Robust Multi-Object Tracking Using Re-Identification Features and Graph Convolutional Networks

Christian Lusardi, Abu Md Niamul Taufique, Andreas Savakis
Rochester Institute of Technology
Rochester, NY, 14623
{cml8292, at7133, andreas.savakis}@rit.edu

Abstract

We propose a graph neural network-based framework for multi-object tracking that combines detection and association along with the use of a novel re-identification feature. We explore the combination of multiple appearance features within our framework to obtain a better representation and improve tracking accuracy. Data augmentations with random erase and random noise are utilized to improve robustness during tracking. We consider various types of losses during training, including a unique application of the triplet loss to improve overall network performance. Results are presented on the UAVDT benchmark dataset for aerial-based vehicle tracking under various conditions.

1. Introduction

Multi-Object Tracking (MOT) is an important task in computer vision with applications in surveillance [2, 25, 36], autonomous navigation [26, 40], and traffic monitoring [46, 23]. However, challenges arise when keeping track of multiple objects at once, as objects experience heavy occlusion or their paths merge with one another [48]. Models also struggle when their bounding box localization decreases, as the subjects fall out of alignment with detection anchor boxes [42]. Even in cases where the detector is accurate, poor re-identification (ReID) can cause problems if the tracking takes place under occlusion or out-of-frame conditions [46]. In situations such as these, it is important to have a strong object association network that can properly re-identify and correctly match the tracked objects under these complex scenarios.

Many MOT trackers do not have dedicated object detectors to utilize during tracking [23, 16, 42, 22], and the detectors used are frequently anchor based. When tracking from the air, such detectors provide relatively accurate bounding box locations at lower altitudes, but their performance diminishes at higher altitudes when the tracked objects are small and fall between the anchor box overlap locations [28, 27]. The lack of accurate bounding boxes in combination with small object size means that the appearance feature extractors may sample more of the background and less of the object, leading to ambiguous features and more confusion during tracking. The problem can become much more difficult for single stage networks which attempt to build a ReID feature from anchor based detectors, as mentioned in Zhang et al. [46]. These networks could end up with poor MOT performance, because the bounding boxes may sample less useful information as a result of the anchor box offset [39].

Graph Neural Network (GNN) based trackers are becoming more popular in MOT settings due to their ability to learn association patterns [23, 16, 18, 31, 19]. Graph based networks utilize the spatial relationships between objects along with appearance features to describe spatial relationships. The physical relationships between objects are important in MOT settings because they allow the model to keep track of surrounding objects by forming sub-graphs and sharing information between them [18]. A typical drawback with many graph trackers is that they have poor re-identification (ReID) networks that are based on pre-trained classifier networks. These detectors are trained to separate a known object class from the background, but they are not designed to accurately separate instances of the same class. CSTrack [17], while not being a graph based network, demonstrated how appearance features extracted from generalized backbones, e.g. ResNet [8], can cause issues during re-identification because their features are built to detect objects in a particular class, but may not be able to distinguish between two objects of the same class. Detector performance also diminishes under occlusion conditions, or even illumination, rotation, and viewpoint changes that are commonly seen in MOT datasets [7]. To overcome these issues, we utilize LABNet[34], a graph-based re-identification network that can be useful for MOT, as it is designed to identify an object among multiple instances of the same class.

In this paper, we introduce Graph Tracking With

ReID (GTREID), an MOT tracking framework with re-identification that is built on an anchorless detection network. The detector is trained with data augmentations for greater generalization ability along with a class-based triplet loss to improve upon ID switches. The bounding boxes are then utilized to locate and extract appearance features using an object centerpoint feature [46] as well as a novel ReID feature [34]. All location and appearance features are then sent through a graph neural network to perform a similarity comparison before final assignments are made. The main contributions of this work are as follows:

- We present an end-to-end trainable MOT framework based on graph neural networks that achieves state-of-the-art results on the UAVDT dataset.
- Our novel approach includes a dedicated re-identification network in conjunction with a centerpoint appearance feature to strengthen the overall association capabilities of our MOT tracker.
- We incorporate a class-based triplet loss during training to improve the discriminating capabilities the centerpoint appearance feature.

2. Related Work

2.1. Single Object Tracking

Single object trackers based on Siamese architectures [4] have gained popularity due to their effective balance of computational speed and tracking performance. Over the years, Siamese trackers have increased their capabilities through better detections [15], better training strategies [50], as well as better feature extraction techniques [14]. For aerial tracking, SiamReID [35] recently utilized a dedicated ReID appearance feature for reacquiring the tracked object after occlusion when local distractors were present.

2.2. Multi-Object Trackers

One of the most successful single stage MOT methods is the FairMOT network [46] which tackled the issue of accuracy by using a state-of-the-art object detector. FairMOT includes CenterNet, an anchorless object detection network [49] that is more precise than the anchor based detectors utilized by most tracking by detection methods [39, 17]. This detector model is designed to detect objects using center point regression of the object. The absence of region proposals eliminates the need to perform non-maximal suppression, as each object is only given a single detection hypothesis. The CenterNet architecture [49] has no manual thresholds within the network, with all parameters learned during training. This results in much stronger object localization and placement of bounding boxes.

