Improving multi-object re-identification at night with GAN data augmentation

Midas Amersfoort[1,3], Michael Dubbeldam[2], and Arnoud Visser[3]

[1] Vrije Universiteit Amsterdam, The Netherlands
[2] Technolution B.V., The Netherlands
[3] Universiteit van Amsterdam, The Netherlands

Abstract. This study concentrates on a camera-based traffic sensor that measures bicycle, vehicle and pedestrian trips called FlowCube™. To achieve multi-object tracking, FlowCube uses a model chain consisting of object detection, local tracking, trip filtering and re-identification (re-id). Whereas FlowCube's performance is fit-for-purpose during the daytime, it degrades in more challenging nighttime conditions. With that, this study is aimed at improving FlowCube's nighttime re-id performance. The hypothesis is that the poor nighttime re-id performance is due to a lack of nighttime re-id training data. So, in this paper a Generative Adverserial Network based data augmentation with alpha blending is proposed to enrich FlowCube's re-id training data with synthetic nighttime imagery. The findings show that this method improves FlowCube's mean re-id F1 scores and reduces the variance between results across multiple training runs, both for nighttime and general re-id. The same improvement can be expected for other camera-based traffic sensors which use multi-object tracking with re-identification.

1 Introduction

1.1 Context

Today, about 55% of the world's population lives in cities [10]. Such a grand amount of people will result in an abundance of commute which, if not managed correctly, can lead to problematic traffic scenarios. Ideally, pedestrian, cyclist and vehicle traffic have to be managed in an efficient manner that minimizes congestion. To that aim, governments could create incentives to stimulate the use of bicycles over cars for short distances, alter road infrastructure to remove traffic flow bottlenecks, or make use of intelligent traffic light control based on real-time traffic flow analysis. This can be done with the use of data driven traffic management, where (real time) traffic flow data is the driving factor behind infrastructure optimization. However, the problem that arises is a lack of fine-grained measurement data on vehicle, pedestrian and cyclist traffic flows. So, not only point measurements, but the complete overview of traffic throughput, routes and travel time.

This is where the FlowCube™ can provide a solution [9]. FlowCube is a proprietary camera-based traffic sensor developed by Technolution that can measure bicycle, vehicle and pedestrian trips between sensors in a privacy-safe manner[4]. The system does not identify specific persons, nor does it use face recognition. It translates certain object characteristics, such as color and shape, into an arithmetic representation. This representation is sufficient to match traffic patterns, but it is not so specific that it can be used to identify specific individuals. The FlowCube™ combines several detection and matching techniques, but in this paper the focus will be on improving the re-identification (re-id) model. This project was initiated by Technolution and executed internally.

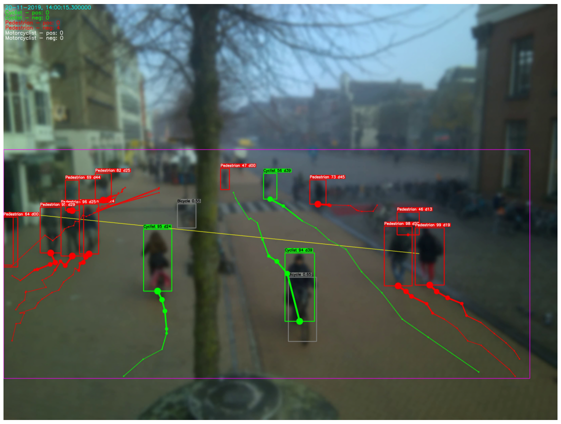


Fig. 1: Example of FlowCube's multi-object tracking (Courtesy [9]).

Currently, multiple objects can be tracked in video streams very accurately [11] as long as no occlusions occur, for instance by predicting the movement directions of the objects. Yet, when occlusion occurs the objects merge and one of the tracks is lost. When the tracks separate again it is essential to assign the original ID again (re-id) to each of the two objects [5]. Otherwise, the algorithm loses efficiency due to ID switches. To measure bicycle, vehicle and pedestrian trips over longer periods one should maintain the correct ID for each of the tracked objects over that period.

