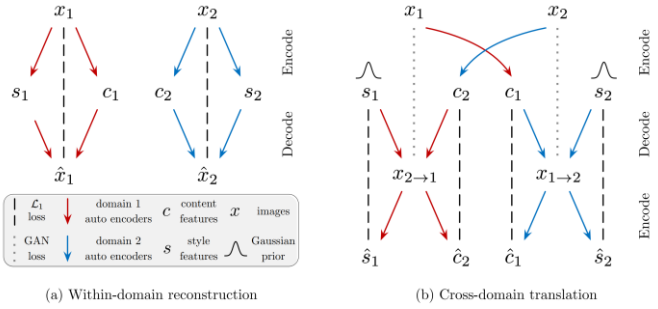

Fig. 3. MUNIT model overview [1]

Each domain X_i consists of a style encoder E_i^s and a content encoder E_i^s, which encode the latent codes c_i and s_i for a given image x_i. The decoder D decodes latent codes into an image. Let the visible domain be *i*=1 and the IR domain be *i*=2, then the IR to visible conversion is performed by $x_{2\to1} = G(c_2, s_1)$. The loss function is based on (a) bidirectional reconstruction loss to ensure the encoder and the decoder are inverses and (b) adversarial loss to ensure the distribution function of the translated images is undistinguishable compared to target domain. Fig. 3 demonstrates this logic.

The loss functions play a key rule during the training process of the network. In the original paper, there are three losses which are used for the training process: (a) image reconstruction, (b) latent reconstruction and (c) adversarial loss. Both image and latent code reconstruction are calculated based on L_1 matric. Image reconstruction is defined as the total loss caused by the difference between an image to its reconstruction after encoding and decoding. Latent reconstruction is defined as the total loss caused be by the difference between an image content and style code to the reconstrued code after decoding and encoding. The adversarial loss is defined as the GAN loss, caused by the difference between images generated by the model to the real images in the target domain.

Using the original MUNIT implementation to perform IR to visible conversion had several problems. The output images were blurry, and many of them (about 10%) had strange colors in random places across the image.

To deal with those imperfections, I performed some changes to the original implementation. It is known that a simple L_1 loss produce blurry results on image generation tasks [23]. In addition, as it was explained before, unlike pairs of visible images, IR-visible pairs do not share the same meaning for luminance. Moreover, the human visual system has higher acuity for luminance differences than for color differences [24].

According to those assumption, I defined the improved loss function (1) below:

(1) $$loss = \lambda_1(L_{GAN}^{x_1} + L_{GAN}^{x_2}) + \lambda_2 L_{recon\_luma}^{x_1} + \lambda_3 L_{recon\_chroma}^{x_1} + \lambda_4 L_{recon\_luma}^{x_2} + \lambda_5 L_{recon\_chroma}^{x_2} + \lambda_6(L_{recon}^{c_1} + L_{recon}^{c_2}) + \lambda_7(L_{recon}^{s_1} + L_{recon}^{s_2})$$

Where $L_{GAN}^{x_i}$ is the discriminator (GAN) loss, $L_{recon\_luma}^{x_i}$ is the chrominance loss, $L_{recon\_chroma}^{x_i}$ is the luminance loss, $L_{recon}^{c_i}$ is the content loss and $L_{recon}^{s_i}$ is the style loss for an image from domain *i*. As one can see, I propose to separate the loss into its luminance and chromatic channels. The loss for the luminance channel is calculated using L_1 loss:

(2) $$L_{recon_{luma}}(x, x') = |Y(x) - Y(x')|$$

The x and x' in (2) are RGB images.

