Using Synthetic Data for Person Tracking Under Adverse Weather Conditions

Abdulrahman Kerim[a,c,*], Ufuk Celikcan[a,*], Erkut Erdem[a], Aykut Erdem[b]

[a]*Hacettepe University, Department of Computer Engineering, Ankara, Turkey*
[b]*Koç University, Department of Computer Engineering, Istanbul, Turkey*
[c]*Lancaster University, School of Computing and Communications, UK*

Abstract

Robust visual tracking plays a vital role in many areas such as autonomous cars, surveillance and robotics. Recent trackers were shown to achieve adequate results under normal tracking scenarios with clear weather condition, standard camera setups and lighting conditions. Yet, the performance of these trackers, whether they are correlation filter-based or learning-based, degrade under adverse weather conditions. The lack of videos with such weather conditions, in the available visual object tracking datasets, is the prime issue behind the low performance of the learning-based tracking algorithms. In this work, we provide a new person tracking dataset of real-world sequences (PTAW172Real) captured under foggy, rainy and snowy weather conditions to assess the performance of the current trackers. We also introduce a novel person tracking dataset of synthetic sequences (PTAW217Synth) procedurally generated by our NOVA framework spanning the same weather conditions in varying severity to mitigate the problem of data scarcity. Our experimental results demonstrate that the performances of the state-of-the-art deep trackers under adverse weather conditions can be boosted when the available real training sequences are complemented with our synthetically generated dataset during training.

Keywords:
Person Tracking, Synthetic Data, Rendering, Procedural Generation

*Corresponding authors
Email addresses: a.kerim@lancaster.ac.uk (Abdulrahman Kerim), ufuk.celikcan@gmail.com (Ufuk Celikcan), erkut@cs.hacettepe.edu.tr (Erkut Erdem), aerdem@ku.edu.tr (Aykut Erdem)

1. Introduction

Recently, convolutional neural networks (CNN) have shown a remarkable progress in various computer vision tasks such as object detection [1], object tracking [2], semantic segmentation [3], depth estimation [4], optical flow estimation [5], and person Re-Identification (ReID) [6]. While utilizing CNNs for computer vision field can improve both generalizability and accuracy, CNNs have an intrinsic restriction in terms of the data needed for training. Usually, better performance comes with deeper and larger CNNs which give a higher degree of non-linearity and more freedom in solving complex tasks. However, that introduces more variables for tuning. Unfortunately, training such models requires more data and more powerful computing devices. The introduction of cheap general purpose graphics processing units (GPGPUs) alleviated hardware limitation. However, the scarcity of large-scale datasets for training supervised learning methods remains as the main bottleneck for many computer vision tasks, especially, the ones that require enormous efforts for annotation, such as semantic segmentation and visual object tracking. Besides, for some others such as optical flow and depth estimation, it becomes extremely hard or even impossible to provide large-scale annotated datasets.

In addition to the need for large scale datasets, another requirement is a high level of diversity to allow deep learning models to work well in practice and not overfit to certain attributes. However, obtaining suitable datasets that are large and diverse from real world is not a simple task. Thus, small scale and mostly normal attributes tend to be the main features of the available datasets. Consequently, most of the available datasets tend to focus on normal scenarios under typical light conditions and camera parameters. The first reason behind this is the assumption that the computer vision model is going to be tested under these normal circumstances such as clear sky, optimal lighting, and standard recording conditions. While the second is the difficulty of obtaining datasets under rare conditions. Unfortunately, training computer vision models under these normal conditions causes unexpected behaviour or complete failure in adverse conditions.

Visual object tracking (VOT) is one of the major tasks in computer vision field that is essential for higher-level tasks such as pedestrian detection, action recognition, or trajectory estimation. Therefore, it is vital for many real-world systems such as self-driving vehicles, automated retail or visual surveillance. Failure of such systems under adverse conditions can lead to property damages or human injuries.

In this work, we focus on person tracking under adverse weather conditions such as snowy, rainy and foggy weather conditions. Thereby, to assess the performance of the state-of-the-art trackers in person tracking in video feeds taken under such adverse conditions, we collect a novel real dataset, PTAW172Real, that consists of 172 videos featuring weather with heavy snow, rain or fog. Our experiments expose the poor performance of the state-of-the-art trackers when tested on PTAW172Real and this can be linked to the limited number of videos taken under adverse weather conditions in the current VOT datasets that these trackers were trained with. We offer a remedy for the lack of data availability by using our NOVA engine to generate a synthetic dataset, PTAW217Synth, that provides diverse and rich training sequences under adverse weather conditions. To the best of our knowledge, no work has been done to validate the usability of synthetic data for person tracking under adverse weather conditions. In this work, we show that using synthetic data, we can bridge the aforementioned gap and improve the performance of the learning-based trackers under adverse weather conditions.

Our main contributions in this paper can be summarized as follows:

- We present a novel real dataset called PTAW172Real for visual object tracking under adverse weather conditions. The dataset contains 172 videos manually annotated covering snowy, rainy and foggy weather conditions.

- We highlight the poor performance of the state-of-the-art trackers under adverse weather conditions with PTAW172Real.

- Using our NOVA rendering engine, we procedurally generate a new dataset called PTAW217Synth made up of synthetic sequences under adverse weather conditions complete with automatically-generated per-frame annotations including bounding boxes at pixel-level accuracy, occlusion state and other relevant metadata such as time-of-day and camera type. The dataset consists of 217 sequences for person tracking spanning the three adverse weather conditions.

- We show that fine-tuning the pre-trained models on our synthetic dataset PTAW217Synth is able to improve the performance of the deep trackers. Similarly, we also show that training from scratch on only our synthetic training dataset can achieve comparable results to training on large scale real datasets.

2. Related Work

Despite the fact that deploying synthetic data in computer vision field has just started recently, a number of works investigated the usability of synthetic data for different computer vision tasks. In general, synthetic data can be employed for both training and testing purposes. For training, they can be used as the only training data, or to augment the real ones. It is possible to apply synthetic data for pre-training or fine-tuning learning models as well.