In this work, we address FlowCube's current re-id limitations in the nighttime setting and propose an improvement by generating additional training data with Generative Adverserial Network (GAN) data augmentation, introduced in section 2. More specifically, AU-GAN [8] is used to augment FlowCube's re-id training set with synthetic nighttime imagery. The rationale behind using AU-GAN for this purpose is its ability to learn how to realistically transform an image from any source domain (e.g. daytime) to any target domain (e.g. nighttime). When combined with an alpha-blend between the original and AU-GAN transformed imagery this method has the ability to produce generally higher and more stable re-id results.

[4] More details on the FlowCube™: https://www.technolution.com/move/flowcube/

1.2 FlowCube

At Technolution, the FlowCube has been developed using a model chain consisting of object detection, local tracking, trip filtering and re-id[5]. Here, object detection and re-id models are trained on an in-house dataset that is created with the use of pseudo-labeling. This dataset consists of 262403 images of 2693 different entities (Table 1), captured at any time of day. These entities can be pedestrians, cyclists or vehicles. A few examples of such entities during nighttime can be seen in Fig. 2. Local tracking and trip filtering is done with the use of the Kalman filter and Intersection Of Union (IOU); the Kalman filter is used to predict the location of an object's bounding box in the next frame, given its previous boxes (inspired by [4]). Next, the IOU of the predicted box with the actual box is calculated to create a tracklet (multiple image crops containing a certain object of interest) [2]. This process of creating tracklets produces the described pseudo-labeled dataset, where each tracklet has its own unique ID.

The FlowCube is a productions system that despite its limited memory and computing power still functions well during daylight. However, its performance degrades in more challenging lighting conditions. Focusing on the re-id part of FlowCube, this degradation in performance could be caused by multiple issues.

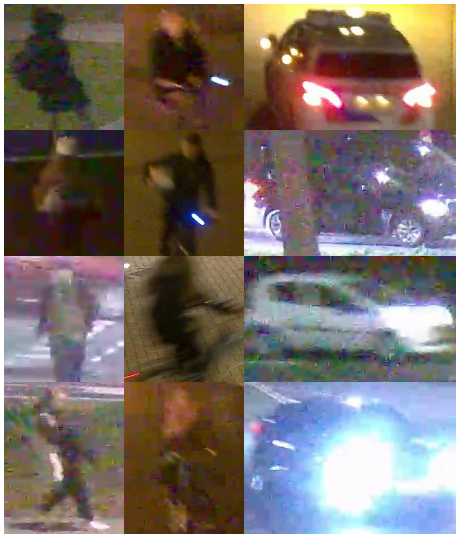


Fig. 2: Example nighttime imagery from FlowCube's re-id training dataset.

Firstly, whereas the dataset used for training FlowCube's re-id model contains hundreds of thousands of images of vehicles, cyclists and pedestrians, it mainly captures them in the daylight setting. This is assumed to be due to suboptimal object detection performance in the pseudo-labeling step for nighttime

[5] This is proprietary code of Technolution and as such not published open source.

imagery. As a result of producing few nighttime tracklets, the nighttime limitations transfer over to the re-id model. Secondly, FlowCube uses a relatively small sized camera module, which limits its ability to capture a lot of light in dimly lit scenarios. With that, contrast and color information is lost, which makes the task of re-id harder. Thirdly, for such a small camera a trade-off between brightness, noise and motion blur has to be made in such dimly lit scenarios. The light sensitivity (ISO) of the camera can be raised, which has the negative side effect of introducing more shot-noise, or lower its shutter speed, which causes motion blur. Figure 2 shows the result of these adjustments. The additional noise and motion blur makes the task of re-id harder in dimly lit scenarios.