FairMOT's appearance feature is built using a two layer convolutional network on top of the DLA-34 [43] backbone network. The appearance feature is located at the peak location of the heatmap and the model creates a single ReID feature that is located at the centerpoint of the object.

CSTrack demonstrated how utilizing an appearance feature that is designed for intra-class separability can be beneficial. [17]. Wang et al. demonstrated the effectiveness of adding a graph neural network to the backbone of FairMOT in order to reduce the identity switches by providing a spatial relationship between objects in an image [38]. Xiang et al. used a Markov decision process to predict and track objects utilizing a reinforcement based approach to object tracking [42]. During training, they focus on hard example mining by only updating their model when it makes a mistake. The idea of hard example mining is also seen in FairMOT's focal loss. CorrTracker [37] utilizes local spatial context information during testing in order to reduce identity switches by sampling the local area around a target track to reduce the distractors that may prevent the network from continuing on a successful track.

2.3. Aerial Tracking

Aerial based object tracking is a challenging task, as the tracked objects are small in pixel size and densely compacted [33]. The problem is made more difficult due to the large changes in scale, weather conditions, background clutter, and irregular camera movement [1]. The low resolution of the objects makes it difficult to differentiate between the inter- and intra-class examples [11]. To reduce the number of identity switches on small objects, Jadhav et al. utilize a denser set of anchor scales on their RetinaNet based tracking network. They also utilize Squeeze-and-Excitation [9] blocks to increase the tracker's ability to deal with camera motion and small scale features. Azimi et al. implement an aerial tracker with an LSTM to model the motion changes between frames and a graph neural network to cluster the large number of similar looking objects in an image [1].

2.4. Object Detection

Accurate object detector can greatly improve the performance of multi-object tracking networks, especially in densely populated scenarios. Models have been able to overcome some of the issues with anchor-based bounding boxes by utilizing a stronger ReID network built on top of a single shot detector [17, 21]. The CSTrack method is able to achieve higher Multi-Object Tracking Accuracy (MOTA) scores and lower ID switching because of their use of a stronger appearance feature. However, their object tracking precision is still slightly lower because their method is limited by the accuracy of their incoming bounding boxes.

CenterTrack [48] is an example of a model that is built using the anchorless CenterNet [49]. This method tracks objects exclusively through the motion of the bounding

boxes without the use of any appearance models. Their network still achieves high MOTA scores because of the strength of the detection bounding boxes and optical flow [24] based motion predictions, but it faces problems during instances of occlusion and missed tracks because it has no access to the appearance information of the tracked objects.

Zhang et al.[46] utilized the anchorless backbone and combined it with a single stage appearance feature to create a robust multi object tracking network [39, 17]. However, their appearance features are mainly based on a generalized feature extractor which lacks some of the discriminative capabilities needed to track similarly looking vehicles that may be of the same general size and color.

2.5. Graph Neural Networks

Graph networks for MOT typically use the appearance features of each object as the nodes of the graphs [23, 16, 18, 31], but they tend to differ on how they build the edges between nodes. GCNNMatch by Papakis et al. builds graph edges using a concatenation of the bounding box overlaps and feature appearance to create what they call an *interaction feature*. This feature is then used in a cosine similarity function that feeds into a Sinkhorn normalization algorithm [32] before performing a Hungarian algorithm assignment [13] for tracking.

Other approaches [16] utilize two separate graph models, a model for the appearance features and another for the motion features. Their motion features are obtained from the bounding box coordinates and the distance between the boxes is used for their graph edges. A similar architecture is used in their appearance model, except that the feature vectors are the nodes and the feature distances are the edges. The output of each model is then combined before being fed into a Hungarian algorithm for final assignment. This architecture suffers from the limitation of keeping the motion and appearance learning separated, which defeats some of the benefits of multiple modality tracking. Their matching algorithm relies on the overlap of separate assignments rather than using both location and appearance together in a single network.

The novelty of GCNNMatch [23] comes from the use of the Sinkhorn algorithm during training and testing. The Sinkhorn algorithm helps to solve the problem of optimal transport between a set of known tracks and the new set of detections. It is not a hard assignment algorithm, meaning it does not make final assignments between each pairing, but it is commonly used to solve graph matching problems [29]. Papakis et al. [23] use it within their objective function that feeds into their binary cross-entropy loss.

Other models [18] utilize extra conditionals to determine whether a connection between nodes is made. In their layout, a spatio-temporal condition has to be met in order to build a neighbor graph among the set of detections. The objects in question must appear in the same frame and be within a certain Euclidean distance of each other in order to have a connection formed. This concept, while strict and accurate in ideal cases, causes problems if the object detector does not detect all objects in the same frame. This means that their information would not be passed amongst the nodes and could lead to greater ID switching or lost tracks.

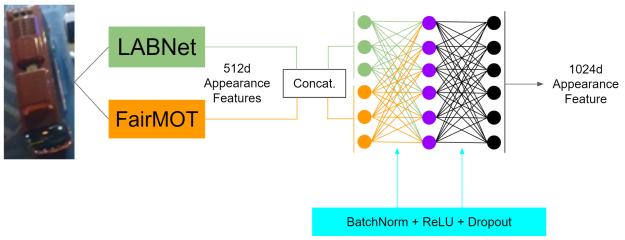

Figure 1. Proposed appearance feature combination model. The appearance features consist of the combination of the FairMOT based ReID head as well as the LABNet feature vector. The detections and bounding boxes are formed using the FairMOT/CenterNet predictions.