One work [7] investigated the usability of synthetic data for instance segmentation and object detection. They concluded that training on both synthetic and real data achieves better results as compared to training on a small set of real data. At the same time, they show that fine-tuning on their augmented data can achieve even better results. Similarly, Cheung et al. [8] proved that synthetic data can be used with real data to improve accuracy for crowded scene understanding. They show that using their generated synthetic dataset, LCrowdV, with real datasets can improve the accuracy as compared to using these real datasets alone.

Varol et al. [9] demonstrated the usability of synthetic data for human depth estimation and part segmentation. They prove that training on synthetic and real images increases the accuracy for semantic segmentation and reduces the root-mean-squared-error for depth estimation. In the same way, Barbosa et al. [10] extensively studied the advantages of using their generated synthetic dataset, SOMAset, for the task of person ReID. They show that pre-training on their synthetic dataset then fine-tuning on real datasets achieves better results as compared to training only on real datasets.

Under the scope of visual object tracking, Gaidon et al. [11] provided a detailed analysis on the advantages of using synthetic data for the task of multi-object tracking. They show that training on their synthetic dataset then fine-tuning on real datasets achieves the best results as compared to only training on synthetic or real datasets.

Similarly, Zhang et al. [12] used image-to-image translation method to generate synthetic thermal infrared tracking videos using the RGB ones. They show that training on their synthetic videos then fine-tuning on real ones or training on both synthetic and real videos achieve better results as compared to training on the available small scale real datasets.

Figure 1: On the left half, sample frames from the currently-available real (top-left quarter) [13, 14, 15, 16] and synthetic (bottom-left quarter) [17, 18, 19, 11] visual object tracking datasets demonstrate the lack of adverse weather conditions. The right half presents sample frames from sequences spanning raining, foggy and snowy weather conditions from PTAW172Real (top-right quarter) and PTAW217Synth (bottom-right quarter) datasets that we introduce in this work.

Similar to the previously mentioned works, we also investigate the advantages of using synthetic data for training learning-based models. However, this work sheds light on the limitations of the available real and synthetic visual object tracking datasets. As shown in Fig. 1, the adverse weather conditions seem to be underrepresented in most of the available real and synthetic VOT datasets. This causes the state-of-the-art trackers perform poorly under these challenging weather conditions. Bearing this in mind, we present synthetic data as a legitimate solution for the lack of the adverse weather conditions in the real datasets. To this end, we utilize our procedural content generation engine NOVA to generate a visual object tracking dataset to be used in the training of general purpose visual object trackers. The generated dataset is specifically designed for tracking people under adverse weather conditions in outdoor environments.

3. Extensions to NOVA Framework

To procedurally generate synthetic sequences of pedestrians under adverse weather conditions, we use the NOVA rendering engine [20], which is designed with the goal of allowing researchers with no experience in computer graphics to generate high quality datasets with accurate and dense annotations. NOVA operates in two modes. The first is to generate a single sequence while the other is to generate a full dataset. The first mode gives the user full control of the sequence being generated where it is possible to specify the environment, the weather condition, time of day, camera type, number of cars and number of pedestrians and their density. The dataset mode requires nothing to be specified except the number of sequences to be generated so that NOVA varies the other parameters automatically.

For the particular task of person tracking this work deals with, NOVA generates, for each frame, a bounding box specifying the exact location of the person(s) being tracked in the frame and the occlusion state, that is, whether any other object or person in the scene occludes the person(s) being tracked at that instant. In addition to these, a supplementary metadata are provided with each sequence denoting the environment, weather condition, time of day, camera type, number of people and cars and people density.

One of the major highlights of NOVA is its capacity to procedurally generate highly diverse and photorealistic sets of synthetic humans. So much so that, each generated human is practically unique in appearance due to the practically infinite number of recipes (combinations of parameters that are assigned randomly on the fly but in cohesion with each other) that NOVA uses in creating them. In this work, we further develop this aspect of NOVA by incorporating premade synthetic humans from Microsoft Rocketbox Avatar Library [21].

Since the main aim of this work is to enhance the performance of the trackers under adverse weather conditions, we also extended other capabilities of NOVA toward photorealistic simulation of the generated humans under adverse weather conditions. The environment is built to change dynamically to match the corresponding weather condition and time of the day. Accordingly, the textures of buildings are changed to have lit windows at nighttime. Furthermore, we implemented the following for the three weather conditions to facilitate the generation of synthetic sequences with similar visual characteristics to the ones observed in the real-world videos captured under adverse weather conditions.

Snowy Weather Condition. First, the variety of clothing used to generate humans in snowy weather is restricted only to outdoor cold-weather clothes. At the same time, humans are randomly assigned umbrellas. An umbrella is attached to the right or left hand at random. The animation of the character is set to match the umbrella mode, *i.e.*, open or close. Snow tracks left by cars and pedestrians are simulated. Furthermore, snow banks and melt snow are created on the pavements and roads to give a higher degree of realism. For this, a set of street light poles in the scene are selected at random to determine the positions of the snow banks. Then, from a predefined set of snow banks, one snow bank is instantiated for each position. After that, snow materials are assigned at random to the snow banks. Following this, the scale and rotation of these models are randomized to allow for even more diversity. On the other hand, the melt snow is simulated by the same snow shader that is used to simulate accumulated snow but with the accumulation parameter set to a random smaller number than the one used for accumulated snow. Making use of the particle system and post-processing effects, falling snow particles and blizzard were randomly introduced to the simulation, as well.

Rainy Weather Condition. Similar to the snowy weather condition, humans in rainy weather are also generated with outdoor cold-weather clothes; and umbrellas are given to some of the generated humans in the same way. In addition, water puddles are simulated to account for water accumulation due to the rain. This is realized by using a puddle shader that is assigned to some of the ground materials (pavements, roads etc.) randomly. For the heavy rain, the rain splash is activated and additional water puddles are instantiated from a predefined set of water puddles. Rain drops are generated using the particle system. Furthermore, rain drops falling on camera lens are simulated using post-processing effects to match the characteristic of the rainy videos in real life.

Foggy Weather Condition. The clothes of the people produced in the foggy weather simulation are not limited to a specific category, but instead are randomly selected. Additionally, the fog is simulated using post-processing effects and the Enviro system [22]. The fog density is randomized at run time to give more diversity.