With these limitations in mind, there is a call for methodologies that can improve FlowCube's re-id performance, with an emphasis on nighttime conditions. For this study, we explore the use of AU-GAN data augmentation to enlarge FlowCube's re-id training dataset with synthetic nighttime imagery, to see how this affects the night-time and general re-id performance. AU-GAN was chosen over alternative (symmetric) GAN's – such as CycleGAN or ForkGAN – as it utilizes an asymmetric architecture, which is better suited for disentangling the domain-invariant (e.g. object) and domain-specific (e.g. noise and blur) features for adverse domain (e.g. day and night) translation.

2 Generative Adversarial Networks

Generative Adversarial Networks can generate data which is not earlier encountered, by having two competing neural networks: a generator that generates new data instances, and a discriminator (adversarial) that decides whether or not each instance of generated data belongs to the actual training dataset. The goal of the generator is then to generate new synthetic data that can fool the discriminator into thinking that it's real and, alternatively, the goal of the discriminator is to identify images that come from the generator as fake [3].

2.1 Asymmetrical architecture

One such GAN image-to-image translator is the Asymmetric and Uncertainty aware Generative Adversarial Network (AU-GAN [8]). An overview of the model can be seen in Figure 3. Here, x_A and x_B denote two images from adverse domains, i.e. rainy night and daytime respectively. One can also see two generators that are comprised of an encoder and a decoder. That is, $G_{A\to B} = \{G^E_{A\to B}, G^D_{A\to B}\}$ and $G_{B\to A} = \{G^E_{B\to A}, G^D_{B\to A}\}$. The former translates domain A to B, and the latter translates domain B to A. For day → rainy night, the objective is to construct a modified image x'_A from x_B with generator $G_{B\to A}$.

Many of the GAN image-to-image models that strive towards this same objective are based on CycleGAN [13], which was the pioneer in performing this task in an unsupervised manner. This means that they often exploit the property of cycle consistency.

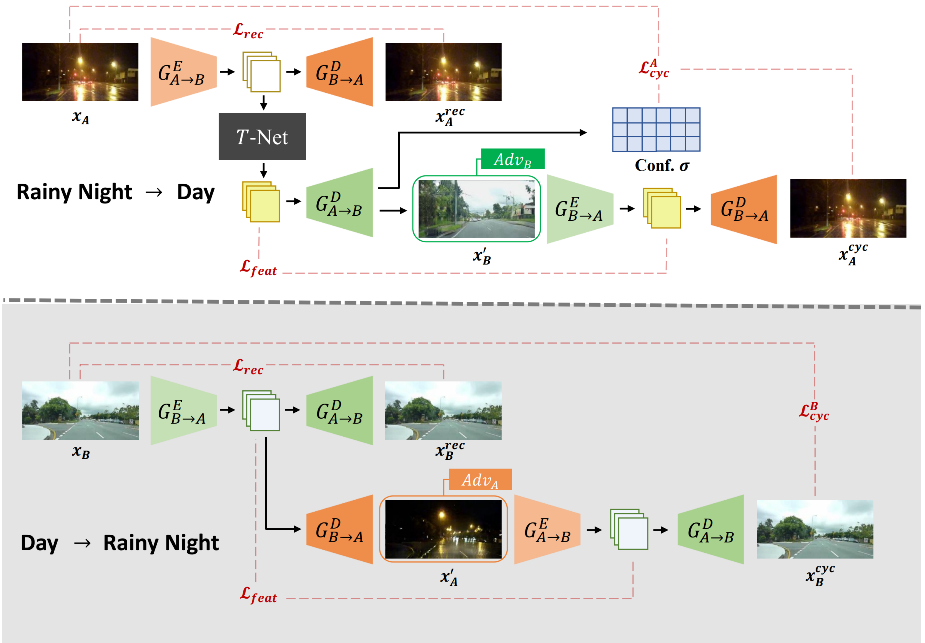


Fig. 3: An overview of AU-GAN's model (Courtesy Jeong-gi Kwak et al. [8]). The upper side depicts the procedure for translating a rainy night image → day, and the bottom side depicts the procedure for translating a day image → rainy night. (G stands for generator, and the superscripts E and D specify whether G is an encoder or decoder respectively.)