3. Methodology

The proposed GTREID approach extends the methodology from GCNNMatch by the usage of two appearance features (Figure 1) for a better overall representation of the object appearance. Our GTREID architecture is detailed in Figure 2. The network combines all tracking steps from detection through object association into a single model. GTREID incorporates its own detector based on FairMOT. This is used in order to improve the bounding box localization as well as providing a ReID feature from the centerpoint of the of the object. The bounding boxes of the CenterNet based architecture are also beneficial for extracting ReID features from the LABNet model [34]. The LABNet features are designed to improve the intra class differentiation between similar objects for reaquisition after occlusion.

3.1. Re-Identification

The ReID network in GTREID (Figure 1) consists of two fully connected layers which combine the appearance features from LABNet and the FairMOT centerpoint network. The FairMOT network can struggle when the centerpoint of the object in question is similar to others in a scene. For example, the roof of multiple black cars appear very similar in many situations, causing the features to be very similar. In such cases, the LABNet model can provide more distinguishing features that would reduce the likelihood of an identity switch. To aid in the balance between the two appearance features, the fully connected layers are implemented with Batch Normalization [10] and Dropout layers,

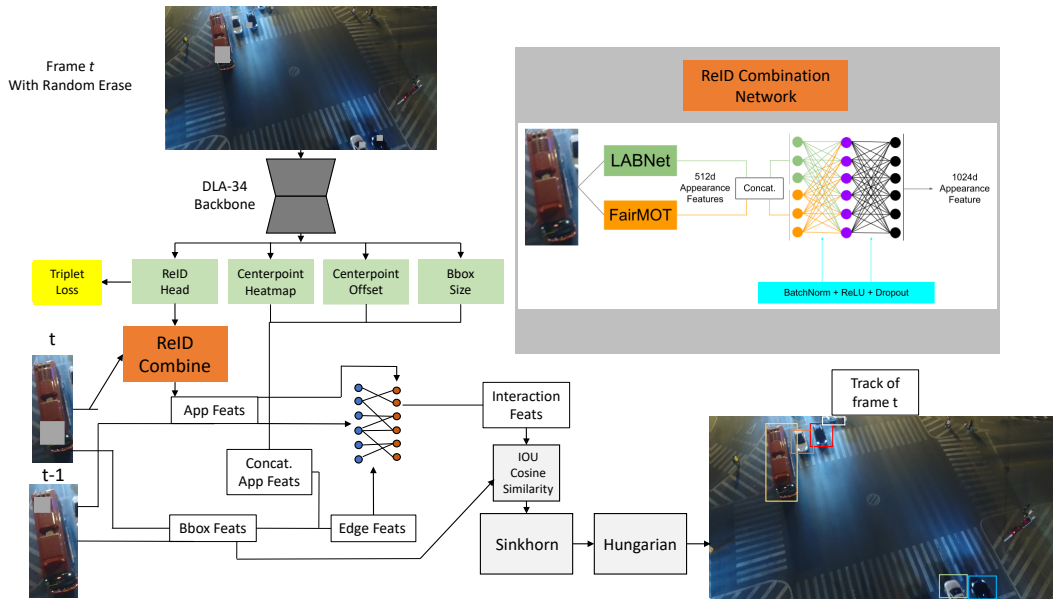

Figure 2. Proposed graph Model for MOT, named GTREID. The appearance features consist of the combination of the FairMOT based ReID head as well as the LABNet feature vector. The detections and bounding boxes are formed using the FairMOT/CenterNet predictions.

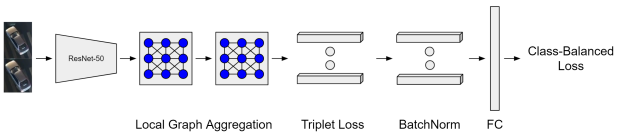

Figure 3. LABNet Re-Identification Network [34] which is used as the second half of the appearance feature combination network detailed in Figure 1.

so that the network learns to take both types of features into consideration.

Taufique's [34] LABNet ReID network (Figure 3) is used to obtain robust appearance feature vectors for vehicle ReID. LABNet utilizes a class-balanced loss [6] during training, so that the underrepresented objects in the training dataset are not unfairly discounted in favor of the more prevalent identities. It also implements a triplet loss [30] when training to increase the differentiating abilities of the network. LABNet features a message passing GCN network which evenly splits each image into a 20×20 grid and passes aggregated image information among all of the regions to improve vehicle re-identification, even if it undergoes a large rotation or scale change after occlusion since the last time that it was viewed.

3.2. Centerpoint Detector

Our detector is motivated by FairMOT's anchorless object detector with centerpoint ReID features [46]. The network has been trained with an additional triplet loss parameter applied to the output of the classifier. This is used to strengthen the centerpoint ReID feature and reduce the number of identity switches. The detector network serves a second purpose of defining accurate bounding boxes which are used by the secondary ReID network as well as the graph network for associations. The advantage of using the CenterNet [49] based detector for aerial applications is that it works effectively at low altitude as well as high altitude. The lack thresholds or predefined box parameters makes the model more suitable to challenging aerial scenery where the objects undergo large changes in scale as well as camera shifts within a single sequence.