Motion Blur and Chromatic Aberration. These camera effects were

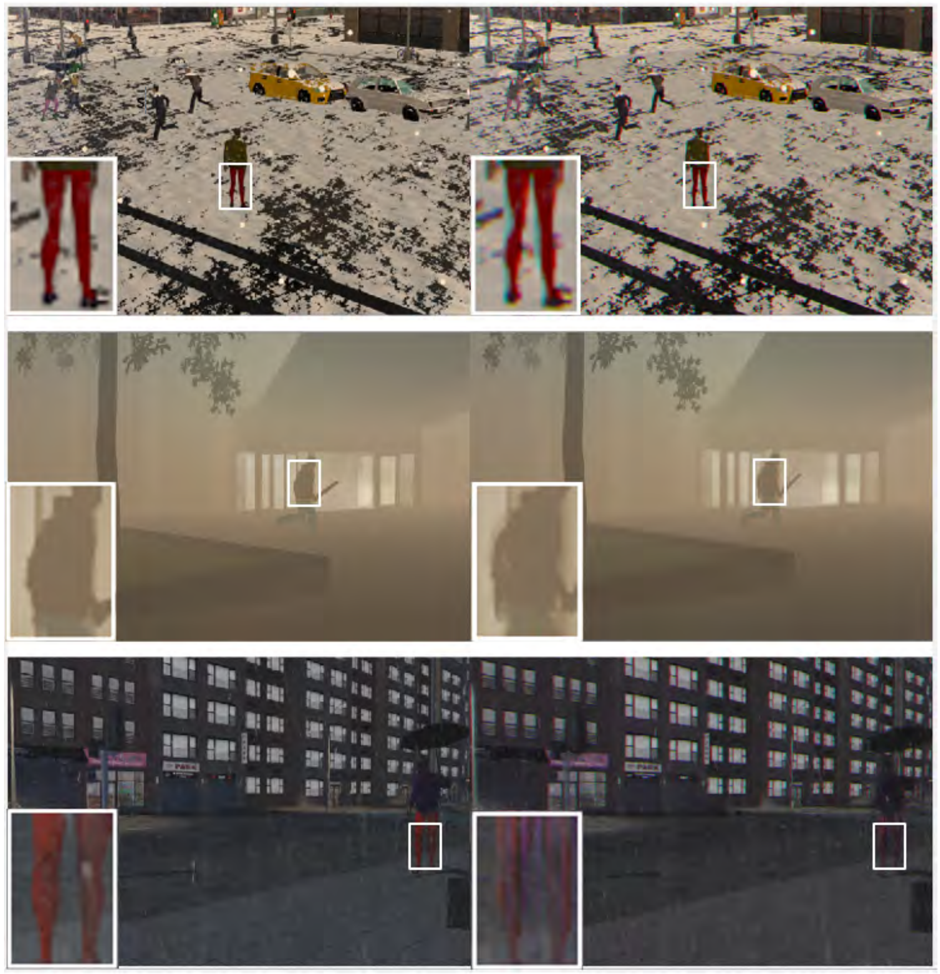

Figure 2: Chromatic aberration, motion blur and both effects are demonstrated in the first, second and third rows, respectively. The first column shows the original frame while the second displays the result of applying the effect(s).

simulated additionally to match the camera degradation observed in real-life adverse weather videos. Using post-processing, NOVA simulates these two effects procedurally and parametrically. Thus, how severe the effect of these two degradations is randomly configured at run time to provide further diversity in the generated synthetic sequences. In Fig. 2, the impact of using these effects over the generated sequences is shown with a sample of images.

Table 1: Dataset statistics of PTAW172Real.

Class	Min Frames	Max Frames	Mean Frames	Total Frames	Videos
Rain	108	1755	498	31888	64
Snow	113	960	394	24010	61
Fog	106	750	328	15394	47
All	106	1755	407	71292	172

4. PTAW172Real and PTAW217Synth Datasets

4.1. Real-World Data Collection for PTAW172Real

For the aim of analysing the performance of the recent general purpose visual trackers under adverse weather conditions, we collected real-world videos from YouTube spanning snowy, rainy and foggy weather. Keywords such as *"adverse"*, *"extreme"*, *"heavy"*, and *"severe"* were used together with the weather names to initiate searches on the Youtube video-sharing platform. Following this, the query results were checked and only the videos satisfying the adverse weather conditions were selected. The acquired videos were edited to assure that the object is not occluded and clearly visible in the initial frame. At the same time, the lengths of the videos were modified as needed to keep them around 400 frames per video to provide compatibility with the sequences in the available visual object tracking datasets. Statistics showing the minimum, maximum, average and total number of frames are given in Table 1. The number of videos in the dataset is 172 and the total number of frames is over 71 thousand. The collected videos are at 24 frames per second (FPS) and average time period per sequence is around 17 seconds. Sample frames from the collected PTAW172Real dataset are shown in Fig. 3.

We used the VGG Image Annotator tool [23, 24] for annotating the dataset. We annotated every 5th frame be drawing a bounding box around the person of interest. The accessories such as handbag etc. that a person can carry were excluded and the tightest box was drawn. When the person was partially or fully occluded, the estimated location of the person was considered. Additionally, each video was associated with four attributes regarding object occlusion, scale change, background clutter and abrupt camera motion. Fig. 4 gives the hierarchical distribution of the attributes in PTAW172Real dataset.

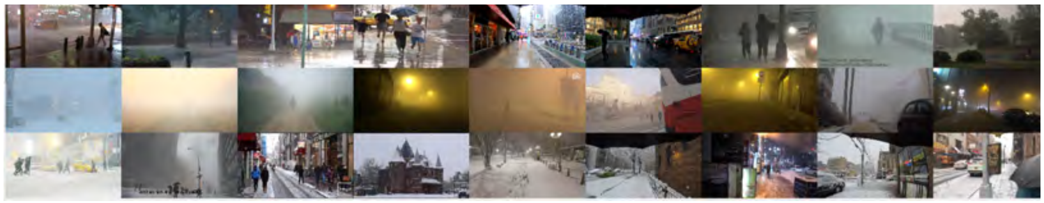

Figure 3: PTAW172Real, our real person tracking dataset, consists of 172 sequences. Each row shows a specific adverse weather condition, namely rain, fog, and snow.