The loss for the chromatic channel is calculated by:

(3) $$L_{recon_{luma}}(x, x') = 0.5|Cb(x) - Cb(x')| + 0.5|Cr(x) - Cr(x')|$$

Y, Cb and Cr in (3) are the image representation on the YUV color space, and are calculated based of the RGB values according to formula (4):

(4) Y=0.299R+0.58G+0.114B
Cb=-0.147R-0.289G+0.436B
Cr=0.615R-0.515G-0.1B

Note that in the proposed loss function (1), each spectral domain has its own chrominance and luminance loss. It enables better control over the reconstruction process, by adjusting the weights according to physical considerations. During the evaluation process, I found that (given *i*=1 is the day domain and *i*=2 is the IR domain) the values which works well are:
$\lambda_1 = 2, \lambda_2 = 18, \lambda_3 = 6, \lambda_4 = 18, \lambda_5 = 4, \lambda_6 = 1, \lambda_7 = 1$

This attitude gives much higher importance for the luminance, while giving priority to visible channel for chromatic reconstruction. This asymmetry forces the network to "try harder" to maintain the luminance information of the reconstrued image, which eventually leads to better reconstruction results of IR to visible transformation.

The objective is the minimize the loss which is proposed in (1). Using this new objective function led to sharper images on the visible domain, and almost vanished the strange colors effect.

Another improvement which was found to improve the results is applying some random image processing over each batch during training. Those operations include zoom (for both domains), saturation change (for visible domain), contrast change (for visible domain) and brightness change (for IR domain). Those operations reduce overfit and improve the results of the translation process for less common spatial elements like big cars or small pedestrians.

As part of the training process, the use of fewer residual blocks was tested, in order to increase training time. It led to much worse output quality and hence it is not recommended.

The training process was done using Tesla K80 GPU, and took about a week. The initial learning rate was 0.0001, and it was decrease by half every 20,000 iterations.

V. EXPERIMENTS

A. Evaluation Methods

In order to evaluate the performance of the suggested method over the test images, the output images (synthetic visible images which were created from IR input images) were compared against the performance of two references; The first one is the output from the original implementation, using the parameters which were proposed on the original paper [1]. The second one is the output from the (supervised) network which was suggested on [2]. For this purpose, I used the model and the weights which were referenced on the article [25].

All three methods (my, original implementation and [25] implementation) were compared against the ground truth visible camera, which is provided on the data set from [19]. To ensure a fair comparison, all the output images were cropped over the center and rescaled to have the same resolution.

Four different methods were used for comparison, each of the them were normalized to have a value of 0 for no-similarity against the ground truth, and a value of 1 if the reference image is identical to the visible image ground truth. The common L1 metric was normalized to 1 by dividing the result by the total number of pixels. In addition, RGB (3 channels) structural similarity (SSIM) which was suggested by Wang *et* al on [26] was used. I used the implementation provided by [27] for this purpose. In addition, I used SSIM over the luminance component on the LAB color space, as it was suggested on [2]. Furthermore, during my work I found out that when comparing synthetic visible image against the ground truth it can be useful to use a correlation matric, which ensures that there is a good correlation between the two images, by giving a "penalty" for unmatched areas. For this purpose, I performed a normalized 3D convolution between the images, using the function from [28].