3.3. Graph Association

After the detection and ReID feature extraction, the bounding boxes and Re-Identification features are passed into a network for graph convolution based association [23]. Here, the physical relationship between the objects in the past and present is used to make more accurate matching across frames. The spatial relationship between objects is crucial to reducing the number of identity switches during tracking. As seen in the baseline model, the graph network utilizes the Sinkhorn Normalization method to make soft assignments during training and testing. This outputs similarity scores ranging from 0 to 1 to determine whether a detection is a good match to a known track.

3.4. Data Augmentation

Random Erase [47] and Random Noise have demonstrated significant performance increases in detection per-

formance under occlusion conditions. Random Erase is designed to prevent a detector from overfitting to the training data, thus ruining the ability to generalize to situations where the object may go under partial occlusion. This is especially common in MOT datasets where vehicles may fully disappear behind a street sign or trees along the side of the road. Having a robust detector helps to reduce the potential for false negatives in the resulting evaluation.

Random Noise has been implemented to help deal with situations in which lighting conditions may be different than those experienced during training. Darker scenes are present in the UAVDT dataset and as such, the noise increase in the camera is more prevalent. This can lead to not only inaccurate detections but also poor ReID features from the ReID head. To combat this, adding noise into the training data allows the system to be more accurate during testing.

3.5. Training Loss Functions

There are multiple loss functions involved in the training of GTREID. The CenterNet-based detector relies on the training of four separate heads shown in Figure 2, including a centerpoint heatmap, offset, bounding box size, and feature extractor. The overall loss for the detector and centerpoint appearance feature is:

$$L_{total} = \frac{1}{2}(\frac{1}{e^{w_1}}L_{det}+\frac{1}{e^{w_2}}L_{identity}+L_{tri}+w_1+w_2) \quad (1)$$

where

$$L_{det} = L_{heat} + \lambda_{size}L_{size} + \lambda_{offset}L_{offset}$$

and w_1 and w_2 are learnable parameters during training and λ_{size} and λ_{offset} are hyper parameters set to 0.1 and 1, respectively. [46].

The heatmap is designed to highlight the central location of each object in an image. The detector takes the element-wise peak location of each Gaussian response as this central location. The heatmap head is trained using a pixel-wise logistic regression with focal loss.

$$L_{heat} = -\frac{1}{N}\sum_{xyc}\begin{cases}(1-\hat{Y}_{xyc})^{\alpha}log(\hat{Y}_{xyc}) & \text{if } Y_{xyc}=1\\ (1-Y_{xyc})^{\beta}(Y_{xyc})^{\alpha}log(1-Y_{xyc}) & \text{otherwise}\end{cases} \quad (2)$$

where Y_{xyc} is the center location of the Gaussian kernel given as

$$Y_{xyc} = exp\left(-\frac{(x-\tilde{p}_x)^2+(y-\tilde{p}_y)^2}{2\sigma_p^2})\right)$$

and $\tilde{p}$ is the keypoint location divided by the output prediction factor, α and β are the hyper-parameters of the focal loss, N is the number of keypoints in the particular image, and σ_p represents the standard deviation based on the object size.

Given that the output feature map is of a lower resolution than the original image, an offset had to be learned in order to relate the feature map back to the original image locations. The offset is trained with a simple L_1 loss between the predicted point and the ground truth point

$$L_{offset} = \frac{1}{N}\sum_{p}\left|\hat{O}_{\tilde{p}} - (\frac{p}{R}-\tilde{p})\right|, \quad (3)$$

where R is the output stride of the network and $O_{\tilde{p}}$ is the ground truth offset. The R value is set by default to 4 to be consistent with the baseline [46].

The network also learns a bounding box size parameter which is able to build a suitably sized bounding box around a given heatmap peak location. The size prediction head is also trained using an L_1 loss at the center location.

$$L_{size} = \frac{1}{N}\sum_{k=1}^{N}\left|\hat{S}_{pk} - s_k\right| \quad (4)$$

where $\hat{S}_{pk}$ represents the ground truth size. The centerpoint p_k is:

$$p_k = \left(\frac{x_1^k+x_2^k}{2}, \frac{y_1^k+y_2^k}{2}\right)$$

and the object size s_k is

$$s_k = (x_2^k - x_1^k, y_2^k - y_1^k)$$

In order to train the ReID head of the network, the peak heatmap location is obtained and the class distribution is learned using the cross-entropy loss in Equation (5).

$$L_{identity} = -\sum_{i=1}^{N}\sum_{k=1}^{K}L^i(k)log(p(k)). \quad (5)$$

where K is the number of classes within the dataset, and $L^i(k)$ is the one-hot representation of the ground truth class labels.