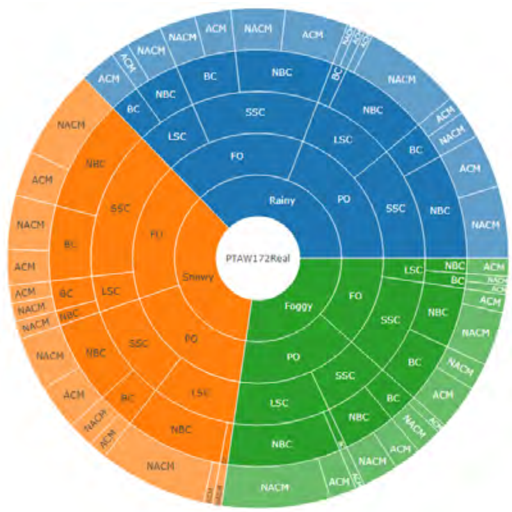

Figure 4: The sunburst chart shows the different attributes distribution across PTAW172Real dataset. The inner circle shows the weather conditions, outer circles show occlusion (FO:Full Occlusion, PO: Partial Occlusion), scale change (LSC: Large Scale Change, SSC: Small Scale Change), background clutter (BC: Background Clutter, NBC: No Background Clutter) and abrupt camera motion (ACM: Abrupt Camera Motion, NACM: No Abrupt Camera Motion).

4.2. Synthetic Data Generation for PTAW217Synth

PTAW217Synth employed in the experiments to train the deep learning trackers consists of 217 synthetic sequences that were generated using the NOVA rendering engine. NOVA allows to specify the attributes of the sequences to be generated. In this work, we configured these attributes to match our goal of generating diverse synthetic sequences under adverse weather conditions. Accordingly, the weather conditions were limited to snowy, rainy and foggy weather. The virtual camera type to capture the

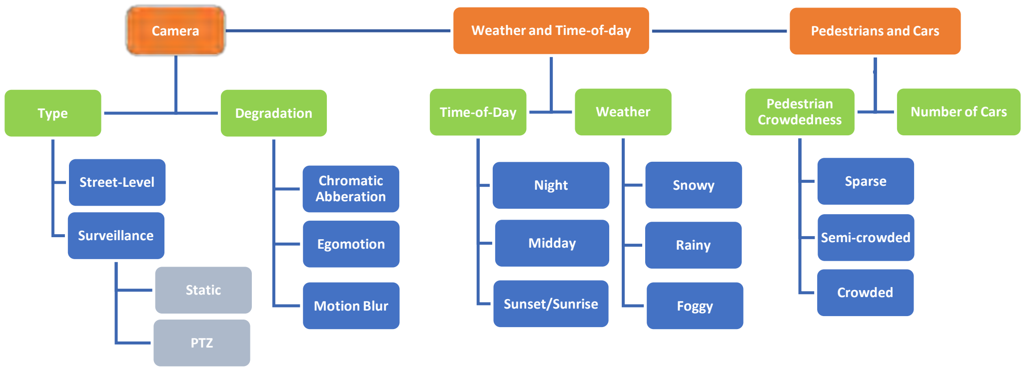

Figure 5: Hierarchical view of the attributes across our training synthetic person tracking dataset, PTAW217Synth, generated by using NOVA.

Figure 6: PTAW217Synth, our training synthetic tracking dataset, consists of 217 sequences, each with a unique set of attributes. Random frames are shown here, illustrating the variations in crowdedness, camera altitude, weather conditions and times of day.

simulations was set as either the street-level camera or the surveillance camera. The simulation environment was limited to the streets of an urban center, since such are the most common settings in the real-world visual object tracking datasets. In parallel to this, all other attributes such as time of day and crowdedness were randomised to ensure the diversity of the generated sequences. The attributes of the generated synthetic sequences are given in Fig. 5. Consequently, the diversity of the generated sequences can be noted in the sample images from these sequences in Fig. 6.

Further information regarding the minimum, maximum, average and total

Table 2: Dataset statistics of PTAW217Synth.

Class	Min Frames	Max Frames	Mean Frames	Total Frames	Videos
Rain	490	510	501	34538	69
Snow	490	510	501	37577	75
Fog	490	510	499	36432	73
All	490	510	500	108547	217

Figure 7: The figure demonstrates the weather variations simulated in PTAW217Synth. The first and second rows present different view points of the same location. Each group of 2x2 images shows one weather condition (from left to right: rainy, foggy, and snowy) in increasing adversity while the leftmost image shows the same location in clear weather.

number of frames are shown in Table 2. The overall average number of frames per sequences is 500 which translates to a duration of 21 seconds as the sequences were generated at 24 FPS. The total number of frames of the 217 sequences within the dataset is more than 108 thousand. We should note that PTAW217Synth has an even distribution of sequences across the rainy, snowy and foggy weather conditions. The sample images captured at a single location from two different view points given in Fig. 7 further demonstrate the variety of the simulated weather conditions.

A visual comparison between PTAW172Real and PTAW217Synth datasets is given in Fig. 8. In each row a specific weather condition is presented. Both datasets exhibit similar visual characteristics for the three weather conditions. The figure also demonstrates the level of photorealism of the PTAW217Synth dataset.

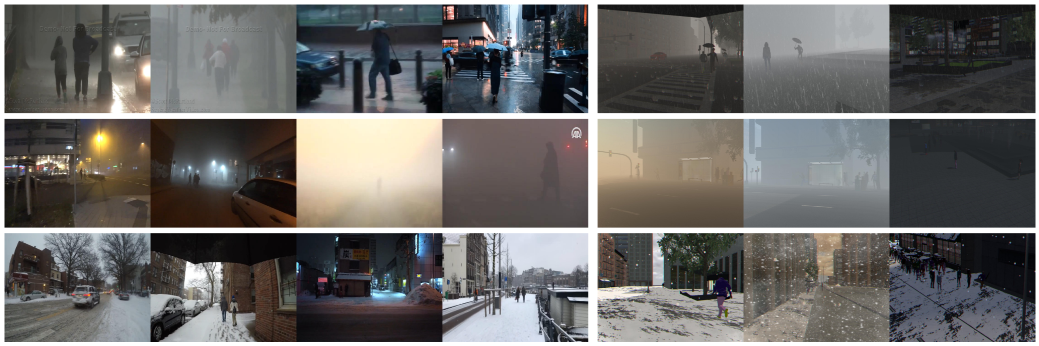

Figure 8: A visual comparison among the synthetic PTAW217Synth (to the right) and real PTAW172Real (to the left) datasets. Each row demonstrate specific weather condition (from top to bottom: rainy, foggy, and snowy).