Cycle consistency means that the mapping functions $G_{A\to B}$ and $G_{B\to A}$ should (approximately) be each other's inverse. The cycle consistency loss encourages this cycle consistency. By exploiting the cycle consistency property we can do translation procedures such as $A \to B \to A$, and end up with approximately the same initial image A. Most CycleGAN based models also include a symmetrical opposite translation $B \to A \to B$ for stability and balance optimization. ForkGAN [12] expands upon this by proposing a fork-shaped generator, i.e. one encoder and two decoders, that disentangles the domain-specific and domain-invariant information.

In accordance with these ideas, but deviating from existing methods, AU-GAN proposes an asymmetric framework for image-to-image translation. In their paper, the authors of AU-GAN [8] explain the rationale behind this alteration. That is, let's say the encoder $G^E_{A\to B}$ could extract domain-invariant features. With these domain-invariant features, the reconstructed image x_A^{rec} and transferred image x'_B are generated by $G^D_{B\to A}$ and $G^D_{A\to B}$ respectively, and then $G_{B\to A}$ generates the cyclic image x_A^{cyc} from x'_B. During training, minimizing the difference between the original and generated image is included in the objective. This is done by aiming for a low reconstruction loss (L_{rec}) and cycle-consistency loss (L_{cyc}). However, if the encoder extracts 'truly' domain-invariant features, it is not feasible to fully reconstruct the original adverse weather image due to some negative domain-specific properties such as rain, noise and reflections being discarded from the feature. It is therefore key to retain some domain-specific properties for the image reconstruction phase, but to not include them in the translation. To that end, an additional transfer network (T-net), which is comprised of residual blocks [6], is inserted into $G_{A\to B}$ to extract an enhanced and disentangled feature for domain translation. As a result, the two domain translation functions, i.e. $f_{A\to B}$ and $f_{B\to A}$, are not symmetrical [8], because of presence of the transfer function $T()$ in equation Eq. 1 and its absence in Eq. 2:

$$x'_B = f_{A\to B}(x_A) = G^D_{A\to B}(T(G^E_{A\to B}(x_A))) \tag{1}$$

$$x'_A = f_{B\to A}(x_B) = G^D_{B\to A}(G^E_{B\to A}(x_B)) \tag{2}$$

The transfer function $T()$ is also present in the asymmetric feature matching loss ($\mathcal{L}_{feat}$) for disentanglement is presented. This loss describes the discrepancy between the encoded feature of the input image and that of the transformed image:

$$\begin{aligned}\mathcal{L}_{feat} = \mathbb{E}_{x_A}[\parallel T(G^E_{A\to B}(x_A)) - G^E_{B\to A}(x'_B) \parallel_1] + \\ \mathbb{E}_{x_B}[\parallel (G^E_{B\to A}(x_B)) - G^E_{A\to B}(x'_A) \parallel_1]\end{aligned} \tag{3}$$

More details can be found in [1] and [8].

3 Methodology

3.1 Motion-based cropping

To train AU-GAN, it needed to be provided with many images from the source and desired target domain, i.e. day and night imagery. This complicated matters, as this project was initiated due to a lack of night imagery. Luckily for AU-GAN, the requirements on training data are less demanding than those for FlowCube's re-id model. Whereas the pseudo-labeling algorithm was required to produce high quality tracklets, AU-GAN only required any imagery from the two domains it needed to translate between. One could for example provide AU-GAN with images from the sky during the day and during the night, and AU-GAN would be able to use this imagery to train itself. However, since AU-GAN would later need to transform vehicles, cyclists and pedestrians in order to augment the re-id training dataset, it was still deemed appropriate to use imagery of such objects for AU-GAN's training data.

In the day- and nighttime video data, the bulk of the content is not relevant for the re-identification task, such as the static scenery of sky imagery. To focus the augmentation's learning to the task-relevant part of the content, the training data was filtered to the regions where movement was present.