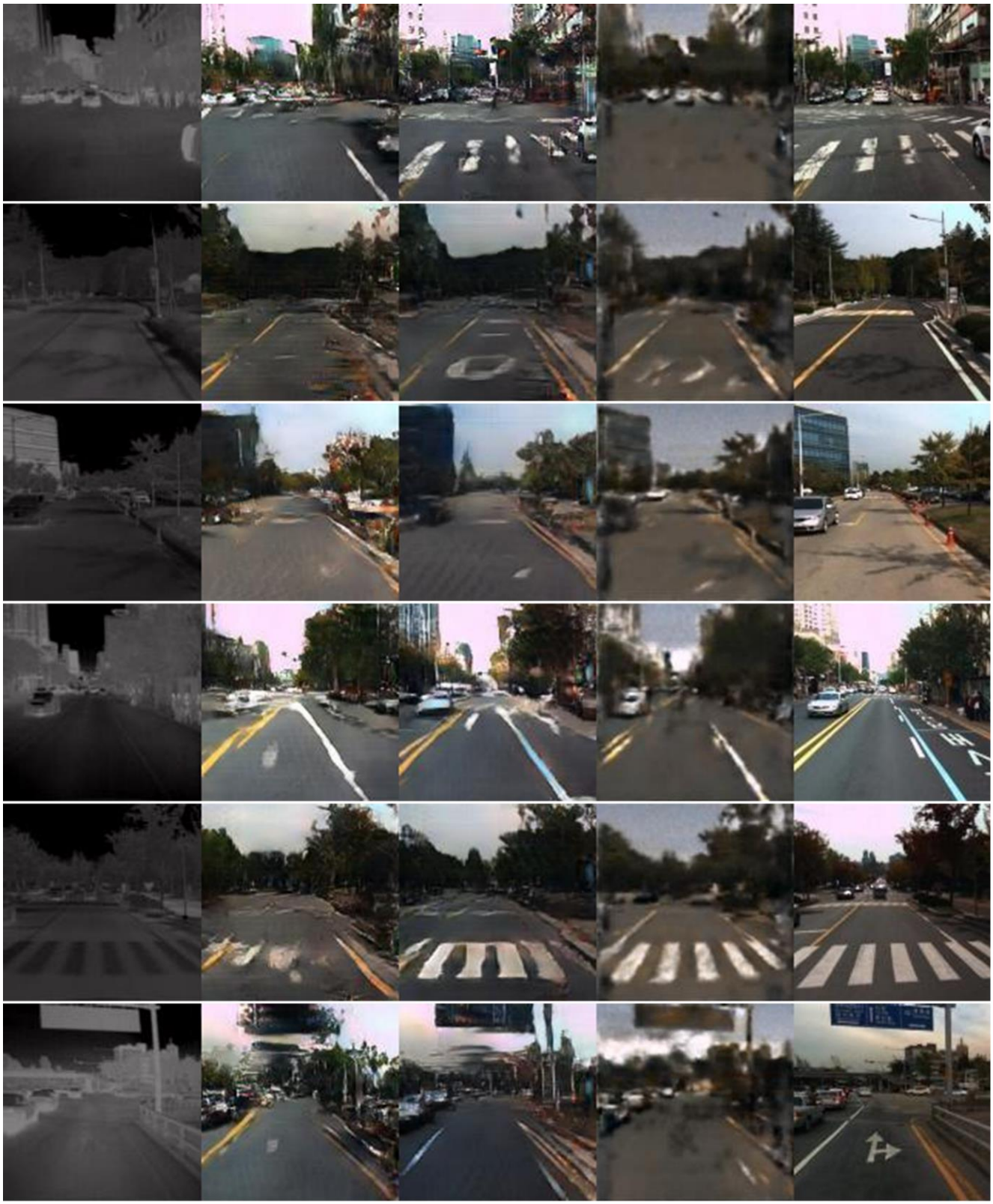

Fig. 4. Transformation examples for three different methods

B. *Quantitative Results*

All the IR test images (total of 30,247) were converted into visible images using three different methods (original MUNIT [1], improved MUNIT for visible to IR conversion (my), and the supervised method from [2]). The output images from each method were then evaluated using the metrics which were discussed before. The averaged results are given in Table 1.

	Supervised method from [2]	**MUNIT from [1]**	**Improved MUNIT (my)**
L1	0.863	0.851	0.864
SSIM (Luminance)	0.441	0.326	0.363
SSIM (RGB)	0.603	0.468	0.513
Templet Correlation	0.729	0.742	0.762

Table 1: Mean image distance test results for 30,247 image pairs

Table 2 shows the results relative to the supervised method.