5. Experiments

In this section, we study the performance of the state-of-the-art visual trackers in adverse weather conditions. The poor performance is highlighted and discussed. In the second set of experiments, we show how the performance of the deep-learning based visual trackers can be enhanced by training on our generated synthetic sequences. First, the evaluation measures are discussed in Section 5.1. Then, the utilized trackers are described in Section 5.2 and the training protocol is explained in Section 5.3. Finally, the results are analysed and explored in Section 5.4.

5.1. Evaluation Measures

The two widely used metrics *precision* and *success* (IoU) are employed for evaluating the performance of the visual trackers analyzed in this work. Precision calculates the distance between the centers of the tracker bounding box and the ground truth bounding box and then checks whether this center error is within the specified limits. We employ the conventional threshold of 20 pixels and consider the tracking as accurate for a frame if the center error is smaller than this value. We then extract the percentage of the accurately predicted bounding boxes for each sequence in our dataset. On the other hand, success measures the intersection over union (IoU) of the tracker and ground truth bounding boxes. We take a tracking to be successful if the IoU is larger than the common threshold of 0.50, and report the percentage of the successfully predicted bounding boxes averaged over the sequences in our dataset.

5.2. Trackers

In order to properly address the poor performance of the state-of-the-art general purpose trackers under adverse weather conditions, two different sets of trackers were selected. The sets present the two main approaches in visual object tracking, *i.e.* correlation filter -based and learning -based tracking.

Five different state-of-the-art correlation filter based trackers were chosen for the experiments. These are ECO [25], $BACF$ [26], and context aware (CA) [27] versions of DCF [28], $SAMF$ [29] and $STAPLE$ [30]. DCF, dual correlation filter, utilizes a kernelized correlation filter (KCF) that has a similar complexity to the linear counterpart of it, which improves tracker speed (FPS) considerably. On the other hand, $SAMF$, scale adaptive with multiple features, uses a scale adaptive template size instead of using a fixed one for the correlation filter kernel which is stated to make the tracker more robust. $STAPLE$, sum of template and pixel-wise learners, fuses template and histogram scores to better handle shape deformation which facilitates tracking deformable objects more accurately. ECO uses a modified version of DCF to improve memory usage, tracking speed, and robustness. $BACF$ uses a background-aware correlation filter that utilizes specific manually extracted features that account for both background and object of interest change over time. The context aware versions of DCF [28], $SAMF$ [29] and $STAPLE$ [30] that we used improve the original implementations by utilizing the global context information into the standard correlation filter tracking algorithms.

Similarly, for investigating the benefits of training on our generated synthetic sequences, four state-of-the-art learning based deep trackers were used. They are DiMP [31], ATOM [32], PrDiMP [33], and KYS [34]. DiMP is an offline learning based tracker that can be trained in an end-to-end manner. It applies both background and target information in the process of predicting the object of interest location. The tracker is based on the Siamese tracking architecture. It learns the discriminative loss function during the training phase. ATOM, however, is a deep-learning tracker that is trained both offline and online. Its tracking algorithm deploys target estimation and classification that are learnt offline and online respectively. At run-time, the classification component predicts the IoU between the target object and the estimated bounding box. PrDiMP is another learning based tracker that is based on the DiMP architecture. However, unlike DiMP tracker, PrDiMP applies probabilistic regression concept and predicts the probability density of the target given the input frame. This tracker is trained by minimizing KL-divergence in offline manner. KYS tracker, however, uses the visual scene information to better enhance the target localization and tracking. KYS encodes this information using localized state vectors and propagates it through the sequence to achieve better knowledge of the scene. Thus, it achieves better performance during testing. KYS is trained offline to learn how to propagate the scene information.

5.3. Training Protocol

We perform two training scenarios to assess the benefits of the generated synthetic sequences when used for training visual object trackers. For both experiments, the training was done using the whole PTAW217Synth dataset of 217 synthetic sequences. At the same time, the validation and testing were performed on the whole PTAW172Real dataset. For validation, 33 videos spanning the rainy, foggy and snowy weather conditions were selected at random. While the remaining 139 videos were applied for testing.

Training from Scratch. In the first scenario, we train the trackers from scratch using only the generated synthetic sequences. Then, the best model on the validation set is tested on the test set. The mean and the standard deviation of the tracker performances are reported for 5 iterations to account for the stochastic nature of these trackers. Both validation and test sets are real and contain no overlapping videos.

Fine-Tuning. In the second scenario, the pre-trained versions provided by the authors of the four trackers are fine-tuned on our synthetic sequences. Later, the performance of these models are stated as done in the previous case.

5.4. Results

The performance in terms of precision and success score are shown in Tables 3 and 4 for the studied trackers on the test partition of PTAW172Real, namely 163 videos. These results show that the trackers from both tracking mainstreams, correlation filter based and learning based, performed poorly under adverse weather conditions. This observation confirms that adverse weather conditions pose certain challenges for the state-of-the-art tracking

Table 3: Precision results of the available state-of-the-art trackers on the adverse weather condition real dataset, test partition of PTAW172Real.

Class	ECO	BACF	STAPLE_CA	SAMF_CA	DCF_CA	ATOM	DiMP	PrDiMP	KYS
Rain	0.59	0.50	0.46	0.38	0.22	0.61+/-0.01	0.60+/-0.01	0.61+/-0.01	0.63+/-0.02
Snow	0.56	0.53	0.49	0.46	0.35	0.60+/-0.01	0.62+/-0.01	0.59+/-0.01	0.58+/-0.01
Fog	0.67	0.65	0.59	0.42	0.37	0.73+/-0.01	0.74+/-0.01	0.74+/-0.01	0.77+/-0.02

Table 4: Success scores of the available state-of-the-art trackers on the real adverse weather condition dataset, PTAW172Real.