To that end, a custom cropping algorithm utilized frame differencing to highlight all moving objects in the scene. Then, K-means clustering with a (high) fixed K was used to identify clusters of movement. Afterwards, agglomerative clustering with a distance threshold was used to merge clusters that described the same object. This resulted in the newly formed MBC dataset (see Table 1), containing day- and nighttime imagery that could be used to train AU-GAN.

3.2 AU-GAN data augmentation

With AU-GAN trained and the re-id training dataset's daytime imagery identified, the desired day-to-night augmentation could now be performed. Figure 4 illustrates some of these translations with their corresponding confidence maps.

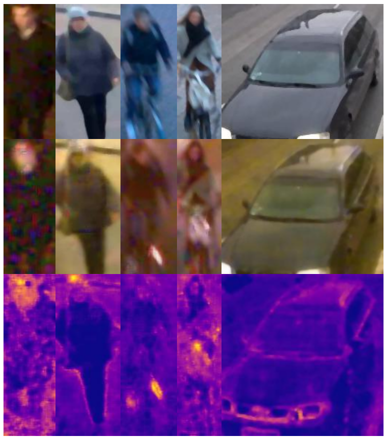


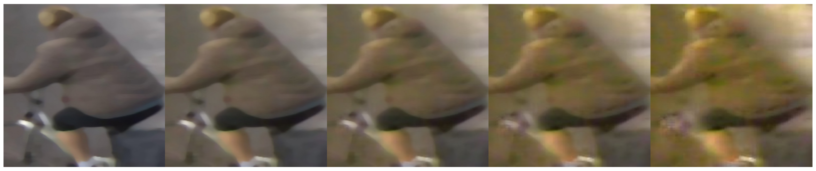
Fig. 4: Examples of AU-GAN's input imagery (top), i.e. imagery from FlowCube's re-id training dataset, their respective AU-GAN transformed nighttime equivalent (middle), and the corresponding confidence maps (bottom).

3.3 Alpha blending

Since AU-GAN could transform any image to the nighttime domain, we could potentially double the size of FlowCube's current re-id training dataset with additional synthetic nighttime re-id imagery. Moreover, by incorporating the gradual transition between the original image and the AU-GAN transformed image, the effective dataset could grow even larger. This was done with the use of an alpha blend. Alpha blending is a data-augmentation technique which is computational inexpensive but provides a diversity of training samples which are realistic enough to be encountered in reality (see Fig. 5). Alpha blending takes two images and blends them together with a ratio $\alpha \in [0, 1]$, that specifies the transparency (1 - α) and opaqueness (α) of the two images I_* respectively, as can be seen in equation 4:

$$I_{blend} = (1 - \alpha) \cdot I_1 + \alpha \cdot I_2 \tag{4}$$

The use of alpha blending provided a more continuous approach to the AU-GAN data augmentation. With that, it facilitated an exponentially larger number of possible augmentations to incorporate into the re-id dataset, as can be seen in Figure 5.



(a) α=0 (b) α=0.25 (c) α=0.5 (d) α=0.75 (e) α=1

Fig. 5: Illustration of the influence of α with alpha blending.
(α=0 yields the original image and α=1 yields the AU-GAN transformed image)

3.4 Performing Alpha blending

The alpha blending procedure was implemented to transpire while training the re-id model. That is, for each image in the re-id training set, if it was marked to be alpha blended, also its AU-GAN transformed equivalent was loaded. Then, an alpha blend between the two images – with $\alpha \in [0, 1]$ – was applied in a random uniform manner, where $\alpha \leq \alpha_{max} \in [0, 1]$. This alpha blended image was then added to the re-id training batch. For reproducibility, the random uniform value for α was ensured to, for each epoch-image combination, be similar across re-id training runs with the use of a seed.

For the experiments, AU-GAN was pretrained on the motion-based cropped (MBC) dataset (Table 1). Training was performed using 8 epochs with a batch size of 1. During training, images were reduced from their original size of 500x500 to a 'fine size' of 224x224 (i.e. the image size that the re-id model uses) by randomly cropping an area of 224x224 from the original image. Furthermore, following [8], we adopt an Adam [7] solver for which we set β_1=0.5, β_2=0.999 and $\epsilon = 1e^{-08}$. Coefficients of the full model objective were then set to $\lambda_{feat} = 1$ and $\lambda_{rec} = \lambda_{cyc}$, and the learning rate = 0.0002.