The triplet loss used in GTREID is designed to increase the separability between the classification outputs of the backbone ReID network. The triplet loss [30] is defined as

$$L_{tri} = \|f(x_i^a) - f(x_i^p)\|_2^2 + \alpha < \|f(x_i^a) - f(x_i^n)\|_2^2. \quad (6)$$

where x_i^a is an anchor classification, x_i^p is a positive classification, and x_i^n is a negative classification and $f()$ represents the classifier prediction for the centerpoint ReID model. α represents the margin that is enforced between the positive and negative images in order to prevent the model from collapsing. A weighted binary cross-entropy loss is used to

train the graph network within GTREID. The goal of the graph network is to assign a tracklet-detection pair with a similarity score of 1 and a non-matching pair with a similarity score of 0. Given the inherently imbalanced nature of graph data, there will be a far greater number of mismatched pairs in comparison to the matching pairs. Therefore, the matches need to be weighted higher to avoid the possibility of the model settling and constantly choosing a non-match. GTREID uses the same weighted binary cross-entropy loss as GCNNMatch: [23]

$$L_{wb} = -w_0 * (y * log(x)) - w_1 * ((1-y) * log(1-x)) \quad (7)$$

where w_o and w_1 are predefined weights that are set to (10,1) to be consistent with GCNNMatch.

4. Dataset and Experiments

4.1. UAVDT Dataset

The Unmanned Aerial Vehicle Benchmark: Object Detection and Tracking (UAVDT) [7] is one of the only aerial based MOT benchmarks that is shot from three altitude levels at varying times of day with diverse weather. The benchmark is shot at altitudes ranging from low-altitude (10-30 meters), medium-altitude (30-70 meters), and higher-altitude (greater than 70 meters). The dataset also features three kinds of camera view points on objects: front-view, side-view, and bird-view. The final attribute within the dataset is the weather condition, which ranges from daylight, night, and fog. The training and testing sequences can have any combination of these attributes which makes tracking especially challenging.

The dataset used to pretrain the LABNet [34] network for vehicle re-identification was the VeRi776 dataset [20]. This dataset contains 50,000 images with 776 unique vehicles which are imaged using 20 separate cameras. The cameras are spread out across a 1.0 km^2 area and all images were taken within 24 hours.

4.2. Evaluation

The primary metrics that are used to judge the success of a Multi-Object Tracker are the CLEAR Metrics [3] which include the following. MOT Accuracy (MOTA):

$$MOTA = 1 - \frac{\sum_t (m_t + fp_t + mme_t)}{\sum_t g_t}, \quad (8)$$

where m_t represents the number of missed targets, fp_t represents the false positives, and mme_t represents the number of identity switches over the number of ground truth objects (g_t) in the image. MOT Precision (MOTP):

$$MOTP = \frac{\sum_{i,t} d_t^i}{\sum_t c_t}, \quad (9)$$

where d_t^i represents the distance between the tracklet prediction and the ground truth object, while c_t represents the number of total objects in the image. Other metrics are Mostly Tracked (MT), Mostly Lost (ML), False Positive (FP), False Negatives (FN), ID Switching (IDs), MOTA is a judgement of all of the failure cases of the system. It outlines all of the times the tracker has a false positive, false negative, or identity switch per frame. MOTP is a measurement of how accurately the tracker's bounding boxes align with the ground truth bounding boxes per frame.

4.3. Implementation Details

In ablation studies with the graph backbone, the network converges within 7 epochs of training on UAVDT, whereas the detector network converges at 30 epochs. Therefore the GCN training was paused until the 24th training epoch where it was then added in to begin learning. Another reason for waiting was to avoid having weak features enter the appearance feature combination network. This could cause the linear layers to converge too quickly to the stronger LABNet features, defeating the purpose of the combined models.

The Random Erase and Random Noise augmentations were implemented from epochs 3 to 13 with Random Erase performed 40% of the time with a scale ranging from 0.02-0.25 percent of the bounding box and an aspect ratio of 0.2, 0.5. Random Noise was performed on 30% of images with a mean and standard deviation of 2. The learning rate for the backbone network started at 1^{-4}, then dropped to 1^{-5} at epoch 20 and remained at that value until the end of training. As for the Graph Network training, the learning rate was set to 1^{-3} for epochs 24 thru 26, then dropped to 1^{-4} for the remainder of the training. The Adam Optimizer [12] was used to train all portions of the final network. Separate optimizers were used for the backbone as well as the GCN model.

5. Results

Method	IDF$_1$ (%)	MOTA (%)	MOTP (%)	MT (%)	ML (%)	IDs
MDP [42]	61.5	43.0	73.5	45	22.7	541
DSORT [41]	58.2	40.7	73.2	41.7	23.7	2,06
IPGAT [44]	49.4	39.0	72.2	37.4	25.2	2,09
SBMA [45]	48.5	38.6	72.1	38.9	24.4	3,49
IOUT [5]	23.7	36.6	72.1	37.4	25.0	9,94
FairMOT	**68.03**	49.63	78.0	49.96	17.69	512
GCNNMatch	49.20	31.88	70.79	**51.34**	**15.65**	6,82
GTREID	68.01	**50.00**	**78.8**	50.12	17.85	**444**

Table 1. UAVDT MOT tracking results. FairMOT and GTREID utilize built in detectors, while all other methods use detections provided by Faster-RCNN.

Table 1 shows GTREID having the best reported results on the UAVDT Benchmark. The largest improvements came from the increase in MOTP and IDs over the previously reported results. The final MOTP is 5.3% better than

MDP and 0.8% better than the FairMOT baseline. The ID switches are 97 less than MDP and 68 better than the FairMOT baseline. The data augmentations during the detector training are important in obtaining more detections, when objects go under occlussions such as street signs or trees on the side of the road. These small breaks in detections can result in ID switches that are avoided in the case of GTREID but may have troubled some of the other models.

