SMAT: Smart Multiple Affinity Metrics for Multiple Object Tracking

Nicolas Franco Gonzalez(✉), Andres Ospina(✉), and Philippe Calvez(✉)

CSAI Laboratory at ENGIE, Paris, France
nicolas.franco@uao.edu.co, andres.ospina@external.engie.com,
philippe.calvez1@engie.com

Abstract. This research introduces a novel multiple object tracking algorithm called SMAT (Smart Multiple Affinity Metric Tracking) that works as an online tracking-by-detection approach. The use of various characteristics from observation is established as a critical factor for improving tracking performance. By using the position, motion, appearance, and a correction component, our approach achieves an accuracy comparable to state of the art trackers. We use the optical flow to track the motion of the objects, we show that tracking accuracy can be improved by using a neural network to select key points to be tracked by the optical flow. The proposed algorithm is evaluated by using the KITTI Tracking Benchmark for the class CAR.

Keywords: Online multiple object tracking · Tracking by detection

1 Introduction

Multiple object tracking or MOT is an important problematic in computer vision. This problematic has many potential applications, such as tracking and analyzing the movement of vehicles and pedestrians on the road, helping self-driving cars to make decisions, tracking and analyzing the movement of cells or organisms in time-lapse microscopy images or helping robots to pick up and place things in environments such as farms or industries. The broad area of application reflects the importance of developing accurate objects trackers.

MOT can be explained as the task of locating and tracking multiple objects of interest in video footage, identifying their position in every frame, and maintaining the identity (ID) of each target through its trajectory. There are many challenges in tracking multiple objects, such as the random motion of objects, crowded scenes, partial or full occlusion of objects, objects and camera viewpoint variations, illumination changes, background appearance changes, and non-ideal weather conditions.

This paper introduces an online MOT based on the tracking-by-detection paradigm. In an online approach, uniquely the previous tracked objects and the current frame are available to the algorithm. Tracking-by-detection means that in every frame, the objects are detected and considered as targets. The proposed

A. Campilho et al. (Eds.): ICIAR 2020, LNCS 12132, pp. 48–62, 2020.
https://doi.org/10.1007/978-3-030-50516-5_5

pipeline is composed of two main modules: detection and tracking. For detection, we test our system with two different detectors Faster-RCNN [24] and RRC [23]. The tracking algorithm is composed of three elements: Affinity metrics, data association and past corrector. for multiple object tracking The affinity metric outputs the probability of two observations from different frames being of the same target. For this we rely on three factors estimated from an observation: position, appearance, and motion. We use three affinity metrics inspired by state of the art trackers such as [4,5,30]: Intersection over Union (IoU) score, appearance distance, and optical flow affinity. The scores generated by the affinities are analyzed by the data association component with the objective of linking the current observations to the past observations, by giving the same ID in the cases that the target is the same. These process results are then passed into the corrector component, called tubulet interpolation which aims to fill empty spaces in the trajectories produced by detection failures.

There are multiple challenges and benchmarks for Multiple object tracking as MOT Challenge [7], KITTI Tracking Challenge [8], DETRAC [20] between others. This work uses to train, test and experiment using the KITTI Tracking Challenge [8] dataset for the car category. This limitation is due that we don't want to concentrate our efforts training the detector. The main idea is to concentrate on the tracking.

The main contributions of this paper are:

- The development of a novel tracking algorithm called Smart Multiple Affinity Metrics Tracker (SMAT). This algorithm combines three affinity metrics that evaluate the position, appearance, and motion of the targets.
- We tested the algorithm on the KITTI Tracking Benchmark and our approach produces competitive results. It was ranked 12th in this challenge (01/2020). Having the best multiple object tracking precision (MOTP). Also, in the subset of the top 12 submissions: we have the least identity switches (IDs) and the second best trajectory fragmentation (FRAG).
- Our experiments showed that the proposed affinity metrics complement each other to reduce errors produced along the tracking-by-detection framework.
- In near online [5], an affinity metric is used which is based on optical flow. We propose an improvement on the way the interest points are chosen for this metric by using an neural network called "hourglass"[26], instead of popular corner detectors as [27] and [25]. Better tracking accuracy results are obtained with the use of this network.
- A tubulet interpolation method was used to fill the empty spaces in a trajectory produced by detection failures. This technique allowed us to correct the past observations relying on the information provided by the motion model.

2 Related Work

Due to the rapid advancement in object detection thanks to convolutional neural network (CNN), tracking-by-detection has become a popular framework for

addressing the multiple target tracking problem. These methods depend on an object detector to generate object candidates to track. Then, based on the information extracted from detections (for example the position or appearance), the tracking is done by associating the detections.