Class	ECO	BACF	STAPLE_CA	SAMF_CA	DCF_CA	ATOM	DiMP	PrDiMP	KYS
Rain	0.64	0.56	0.47	0.45	0.20	0.66+/-0.01	0.63+/-0.01	0.64+/-0.01	0.65+/-0.02
Snow	0.56	0.55	0.49	0.43	0.28	0.59+/-0.01	0.61+/-0.01	0.59+/-0.01	0.57+/-0.01
Fog	0.70	0.69	0.59	0.42	0.27	0.73+/-0.01	0.73+/-0.01	0.73+/-0.01	0.78+/-0.02

algorithms. The correlation filter trackers perform worse than the deep trackers because they are mostly online learning trackers. On the other hand, the deep trackers, which are based on offline learning algorithms, were trained on large scale datasets, which may have contained a number of videos under adverse weather conditions. Thus, they performed slightly better than the correlation ones.

It seems that rain and snow particles, that partially occlude the object of interest, cause a significant change on the visual characteristics of the trackers. Thus, it makes it hard for the tracker to differentiate the target object from the background. This effect is particularly clear when the size of the object of interest is relatively small. In parallel to that, fog causes both the background and the object of interest regions to have similar visual appearance. Thus, it makes it hard for the tracker to distinguish the target object from the background. Even so, foggy weather condition seems to be slightly less challenging as compared to the others.

The results of our training experiments are shown in Fig. 9. The IoU scores for the four trained trackers, namely DiMP, ATOM and PrDiMP, are presented for the two training scenarios. Moreover, these results are compared to the ones of their corresponding baselines. Both average and standard deviation on five iterations were reported to account for the stochastic nature of these trackers. Training these trackers from scratch on our adverse weather synthetic sequences achieves comparable results to the ones obtained using the baseline for DiMP and PrDiMP. For ATOM and KYS, however, the trained models from scratch surpassed their baselines. On the other hand, fine-tuning the pre-trained models on our synthetic sequences improved the performance of the three trackers ATOM, DiMP and PrDiMP distinctly.

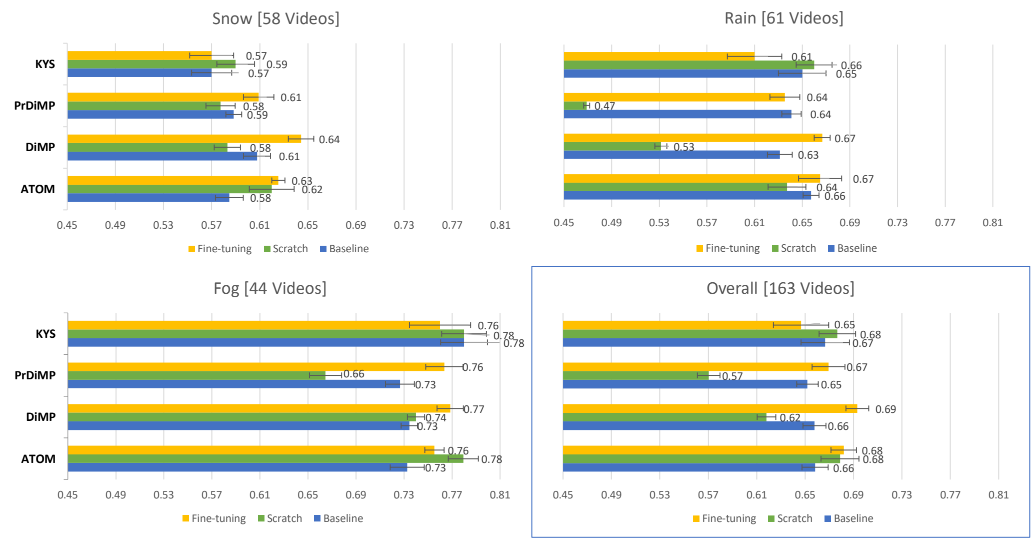

Figure 9: IoU results obtained with the two different training scenarios as compared to those of the baselines. Error bars give the standard deviation of the IoU results. Fine tuning the baselines on our synthetic sequences improves the performance.

It is worth noting, that both the tracking algorithm and the training dataset affect how a specific tracker gains from training on our synthetic sequences. Both determine which training scenario, from scratch or fine-tuning, is more beneficial. For example, DiMP and PrDiMP trackers got the most advantage from fine-tuning. On the other hand, training from scratch was better for KYS tracker, while the performance of ATOM was improved in both scenarios. Another point to be noticed is the conspicuous difference in the level of improvement in trackers performance across different weather conditions. This can be directly linked to the varying distribution of the adverse weather conditions in the different training datasets used for these baselines. So much so that, the lack of adverse weather conditions videos in the training dataset stands out to be the main reason behind the observed performance boost since using even a relatively small number of synthetic sequences spanning these absent features helped the trackers to outperform their baselines, given that the trackers were originally trained on large scale datasets such as LaSOT [35], GOT10k [36], COCO [37], and TrackingNet [38], each far exceeding PTAW217Synth in number of sequences.

It is important to note that test set contains only real sequences. Thus, the domain gap problem is not a playing factor under the scope of this analysis. In contrast, diversity of the synthetic sequences in terms of weather conditions, times of day, lighting conditions, camera attributes and synthetic humans altogether enhanced the training process significantly. Additionally, the high level of photorealism of these synthetic sequences mitigated the gap across the real and synthetic domains. Thus, training from scratch or fine-tuning on our synthetic sequences directly improved the trackers performance.

A qualitative comparison among the tracking results achieved by the baselines and the trained models is presented in Fig. 10. It is seen that utilizing our synthetic data for training improves the performance of the baselines under adverse weather conditions.