FairMOT, GCNNMatch, and GTREID all experience a much higher percentage of mostly tracked vehicles. This is due to the stronger appearance features. The physical motion models combined with the superior feature vectors give the graph association networks better information to work with and make associations. GTREID further improves upon the results from FairMOT with the feature combination network. The two features act together as a multi-scale appearance feature as the baseline provides a single center-point representation. LABNet adds to this with a full representation of the entire bounding box. This, along with the graph network's ability to relate spatial information as part of the association, allows GTREID to gain an extra 0.37% MOTA and track 0.16% more vehicles during their track lifespan.

5.1. Ablation Study

To demonstrate the effectiveness of each network component, ablation studies were conducted. Baseline tests were performed with a batch size of 12, whereas the GTREID tests were done with a batch size of 8 due to GPU memory resource constraints. Random erase was performed 40% of the time with a scale ranging from 0.02-0.25 percent of the bounding box and an aspect ratio of 0.2, 0.5. Random noise was performed on 30% of images with a mean and standard deviation of 2.

Trial	Triplet	Augment	MOTA (%)	MOTP (%)	FP	FN	IDs	MT (%)	ML (%)
Baseline			47.455	78.78	43,776	134,990	363	43.68	**22.57**
Baseline		✓	47.209	78.708	43,144	136,530	293	42.87	23.31
Baseline	✓		47.195	79.059	42,266	137,470	285	42.54	23.31
Baseline	✓	✓	47.56	78.855	43,769	134,720	279	44.01	23.78
GTREID	✓	✓	**48.269**	**79.337**	**42,005**	**134,050**	**269**	**45.06**	22.82

Table 2. All performance tests were trained and tested on the UAVDT Benchmark. Triplet and Augmentation checkmark indicates the usage of each addition respectively. Values displayed in this table are for a minimum tracked area of 200 square pixels.

Trial	Triplet	Augment	MOTA (%)	MOTP (%)	FP	FN	IDs	MT (%)	ML (%)
Baseline			49.629	77.997	54,825	**116,380**	512	49.96	**17.69**
Baseline		✓	49.414	78.012	51,612	120,410	**429**	48.82	19.15
Baseline	✓		49.223	78.353	51,880	120,790	434	48.82	18.09
Baseline	✓	✓	**50.031**	78.104	53,216	116,680	457	50.04	18.42
GTREID	✓	✓	50.001	**78.792**	**50,140**	119,860	444	**50.12**	17.85

Table 3. All performance tests were trained and tested on the UAVDT Benchmark. Triplet and Augmentation checkmark indicates the usage of each addition respectively Values displayed in this table are for a minimum tracked area of 100 square pixels.

Tables 2 and 3 show the performance progression of GTREID for vehicles with a minimum box area of 200 and 100 respectively. GTREID outperforms the baseline in almost all reported metrics except for the number of Mostly Lost vehicles. The object improvements are less pronounced as the minimum box area is decreased from 200 to 100. This indicates the challenging conditions when the tracked objects get smaller and it gets harder to obtain representative features. This is noted in the larger change in ID switches that GTREID experiences compared to the baseline model. A similar effect is observed with the increase in false negatives as the minimum area of the tracked object is reduced from 200 to 100.

5.2. Visual Results

Figure 4. Examples of MOT on the UAVDT Benchmark using MDP (top), FairMOT (Middle), and GTREID (Bottom) where different IDs are represented with different color rectangles. Images are captured from frames 135 to 137 when viewed from left to right. The listed attributes for Sequence 1303 include a side viewpoint, daylight illumination, and low altitude.

Visual tracking examples are illustrated in Figures 4 and 5. Figure 4 shows a cropped example of Sequence 1303 in the UAVDT testing set. In this example, the full effect of GTREID's advantages can be seen. In the bottom left of the frame, it can be seen that MDP suffers from an identity switch as the purple yellow bounding boxes change vehicles between frames. This is due to their heavy reliance on optical flow to perform tracking and lower utilization of the appearance feature. FairMOT and GTREID are able to correctly track the object throughout the frames due to the usage of the appearance model and Kalman Filter together. GTREID is able to differentiate itself from FairMOT in situations such as those in the top of the frame. GTREID is the only model that is able to fully track the two overlapping black cars as well as the black car that is exiting the left hand side of the screen behind the light pole. This is due to the utilization of the data augmentation and graph association assisting to track localize the vehicles even though they are occluded. The graph network is then able to share the