	Supervised method from [2]	**MUNIT from [1]**	**Improved MUNIT (my)**
L1	100%	98%	100% (+2%)
SSIM (Luminance)	100%	74%	82% (+11%)
SSIM (RGB)	100%	77%	85% (+10%)
Templet Correlation	100%	102%	105% (+3%)

Table 2: Mean image distance test results for 30,247 image pairs, relative to the supervised method. The numbers on parenthesis are the improvement relative to the original MUINT.

Table 2 shows that my method provides an improvement of 11% and 10% in the luminance and RGB SSIM, accordantly. In addition, the correlation improved by 3%. As expected, the supervised method still provides better quantitative results relative to the unsupervised method. Still, my method provides a notable improvement to the original MUNIT, making it comparable to the supervised method. Note that in my method,

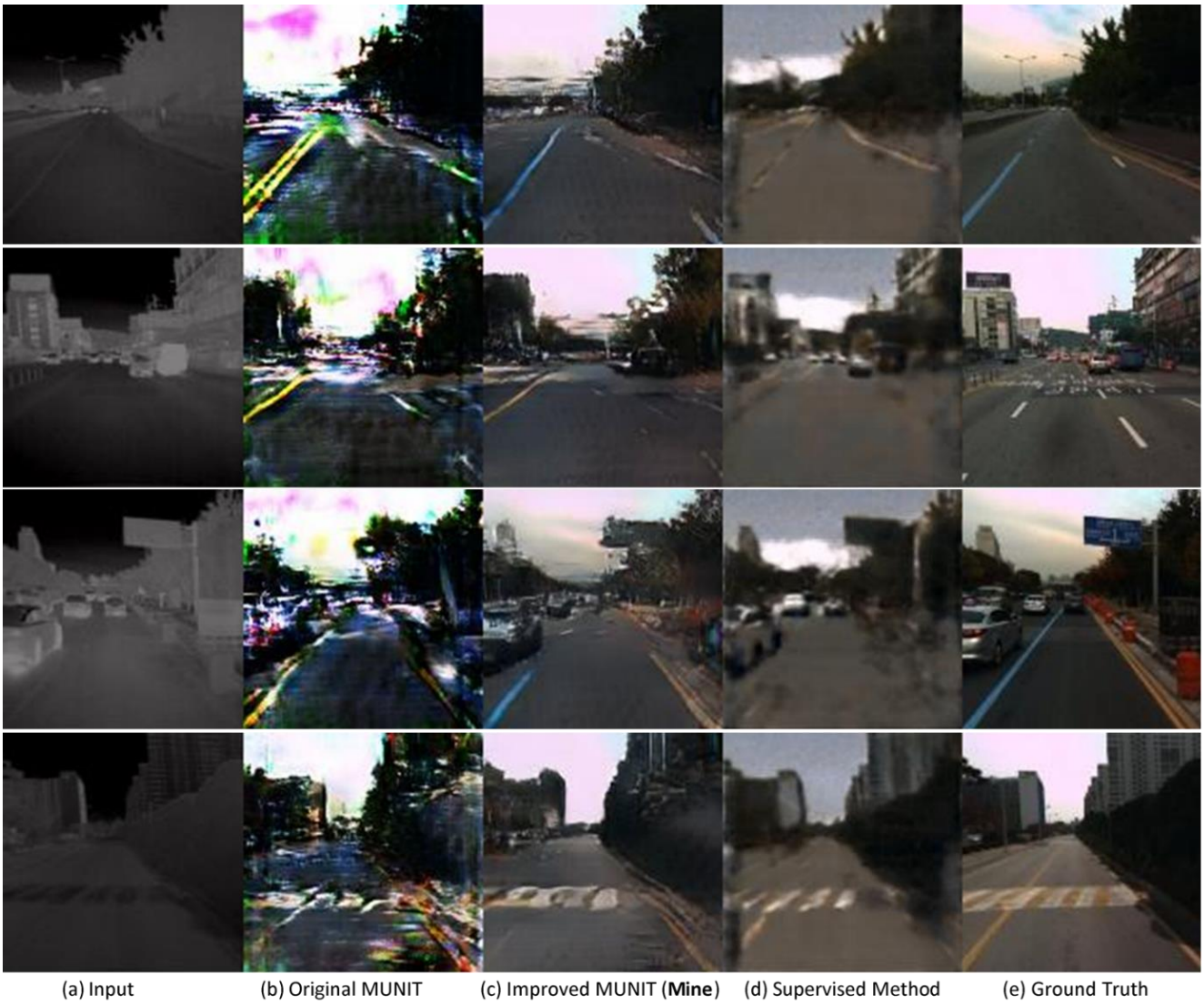

Fig. 5. Failure cases for original MUINT

the training process does not require any aligned pairs – which makes it suitable for scenarios when such data is not available.

C. Qualitative Evaluation

In Fig. 4, six transformation examples for the supervised method from [2], MUNIT from [1] and the suggested improved MUNIT are provided. Based on those examples, it can be seen that for many scenarios, my method provides comparable or better result to the one suggested by Berg *et* al in [2] in terms of subjective assessment. Particularly note the advantage of my method in reconstructing painted road marks. This benefit is especially important for the autonomous car industry, which relies on those marks in order to make decisions. It can be seen that my method supplies sharper images which looks more realistic for many examples. Some artifacts are also seen on those images; Note, for example the second bottom row. In this example, there are two trees in the horizon. While the supervised method outputs 2 blurry trees, my method outputs just one, sharper tree. It can be explained due to the different architectures of the networks – while supervised method encourages "pixel to pixel" transformation, the suggested unsupervised method prefers realism over pixel to pixel accuracy.

Fig. 5 provide some failure examples of the original MUNIT, which were improved using the suggested method. It can be seen that the "weird colors" effect which is commonly seen when using the original MUNIT disappears when using my suggested method. In addition, the output images are much sharper. This result proves my previous assumption, which suggested a separation into luminance and chrominance components on the target function.

Fig. 6 shows some cases when the supervised method provides better results compared to the suggested method. There are some common reasons for those failures. The first one (first upper row for example) is relatively rare spatial elements like big cars or uncommon capturing angle of the scene. It causes failure since an unsupervised method has less information to create the right probability distribution function for those cases. The second reason (bottom row for example) is