Additionally, Fig. 11 displays the success scores for the four deep trackers under full occlusion, scale change, background clutter and sudden camera motion videos. In general, both the baselines and the trained models performed the worst in sequences with background clutter while the ones with sudden camera motion resulted in relatively higher performance. It could be because the background clutter under adverse weather conditions causes the trackers to experience a significant difficulty in locating the object of interest since both have similar visual appearances. On the other hand, the reason that abrupt camera motion does not seem to be effecting trackers as much as the other attributes could be due to the fact that the other three attributes are more closely associated with the object of interest as compared to the camera motion which effect both background and target similarly. A table showing the number of sequences in each weather condition for each of the four attributes is provided in the supplementary material.

6. Conclusion

Our work investigated the lack of adverse weather conditions in the available general purpose visual tracking datasets and highlighted the low performance of the state-of-art trackers under these specific circumstances. As a solution, we proposed using our NOVA rendering engine to generate synthetic sequences that span snowy, rainy and foggy weather conditions. We trained four different deep trackers, namely DiMP, ATOM, KYS and PrDiMP, on 217 synthetic sequences generated by NOVA and tested them on the real videos that were collected from YouTube and annotated meanly by us for that aim. Our analysis reveals that applying our synthetic sequences for

Figure 10: A qualitative comparison of our trained trackers with the baselines on three example sequences. Training on PTAW217Synth improves the trackers performance under adverse weather conditions.

training purposes can bridge the data gap and improve the trackers performance considerably.

A number of limitations have come to light toward the goal of using synthetic sequences for model training as an alternative to the real data. Perhaps the domain gap problem is the one of central concern in this scope. It arises

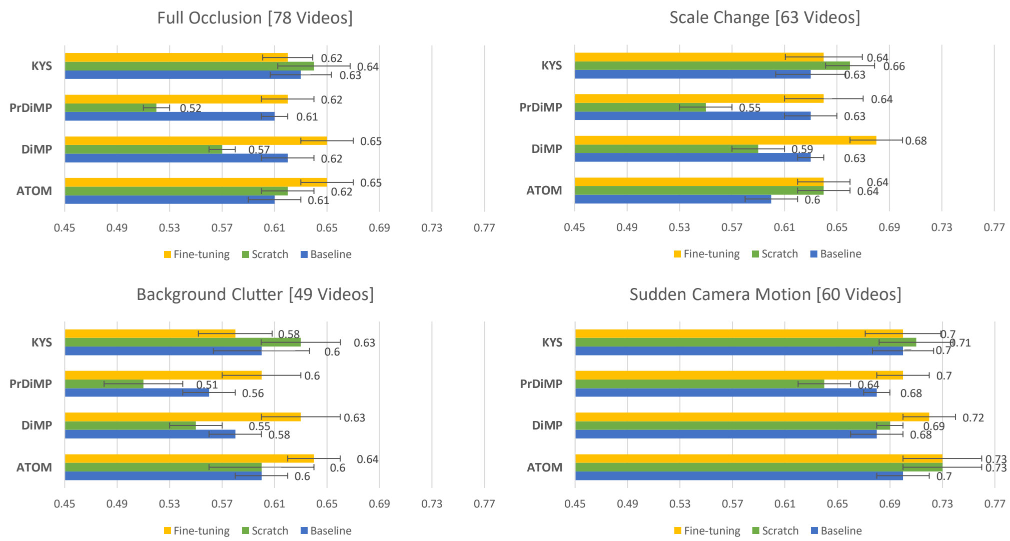

Figure 11: Success scores for ATOM, DiMP, PrDiMP and KYS trackers are shown for four different attributes. Background clutter causes the trackers to perform poorly.

mainly because the training and testing processes take place in two different domains i.e. synthetic and real domains, respectively. To address this point, we paid great attention to the photorealism of the generated synthetic sequences and most specifically the simulated adverse weather conditions. The second key issue is that synthetic sequences are usually generated at optimal lighting and recording conditions. Thus, the lack of image artifacts such as motion blur, chromatic aberration, noise and others may cause models trained on it to fail once such artifacts are encountered in real sequences. To mitigate this problem, we generate our synthetic sequences at different lighting conditions and recording setups. Additionally, we simulate lens artifacts such as motion blur and chromatic aberration. Another note-worthy issue is the fact that repetitive textures, objects, animations, and motions frequently observed in virtual 3D worlds may cause over-fitting. We tackled this issue by diversifying scene elements such as pedestrians, buildings, cars, and other scene objects.

Throughout this work, we demonstrated how our generated synthetic sequences improved trackers performance on adverse weather conditions. However, investigating the effect of adverse weather conditions on other computer vision tasks like optical flow estimation, depth estimation, and person re-identification are sill open questions. The boost in performance upon remedying the lack of sample with adverse weather conditions for the VOT task could be an indication of a similar problem in other computer vision tasks. In the light of this study, we believe that using our rendering engine NOVA to generate synthetic training data can bridge the gap of data scarcity in said tasks toward improvement in both accuracy and robustness.

The datasets PTAW172Real and PTAW217Synth that we featured in this work are available for download at the project website `https://graphics.cs.hacettepe.edu.tr/NOVA-Adverse` along with a supporting video illustrating the motivation behind this work, a sample of sequences from PTAW217Synth and also a sample of the PTAW172Real sequences superimposed with tracking results.

Acknowledgements

This work was supported in part by TUBITAK-1001 Program (Grant No. 217E029), GEBIP 2018 fellowship of Turkish Academy of Sciences awarded to E. Erdem, and BAGEP 2021 Award of the Science Academy awarded to A. Erdem.

References

[1] Z. Cai, N. Vasconcelos, Cascade r-cnn: Delving into high quality object detection, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 6154–6162.

[2] Q. Chu, W. Ouyang, H. Li, X. Wang, B. Liu, N. Yu, Online multi-object tracking using cnn-based single object tracker with spatial-temporal attention mechanism, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 4836–4845.

[3] C. Hazirbas, L. Ma, C. Domokos, D. Cremers, Fusenet: Incorporating depth into semantic segmentation via fusion-based cnn architecture, in: Asian conference on computer vision, Springer, 2016, pp. 213–228.

[4] K. Tateno, F. Tombari, I. Laina, N. Navab, Cnn-slam: Real-time dense monocular slam with learned depth prediction, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 6243–6252.

[5] J. Walker, A. Gupta, M. Hebert, Dense optical flow prediction from a static image, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 2443–2451.