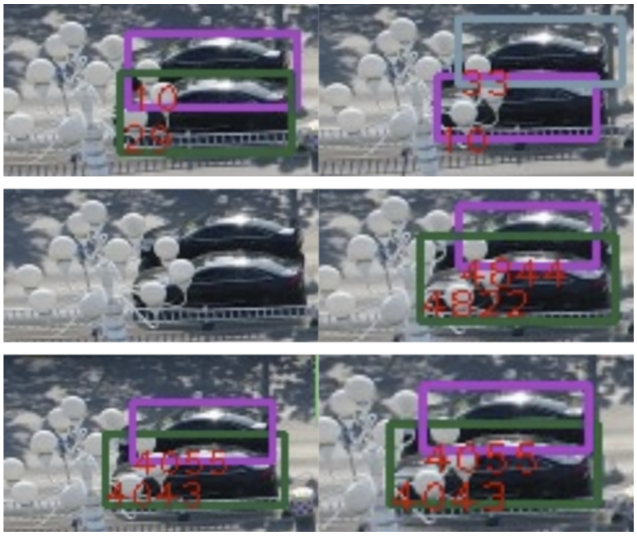

Figure 5. Examples of MOT on the UAVDT Benchmark using MDP (top), FairMOT (Middle), and GTREID (Bottom) where different IDs are represented with different color rectangles. Images are from frames 26 and 27 from left to right. The listed attributes for Sequence 1303 include a side viewpoint, daylight illumination, and low altitude.

bounding box location and appearance features in order to maintain the tracks despite the overlap.

Figure 5 shows another example of GTREID's performance upgrades from earlier in the sequence. In this example, the two vehicles maintain the same identities in both frames despite heavy occlusion and bounding box overlap, but they are also properly detected in both frames. The FRCNN detection from MDP has a slightly better localization around the car but the CenterNet based detection from GTREID localizes more tightly around both vehicles than the FairMOT detections.

6. Conclusion

We presented the GTREID multi-object tracking framework based on graph neural networks that combines center-point detection, graph association and re-identification. We demonstrate the usefulness of combining re-identification features from LABNet and FAIRMOT within our framework to obtain a better representation for tracking in the presence of occlusions. Results on the UAVDT benchmark dataset demonstrate that GTREID achieved state-of-the-art performance for aerial-based vehicle tracking.

7. Acknowledgments

This research was supported in part by AFOSR grant number FA9550-18-1-0121 and the Air Force Research Laboratory, Sensors Directorate (AFRL/RYAP) under contract number FA8650-18-C-1739 to Systems and Technology Research.

References

[1] Seyed Majid Azimi, Maximilian Kraus, Reza Bahmanyar, and Peter Reinartz. Multiple pedestrians and vehicles tracking in aerial imagery: A comprehensive study, 2020.

[2] Seung-Hwan Bae. Online multi-object tracking with visual and radar features. *IEEE Access*, 8:90324–90339, 2020.

[3] Keni Bernardin and Rainer Stiefelhagen. Evaluating multiple object tracking performance: The CLEAR MOT metrics. *EURASIP Journal on Image and Video Processing*, 2008.

[4] Luca Bertinetto, Jack Valmadre, João F Henriques, Andrea Vedaldi, and Philip HS Torr. Fully-convolutional siamese networks for object tracking. *arXiv preprint arXiv:1606.09549*, 2016.

[5] Erik Bochinski, Volker Eiselein, and Thomas Sikora. High-speed tracking-by-detection without using image information. In *International Workshop on Traffic and Street Surveillance for Safety and Security at IEEE AVSS 2017*, Lecce, Italy, Aug. 2017.

[6] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *CVPR*, 2019.

[7] Dawei Du, Yuankai Qi, Hongyang Yu, Yifan Yang, Kaiwen Duan, Guorong Li, Weigang Zhang, Qingming Huang, and Qi Tian. The unmanned aerial vehicle benchmark: Object detection and tracking. *ECCV*, 2018.

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. pages 770–778, 06 2016.

[9] Jie Hu, Li Shen, Samuel Albanie, Gang Sun, and Enhua Wu. Squeeze-and-excitation networks. 42(8):2011–2023, Aug. 2020.

[10] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, page 448–456. JMLR.org, 2015.

[11] Ajit Jadhav, Prerana Mukherjee, Vinay Kaushik, and Brejesh Lall. Aerial multi-object tracking by detection using deep association networks. In *2020 National Conference on Communications (NCC)*, pages 1–6, 2020.

[12] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[13] H. W. Kuhn and Bryn Yaw. The hungarian method for the assignment problem. *Naval Res. Logist. Quart*, pages 83–97, 1955.

[14] Bo Li, Wei Wu, Qiang Wang, Fangyi Zhang, Junliang Xing, and Junjie Yan. Siamrpn++: Evolution of siamese visual tracking with very deep networks, 2018.

[15] Bo Li, Junjie Yan, Wei Wu, Zheng Zhu, and Xiaolin Hu. High performance visual tracking with siamese region proposal network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[16] Jiahe Li, Xu Gao, and Tingting Jiang. Graph networks for multiple object tracking. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, March 2020.

[17] Chao Liang, Zhipeng Zhang, Yi Lu, Xue Zhou, Bing Li, Xiyong Ye, and Jianxiao Zou. Rethinking the competition between detection and reid in multi-object tracking, 2020.

[18] Tianyi Liang, Long Lan, and Zhigang Luo. Enhancing the association in multi-object tracking via neighbor graph, 2020.

[19] Qiankun Liu, Qi Chu, Bin Liu, and Nenghai Yu. Gsm: Graph similarity model for multi-object tracking. In Christian Bessiere, editor, *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 530–536. International Joint Conferences on Artificial Intelligence Organization, 7 2020. Main track.