Fig. 6. Failure cases for proposed method

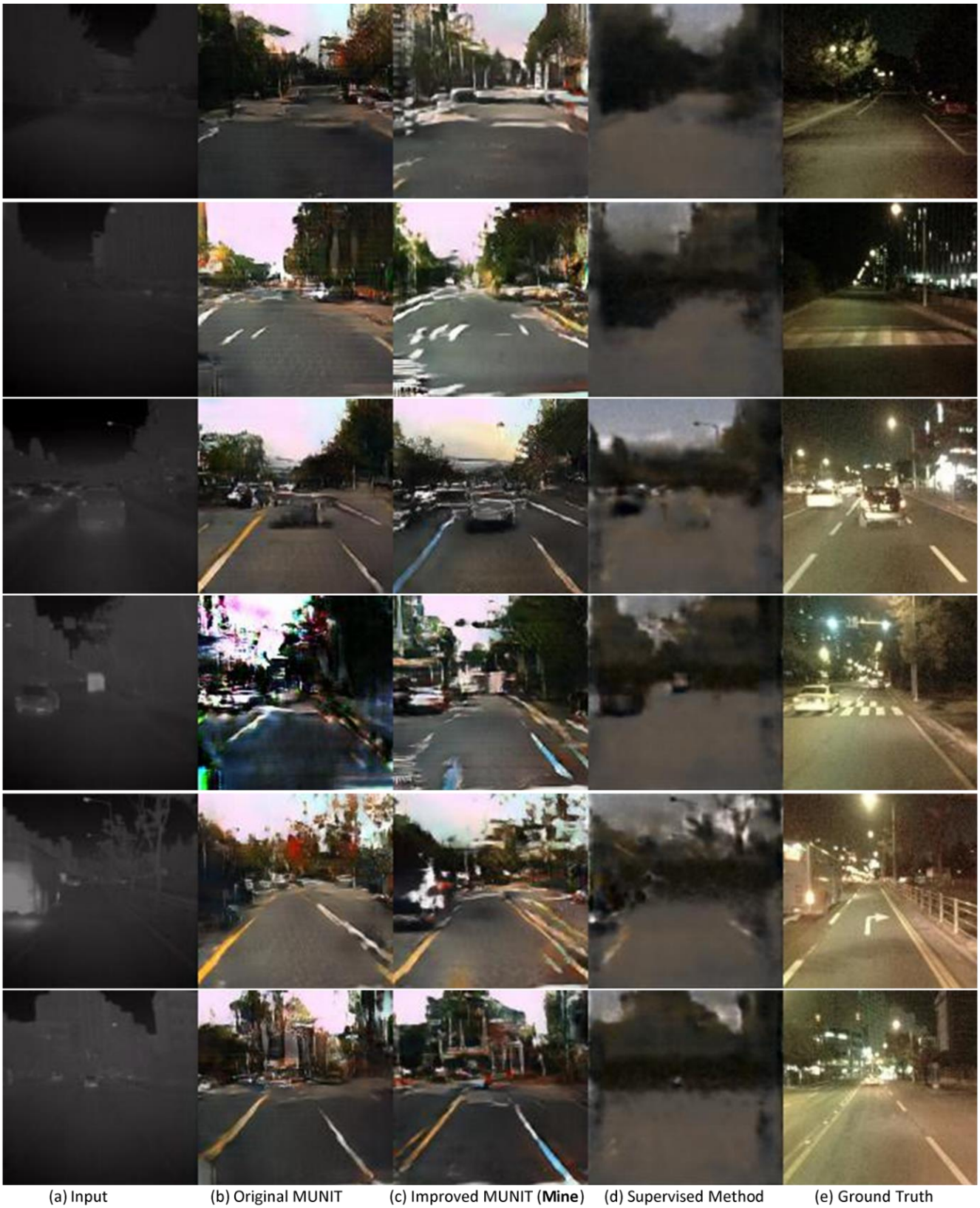

Fig. 7. Night time scenario: Transformation examples for three different methods

wrong interpretation of the scene. It happens when the network translates the input into something which is not there (like a tree which is been translated into a building). Supervised method suffers less from this effect since it is guided to perform the exact right output during the training process. Note that in those failure examples, it is still can see that the suggest method provides better results compared to the original MUNIT.

D. Night Scenario

As it was mentioned before, the training set consist of images which were taken only during day time. It was done since I wanted the network to learn how to produce "day looking" output images. Still, since visible to IR translation could be used for IR images which are taken at nighttime, it is interesting to evaluate this scenario. Since the ground truth comparison for this case is unavailable only qualitative evaluation could be done. Fig. 7 provides some examples of transformation for the supervised method from [2], MUNIT from [1] and the suggested improved MUNIT.

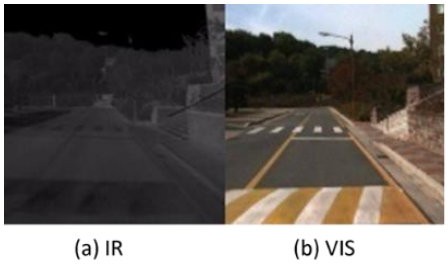

Fig. 9. Different types of crosswalks

At night, the contrast of the IR images is worse, since the temperature differences are lower. The low contrast makes it more difficult for the network to perform the conversion, since objects are harder to recognize. As it can be seen from the examples, the suggested method creates much better results compared to the supervised method. The images are much sharper, and the road marks are much better seen. It can be explained by the fact that unsupervised method is more immune to contrast changes, since it learns to recognize the "big picture". Supervised method, on the other hand, is trained to perform a transformation which converts an input to a specific output. Hence, it is more sensitive to contrast variations. This weakness of the supervised method was also discussed on [2].