With that, after training, AU-GAN was used to transform the TrainO to TrainA image-to-night (I2N) translation (see Table 1). These datasets were used for training. The result of this training was checked with the re-id F1 scores on the validation datasets, i.e. Val and Val_night, where the 'baseline' represents the current implementation of the re-id algorithm, without any AU-GAN data augmentation. Each specific setup of an experiment was repeated 8 times, and validation results were averaged over the last 10 training steps of the re-id model in an effort to reduce the influence of noise, caused by randomness in the stochastic training process.

Table 1: An overview of the used datasets. The MBC dataset holds the imagery generated by the motion-based cropping algorithm. TrainO is the original re-id training dataset, and TrainA is its (image-to-night) AU-GAN transformed equivalent. Val is the re-id validation dataset, and Val_night is a subset of Val, holding only its nighttime imagery.

Dataset name	**# images**	**# identities**	**# sites**	**day/night ratio**
MBC	713141	-	-	0.70/0.30
TrainO	262403	2693	29	0.79/0.21
TrainA	62403	2693	29	0.00/1.00 *
Val	3928	112	25	0.84/0.16
Val_night	612	19	11	0.00/1.00

* Nighttime imagery is synthetic.

4 Experiment results

4.1 Augmentation ratio experiments

In the first experiment, it was tested how adding raw AU-GAN image-to-night transformed imagery to the training set, without any alpha blending, affected the re-id validation F1 scores on both Val and Val_night compared to the baseline. To that end, we set the augmentation ratio $\in$ [0, 0.10, 0.20, 0.30, 0.40].

As can be seen in Figure 6, on Val the baseline produced a mean re-id F1 score of 91.3 and a spread of 4.3. In contrast, for augmentation ratios 0.10-0.40 the mean re-id F1 scores were lower at 88.6, 87.5, 86.7 and 85.8 with a spread reduced to 1.7, 1.7, 2.6 and 3.7 respectively. As for why adding the raw AU-GAN transformed imagery to the training dataset decreased the mean re-id performance but improved the spread. It was thought that adding some amount of synthetic nighttime imagery provided the re-id model with a useful amount of hard (synthetic) nighttime training examples that made the predictions more stable. However, too many of such hard nighttime examples caused the model to put too much emphasis on them. As such, this would have the adverse effect of confusing the model.

On Val_night, the baseline produced a mean re-id F1 score of 89.5 and an even higher spread of 9.6. In comparison, for augmentation ratios 0.10-0.40, the mean re-id F1 scores turned out higher at 91.6, 91.0, 90.4 and 89.2, and the spread was reduced to 4.4, 5.6, 3.8 and 6.7 respectively. Overall, we see that augmentations with a small ratio can yield a small improvement on the re-id results at night. The presumed reason for this is that, contrary to Val, the Val_night dataset relatively contained more imagery that was similar to the AU-GAN generated imagery, thus the re-id model performed better here.

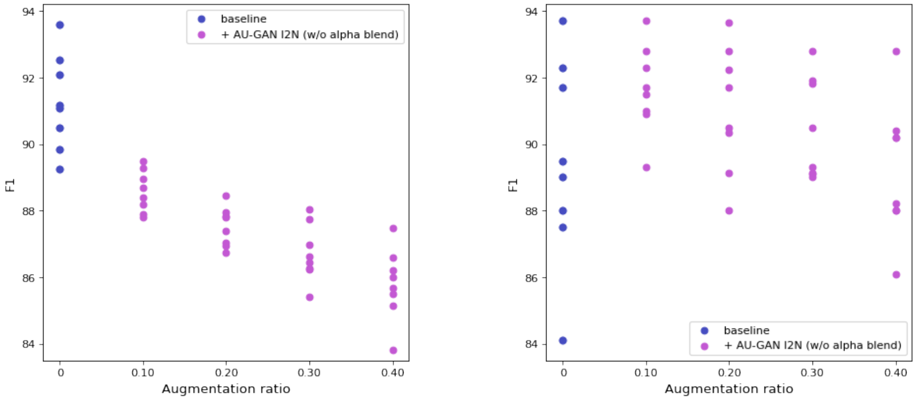


Fig. 6: The re-id F1 scores on the Val (left) and Val_night (right) dataset, trained on TrainO and TrainA using varying augmentation ratios.