[6] J. K. Kang, T. M. Hoang, K. R. Park, Person re-identification between visible and thermal camera images based on deep residual cnn using single input, IEEE Access 7 (2019) 57972–57984.

[7] H. A. Alhaija, S. K. Mustikovela, L. Mescheder, A. Geiger, C. Rother, Augmented reality meets computer vision: Efficient data generation for urban driving scenes, International Journal of Computer Vision 126 (9) (2018) 961–972.

[8] E. Cheung, T. K. Wong, A. Bera, X. Wang, D. Manocha, Lcrowdv: Generating labeled videos for simulation-based crowd behavior learning, in: European Conference on Computer Vision, Springer, 2016, pp. 709–727.

[9] G. Varol, J. Romero, X. Martin, N. Mahmood, M. J. Black, I. Laptev, C. Schmid, Learning from synthetic humans, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 109–117.

[10] I. B. Barbosa, M. Cristani, B. Caputo, A. Rognhaugen, T. Theoharis, Looking beyond appearances: Synthetic training data for deep cnns in re-identification, Computer Vision and Image Understanding 167 (2018) 50–62.

[11] A. Gaidon, Q. Wang, Y. Cabon, E. Vig, Virtual worlds as proxy for multi-object tracking analysis, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 4340–4349.

[12] L. Zhang, A. Gonzalez-Garcia, J. van de Weijer, M. Danelljan, F. S. Khan, Synthetic data generation for end-to-end thermal infrared tracking, IEEE Transactions on Image Processing 28 (4) (2018) 1837–1850.

[13] A. Milan, L. Leal-Taixé, I. Reid, S. Roth, K. Schindler, Mot16: A benchmark for multi-object tracking, arXiv preprint arXiv:1603.00831 (2016).

[14] A. Li, M. Lin, Y. Wu, M. Yang, S. Yan, NUS-PRO: A New Visual Tracking Challenge, IEEE Transactions on Pattern Analysis and Machine Intelligence 38 (2) (2016) 335–349.

[15] Y. Wu, J. Lim, M. Yang, Object tracking benchmark, IEEE Transactions on Pattern Analysis and Machine Intelligence 37 (9) (2015) 1834–1848. doi:10.1109/TPAMI.2014.2388226.

[16] P. Liang, E. Blasch, H. Ling, Encoding color information for visual tracking: Algorithms and benchmark, IEEE Transactions on Image Processing 24 (12) (2015) 5630–5644.

[17] S. R. Richter, Z. Hayder, V. Koltun, Playing for benchmarks, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2213–2222.

[18] C. R. De Souza, A. Gaidon, Y. Cabon, A. M. L. Peña, Procedural generation of videos to train deep action recognition networks., in: CVPR, 2017, pp. 2594–2604.

[19] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, A. M. Lopez, The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 3234–3243.

[20] A. Kerim, C. Aslan, U. Celikcan, E. Erdem, A. Erdem, Nova: Rendering virtual worlds with humans for computer vision tasks, Computer Graphics Forum, in press (2021).

[21] Microsoft, Microsoft rocketbox avatar library git repo, https://github.com/microsoft/Microsoft-Rocketbox, online; accessed: 2020-05-17.

[22] P. Worlds, Enviro webpage, online; accessed: 2019-02-20.
URL http://www.procedural-worlds.com/gaia/gaia-extensions/enviro/

[23] A. Dutta, A. Zisserman, The VIA annotation software for images, audio and video, in: Proceedings of the 27th ACM International Conference on Multimedia, MM '19, ACM, New York, NY, USA, 2019. doi:10.

1145/3343031.3350535.
URL https://doi.org/10.1145/3343031.3350535

[24] A. Dutta, A. Gupta, A. Zissermann, VGG image annotator (VIA), http://www.robots.ox.ac.uk/ vgg/software/via/, version: 2.0.10, Accessed: 2020-07-19 (2016).

[25] M. Danelljan, G. Bhat, F. Shahbaz Khan, M. Felsberg, Eco: Efficient convolution operators for tracking, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 6638–6646.

[26] H. Kiani Galoogahi, A. Fagg, S. Lucey, Learning background-aware correlation filters for visual tracking, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 1135–1143.

[27] M. Mueller, N. Smith, B. Ghanem, Context-aware correlation filter tracking, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[28] J. F. Henriques, R. Caseiro, P. Martins, J. Batista, High-speed tracking with kernelized correlation filters, IEEE transactions on pattern analysis and machine intelligence 37 (3) (2015) 583–596.

[29] Y. Li, J. Zhu, A scale adaptive kernel correlation filter tracker with feature integration, in: European conference on computer vision, Springer, 2014, pp. 254–265.

[30] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, P. H. Torr, Staple: Complementary learners for real-time tracking, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 1401–1409.

[31] G. Bhat, M. Danelljan, L. V. Gool, R. Timofte, Learning discriminative model prediction for tracking, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 6182–6191.

[32] M. Danelljan, G. Bhat, F. S. Khan, M. Felsberg, Atom: Accurate tracking by overlap maximization, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 4660–4669.

[33] M. Danelljan, L. V. Gool, R. Timofte, Probabilistic regression for visual tracking, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 7183–7192.

[34] G. Bhat, M. Danelljan, L. Van Gool, R. Timofte, Know your surroundings: Exploiting scene information for object tracking, arXiv preprint arXiv:2003.11014 (2020).

[35] H. Fan, L. Lin, F. Yang, P. Chu, G. Deng, S. Yu, H. Bai, Y. Xu, C. Liao, H. Ling, Lasot: A high-quality benchmark for large-scale single object tracking, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 5374–5383.

[36] L. Huang, X. Zhao, K. Huang, Got-10k: A large high-diversity benchmark for generic object tracking in the wild, IEEE Transactions on Pattern Analysis and Machine Intelligence (2019).

[37] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C. L. Zitnick, Microsoft coco: Common objects in context, in: European conference on computer vision, Springer, 2014, pp. 740–755.

[38] M. Muller, A. Bibi, S. Giancola, S. Alsubaihi, B. Ghanem, Trackingnet: A large-scale dataset and benchmark for object tracking in the wild, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 300–317.