E. The Multi Output Advantage

The proposed method is multimodal. It means that it can produce various outputs just by using different style codes from the latent code space. When evaluating the results, I used a random image from the visible domain to produce a specific style code which has been used during the test and validation process. In order to make sure that this random choice does not dramatically affect the output images, I selected 10 others random "style images" and verified that the output images have not had notable changes.

However, there is another option. We could create random style codes just by randomly sampling the latent style space. To do it, I created several random style codes by using a normal sampling of the latent space (with mean=0, and a random standard deviation). Some examples are given on Fig. 8. It can be seen that different style codes produce various outputs which could change the meaning of the image. On this example, one can see that the input image is a crosswalk. However, on the images which the network has been trained with, there are two different types of crosswalk- a "black and white" type - "type I" and a "white and yellow" type - "type II" (Fig. 9). Note that different type could mean different decisions when integrating an algorithm into an artificial intelligence architecture.

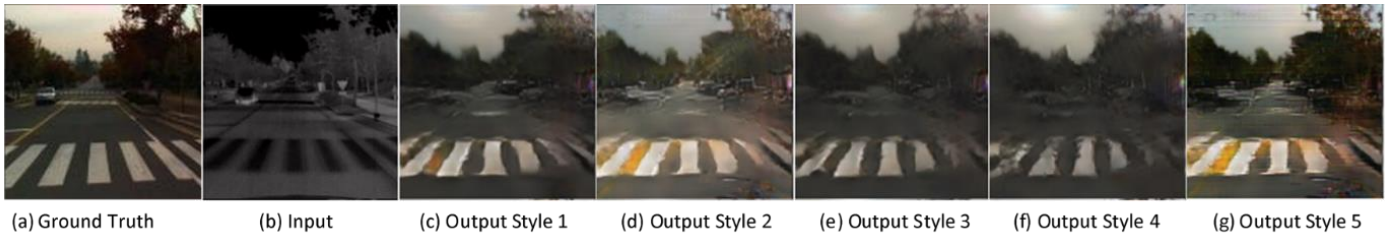

Fig. 8. Various output images created by random style codes

In our case, random style codes produce type I **or** type II, enabling the decision-making algorithm to understand that there are in fact two possible types for this crosswalk. Still, it can be seen on those examples that the network does provide a higher probability for type I, which is the correct translation according to the ground truth image.

VI. CONCLUSION

I have addressed the problem of transforming IR images into daytime visible images. Despite its high relevance, according to my knowledge this problem has not been addressed using an unsupervised learning method. In this work, I proposed some improvements for the MUNIT architecture. I have showed that unlike greyscale to RGB colorization, which only estimates the chrominance, IR to visible transform should separate the luminance from the chrominance during the training process.

The solution has been evaluated both qualitatively and quantitatively on a publicly available dataset. The proposed method dramatically improves visible to IR translation results compared to the original implementation, and supplies results

which are comparable and sometimes better than a state-of-the-art supervised learning technique. I have also shown that when testing night scenarios, the proposed method usually supplies better results compared to the supervised alternative.

Further work includes several possible extensions. First, the proposed method should be tested on other data sets to evaluate performance on different scenarios. In addition, the multimodal capability can be used to evaluate other spectral transformations such as SWIR to visible and MWIR to visible. In this case, it is interesting to analyze if the suggested trained model could be used in order to speed up the training time on other spectral bands. Finally, the combination of both supervised and unsupervised methods into one solution should be examined. This solution should be able to combine paired and unpaired images during the training process, and should be trained differently on each case. This approach might achieve the benefits of both methods while using just one architecture.