[20] X. Liu, W. Liu, T. Mei, and H. Ma. Provid: Progressive and multimodal vehicle reidentification for large-scale urban surveillance. *IEEE Transactions on Multimedia*, 20(3):645–658, 2018.

[21] Zhichao Lu, Vivek Rathod, Ronny Votel, and Jonathan Huang. Retinatrack: Online single stage joint detection and tracking, 2020.

[22] Mohammad Hossein Nasseri, Hadi Moradi, Reshad Hosseini, and Mohammadreza Babaee. Simple online and realtime tracking with occlusion handling, 2021.

[23] Ioannis Papakis, Abhijit Sarkar, and Anuj Karpatne. GCNNMatch: Graph Convolutional Neural Networks for Multi-Object Tracking via Sinkhorn Normalization. *arXiv e-prints*, page arXiv:2010.00067, Sept. 2020.

[24] Aurelien Plyer, Guy Le Besnerais, and Frederic Champagnat. Massively parallel lucas kanade optical flow for real-time video processing applications. *Journal of Real-Time Image Processing*, 11:1–18, 04 2014.

[25] Mahdieh Poostchi, Kannappan Palaniappan, and Guna Seetharaman. Spatial pyramid context-aware moving vehicle detection and tracking in urban aerial imagery. In *IEEE AVSS International Workshop on Traffic and Street Surveillance for Safety and Security (IWT4S)*, 2017.

[26] Akshay Rangesh and Mohan Manubhai Trivedi. No blind spots: Full-surround multi-object tracking for autonomous vehicles using cameras and lidars. *IEEE Transactions on Intelligent Vehicles*, 4(4):588–599, 2019.

[27] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement, 2018.

[28] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2015.

[29] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[30] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[31] Chaobing Shan, Chunbo Wei, Bing Deng, Jianqiang Huang, Xian-Sheng Hua, Xiaoliang Cheng, and Kewei Liang. Tracklets predicting based adaptive graph tracking, 2020.

[32] Richard Sinkhorn and Paul Knopp. Concerning nonnegative matrices and doubly stochastic matrices. *Pacific Journal of Mathematics*, 21:343–348, 1967.

[33] Abu Md Niamul Taufique, Breton Minnehan, and Andreas Savakis. Benchmarking deep trackers on aerial videos. *Sensors*, 20(2):547, 2020.

[34] Abu Md Niamul Taufique and Andreas Savakis. Labnet: Local graph aggregation network with class balanced loss for vehicle re-identification. 2020.

[35] Abu Md Niamul Taufique, Andreas Savakis, Michael Braun, Daniel Kubacki, Ethan Dell, Lei Qian, and Sean M. O'Rourke. Siamreid: Confuser aware siamese tracker with re-identification feature, 2021.

[36] Dong Wang, Meng Yi, Fan Yang, Erik Blasch, Carolyn Sheaff, Genshe Chen, and Haibin Ling. Online single target tracking in wami: benchmark and evaluation. In *Multimedia Tools and Applications*, 2018.

[37] Qiang Wang, Yun Zheng, Pan Pan, and Yinghui Xu. Multiple object tracking with correlation learning, 2021.

[38] Yongxin Wang, Kris Kitani, and Xinshuo Weng. Joint object detection and multi-object tracking with graph neural networks, 2021.

[39] Zhongdao Wang, Liang Zheng, Yixuan Liu, Yali Li, and Shengjin Wang. Towards real-time multi-object tracking, 2020.

[40] Xinshuo Weng, Yongxin Wang, Yunze Man, and Kris M. Kitani. GNN3DMOT: Graph neural network for 3d multi-object tracking with 2d-3d multi-feature learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[41] N. Wojke, A. Bewley, and D. Paulus. Simple online and realtime tracking with a deep association metric. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 3645–3649, 2017.

[42] Yu Xiang, Alexandre Alahi, and Silvio Savarese. Learning to track: Online multi-object tracking by decision making. In *International Conference on Computer Vision (ICCV)*, 2015.

[43] F. Yu, D. Wang, E. Shelhamer, and T. Darrell. Deep layer aggregation. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2403–2412, 2018.

[44] Hongyang Yu, Guorong Li, Li Su, Bineng Zhong, Hongxun Yao, and Qingming Huang. Conditional gan based individual and global motion fusion for multiple object tracking in uav videos. *Pattern Recognition Letters*, 131:219–226, 2020.

[45] Hongyang Yu, Guorong Li, Weigang Zhang, Hongxun Yao, and Qingming Huang. Self-balance motion and appearance model for multi-object tracking in uav. In *Proceedings of the ACM Multimedia Asia*, MMAsia '19, New York, NY, USA, 2019. Association for Computing Machinery.

[46] Yifu Zhang, Chunyu Wang, Xinggang Wang, Wenjun Zeng, and Wenyu Liu. FairMOT: On the fairness of detection and re-identification in multiple object tracking. 2020.

[47] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34, 08 2017.

[48] Xingyi Zhou, Vladlen Koltun, and Philipp Krähenbühl. Tracking objects as points. *ECCV*, 2020.

[49] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. In *arXiv preprint arXiv:1904.07850*, 2019.

[50] Zheng Zhu, Qiang Wang, Bo Li, Wei Wu, Junjie Yan, and Weiming Hu. Distractor-aware siamese networks for visual object tracking, 2018.