4.2 Alpha blend experiments

Since simply increasing the augmentation ratio without alpha blending increased the performance for Val_night only slightly, and actually decreased performance for the Val dataset, now an experiment with alpha blending was performed. We review the influence of α_{max} when alpha blending TrainO to TrainA versions, with the aim of optimizing the augmentation. For this experiment, the augmentation ratio was set to 1.00, and $\alpha_{max} \in [0, 0.15, 0.25, 0.35, 0.50, 0.75, 1.00]$.

In Figure 7, we see that for Val, the baseline produced a mean re-id F1 score and spread of 91.3 and 4.3 respectively. In comparison, α_{max} values of 0.15 and 0.25 produced higher mean re-id F1 scores of 91.5 and 91.9, and lower spreads of 1.6 and 1.4, respectively. For α_{max} values equaling $0.35, 0.50, 0.75$ and 1.00, the mean re-id F1 scores declined again to 91.3, 90.4, 88.5 and 86.6, with increased spreads of 1.9, 3.6, 1.7 and 2.0 respectively. With that, we identify an arch-like trend with an optimum at $\alpha_{max} = 0.25$. If we reason why this arch-like trend occurred, logically, we can say that the higher values for α_{max} generated harder, i.e. darker and more noisy, nighttime training imagery. As we saw in Fig. 7, such training imagery could already make predictions more stable on Val. However, now we also see that, for lower magnitudes of α_{max}, the mean re-id F1 score can also be improved. Presumably by providing more difficult – but not too difficult – training data, making the re-id model more stable and better performing.

When the same experiment was repeated for Val_night, we can again see that the baseline produced a mean re-id F1 score and spread of 89.5 and of 9.6 respectively. Moreover, it can be seen that $\alpha_{max} = 0.15$ produced the highest mean re-id F1 scores of 92.1 and a spread of 3.0. As for α_{max} values for $0.25, 0.35$ and 0.50, the mean re-id F1 score decreased to 91.4, 90.6 and 89.6 with greatly diminishing spreads of 4.4, 2.8 and 0.9 respectively. After that, for α_{max} values of 0.75 and 1.00, the mean re-id F1 scores increased slightly to 89.9 and 90.1, with higher spreads of 4.8 and 3.7 respectively. With that, it is apparent

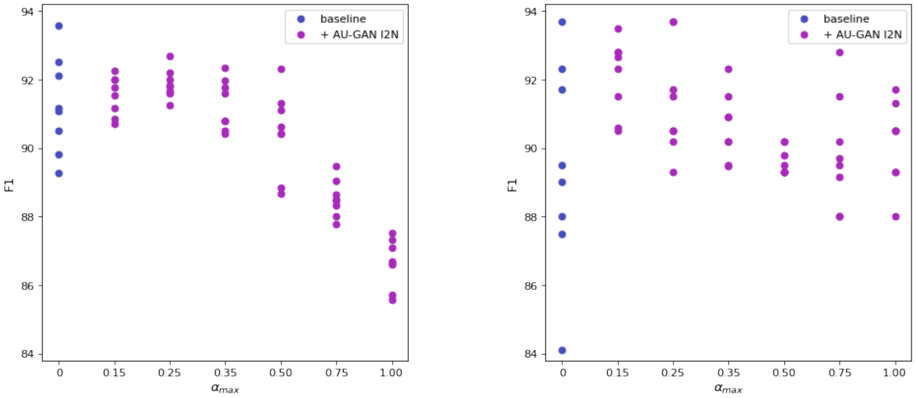


Fig. 7: The re-id F1 scores on the Val (left) and Val_night (right) dataset, trained on TrainO and alpha-blended TrainA with varying values for α_{max}.