VII. APPENDIX

All the code for this work can be found in the Github link: https://github.com/AlonShavit10/IR2DAY

REFERENCES

[1] Huang, Xun, et al. "Multimodal Unsupervised Image-to-Image Translation." Computer Vision – ECCV 2018 Lecture Notes in Computer Science, 2018, pp. 179–196., doi:10.1007/978-3-030-01219-9_11.

[2] Berg, Amanda, et al. "Generating Visible Spectrum Images from Thermal Infrared." 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2018, doi:10.1109/cvprw.2018.00159.

[3] Chris Solomon and Toby Breckon; Fundamentals of Digital Image Processing: A Practical Approach with Examples in Matlab; John Wiley & Sons, Ltd. 2011

[4] Hu, Shuowen, et al. "Thermal-to-Visible Face Recognition Using Partial Least Squares." Journal of the Optical Society of America A, vol. 32, no. 3, 2015, p. 431., doi:10.1364/josaa.32.000431.

[5] Bourlai, Thirimachos, and Bojan Cukic. "Multi-spectral face recognition: identification of people in difficult environments." Intelligence and Security Informatics (ISI), 2012 IEEE International Conference on. IEEE, 2012

[6] Thermal-to-visible face recognition using partial least squares

[7] Hu, Shuowen, et al. "Thermal-to-Visible Face Recognition Using Partial Least Squares." Journal of the Optical Society of America A, vol. 32, no. 3, 2015, p. 431., doi:10.1364/josaa.32.000431.

[8] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In NIPS, 2014.

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

[10] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen. Improved techniques for training gans. In NIPS, 2016

[11] Isola, Phillip, et al. "Image-to-Image Translation with Conditional Adversarial Networks." 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, doi:10.1109/cvpr.2017.632.

[12] Zhang, Teng, et al. "TV-GAN: Generative Adversarial Network Based Thermal to Visible Face Recognition." 2018 International Conference on Biometrics (ICB), 2018, doi:10.1109/icb2018.2018.00035.

[13] Suarez, Patricia L., et al. "Infrared Image Colorization Based on a Triplet DCGAN Architecture." 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2017, doi:10.1109/cvprw.2017.32.

[14] M.-Y. Liu, T. Breuel, and J. Kautz. Unsupervised image-to-image translation networks. arXiv preprint arXiv:1703.00848, 2017

[15] 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.

[16] 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

[17] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. International Conference on Learning Representations, 2016

[18] A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, and R. Webb. Learning from simulated and unsupervised images through adversarial training. Conference on Computer Vision and Pattern Recognition, 2017.

[19] S. Hwang, J. Park, N. Kim, Y. Choi, and I. S. Kweon. Multispectral Pedestrian Detection: Benchmark Dataset and Baseline. In 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 1037–1045. IEEE, jun 2015. 1, 5

[20] https://github.com/shaoanlu/MUNIT-keras

[21] https://github.com/shaoanlu/Conditional-Analogy-GAN-keras/blob/master/instance_normalization.py

[22] I Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. arXiv preprint arXiv:1607.06450, 2016.

[23] A. B. L. Larsen, S. K. Sønderby, H. Larochelle, and O. Winther. Autoencoding Beyond Pixels using a Learned Similarity Metric. CoRR abs/1512.09300, dec 2015

[24] M. Livingstone. Vision and art: the biology of seeing. Harry N. Abrams, New York, 2002.

[25] https://gitlab.ida.liu.se/amabe60/PBVS2018

[26] Wang, Z., Bovik, A. C., Sheikh, H. R., & Simoncelli, E. P. (2004). Image quality assessment: From error visibility to structural similarity. IEEE Transactions on Image Processing, 13, 600-612.

[27] http://scikit image.org/docs/dev/api/skimage.measure.html#skimage.measure.compare_ssim

[28] https://docs.opencv.org/3.0-beta/doc/py_tutorials/py_imgproc/py_template_matching/py_template_matching.html