that the AU-GAN image-to-night augmentation was able to improve both the mean re-id performance and the spread. As for why performance decreased for higher values for α_{max}, it could be that the nighttime training imagery for AU-GAN was darker/more noisy than that in Val_night. Or, since the image-to-night translation also includes night-to-night translations, it could be that the night-to-night augmentations resulted in a magnified 'night effect' that was too strong for higher α_{max} values, which confused the re-id model.

5 Conclusion

In this study, the usage of AU-GAN data augmentation for improving FlowCube's re-id performance was introduced. To that end, FlowCube's re-id training dataset was augmented with synthetic AU-GAN generated nighttime imagery. This imagery was generated by transforming the original re-id training dataset into an AU-GAN transformed equivalent. During training, the original and AU-GAN transformed imagery was then alpha blended to create an augmentation with a modifiable magnitude α, which greatly enlarged the effective training dataset's size.

This made it possible to demonstrate that the AU-GAN image-to-night augmentation, alpha blended with the original imagery, was able to increase the mean nighttime re-id F1 score, maintain the same mean general re-id F1 score, and greatly reduce the variance in performance between training runs for both general and nighttime re-id. The proposed AU-GAN augmentation shows much more consistency and reproducibility.

Overall, we can conclude that the AU-GAN augmentation, combined with an alpha blend, can positively affect FlowCube's re-id performance, when applied in a low magnitude, i.e. with $\alpha_{max} \in [0.15, 0.25]$. The augmentation introduces color adjustments and adds noise, which at the least have a positive effect on robustness; an useful characteristic for any camera-based traffic sensor.

References

1. Amersfoort, M.: Improving FlowCube's re-identification performance with GAN data augmentation. Master's thesis, Vrije Universiteit - Universiteit van Amsterdam (August 2022)
2. Chong, C.Y., Mori, S., Govaers, F., Koch, W.: Comparison of tracklet fusion and distributed kalman filter for track fusion. In: 17th International Conference on Information Fusion (FUSION). pp. 1–8 (October 2014)
3. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial networks. Commun. ACM **63**(11), 139–144 (November 2020)
4. Grabner, H., Bischof, H.: On-line boosting and vision. In: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06). vol. 1, pp. 260–267 (June 2006)
5. Guan, Y., Chen, X., Yang, D., Wu, Y.: Multi-person tracking-by-detection with local particle filtering and global occlusion handling. In: 2014 IEEE International Conference on Multimedia and Expo (ICME). pp. 1–6 (September 2014)
6. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2016)
7. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: 3rd International Conference on Learning Representations, (ICLR) (May 2015)
8. Kwak, J., Jin, Y., Li, Y., Yoon, D., Kim, D., Ko, H.: Adverse weather image translation with asymmetric and uncertainty-aware GAN. In: 32nd British Machine Vision Conference 2021 (BMVC). p. 96. BMVA Press (November 2021)
9. Technolution: Flowcube: betrouwbare informatie over fietsverkeer en voetgangers. Intertraffic special, Mobiliteits Platform (April 2020)
10. Wahba Tadros, S.N., et al.: Demographic trends and urbanization. World Bank Group (April 2021)
11. Yang, F., Chang, X., Dang, C., Zheng, Z., Sakti, S., Nakamura, S., Wu, Y.: Remots: Self-supervised refining multi-object tracking and segmentation. arXiv preprint 2007.03200 (January 2021)
12. Zheng, Z., Wu, Y., Han, X., Shi, J.: Forkgan: Seeing into the rainy night. In: Computer Vision – ECCV 2020. pp. 155–170. Springer International Publishing (December 2020)
13. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: 2017 IEEE International Conference on Computer Vision (ICCV). pp. 2242–2251 (2017)