A MOT approach can treat the association of the detections either as an online or offline problem. Global trackers [1,18,21] assume detections from all frames are available to process. In recent global trackers, the association is done by popular approaches as multiple hypotheses tracking (MHT) [13] and Bayesian filtering based tracking [15]. These methods achieve higher data association accuracy than online trackers as they consider all the detections from all frames. Contrary to global trackers, online trackers [12,30,31] do not use any data from future frames. They use the data available up to the current instance to tackle the association problem. Such trackers often solve this via the Hungarian algorithm [14]. Their advantage is that online methods can be applied in real-time applications such as autonomous cars. In these methods, a key factor for having an excellent performance is to use a relevant affinity metric. The affinity metric estimates how much similarity exists between 2 detections across time. Then, based on this information the association between these detections is done or not. For the affinity metric some trackers such as [3,4] rely in the information provided by the Intersection over Union (IoU) score. Other trackers such as [30] rely on the appearance information.

Recently, near online trackers proposed an Aggregated Local Flow Descriptor (ALFD) [5] to be used as affinity metric. The ALFD applies the optical flow to estimate the relative motion pattern between a pair of temporally distant detections. For doing that, the ALDF computes long term interest point trajectories (IPTs). If two observations have many IPTs in common this means that they are more likely to represent the same target. In [5] they use the algorithm FAST [25] for computing the interest points to be tracked by the optical flow. In contrast to these trackers, we propose a novel architecture that uses IoU score, optical flow affinity and appearance distance to infer if two observations correspond to the same target. The data association is done by the Hungarian algorithm.

3 Smart Multiple Affinity Tracker

The proposed algorithm (SMAT) is shown in Fig. 1. The inputs of the algorithm are the current frame and the identified tracked objects from the previous frame. In first place, the objects are detected in the current frame.

Then, three different algorithms compute the affinity or probability of being the same object between the detections and the previous tracked objects. These algorithms rely on three factors estimated from the detections and frames: position, appearance and motion. We use three affinity metrics: IoU score, appearance distance and optical flow affinity. The generated affinities are used by the data association component to link the current observations with the past observations by assigning the same ID in the cases that the object is the same. At each iteration, the corrector analyses if there is a re-identification of a lost target.

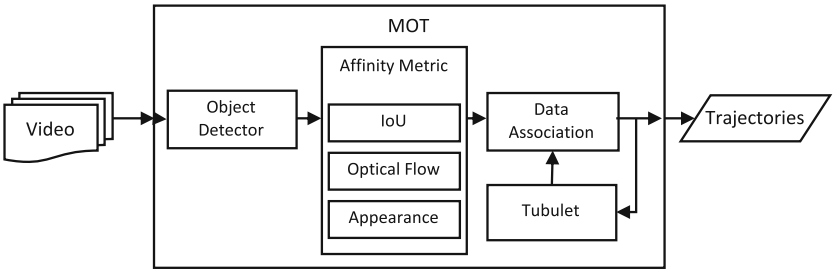

Fig. 1. SMAT overview

In that case, the corrector component will fill the empty spaces in the trajectories produced by detection failures using interpolation.

3.1 Object Detection

The proposed tracker is an online tracking-by-detection approach, where at each frame the objects of interest are detected and then associated to the targets of the past frames.

In this research we had two different stages:

- In the first experiment, we used the detector Faster R-CNN [24] with a ResNET-101 [6] as a backbone. This was selected based on the work developed by [10]. They evaluate the speed/memory/accuracy balance of different feature extractors and meta architectures. This configuration was chosen for the good trade-off between computing time and accuracy. We use this detector for the first experiment related to the affinity metrics.
- In the second experiment, aiming to improve our results, we used the detector RRC [23] due to its strong accuracy in the detection task.

The results obtained for those detectors in the KITTI Object Detection Benchmark [8] were:

Table 1. KITTI Object detection benchmark results

Method	Easy	Moderate	Hard
TuSimple [32] (Best)	94.47	95.12	86.45
RRC [23] (Used)	93.40	95.68	87.37
Faster-RCNN [24] (Used)	79.11	87.90	70.19

3.2 Affinity Metrics

In order to implement a MOT System, it is important to have an accurate measure to compare two detections through time. That is the job of the affinity metrics, which compare the detections from different frames and calculate their similarities scores. This information helps the data association to decide if the two detections represent the same target or not. The following paragraphs describe the affinity metrics used.

Intersection over Union Score: IoU is computed by dividing the area of overlap by the area of union between two bounding boxes that represent the detections. To use it as an affinity metric, the detections from frame t are compared with the tracked objects from frame $t-1$. The results are registered in a cost matrix named C_{IoU} that will be used by the data association. This process is summarized in Fig. 2. Where D_t and D_{t-1} are the predicted bounding boxes for the current frame and previous tracks.

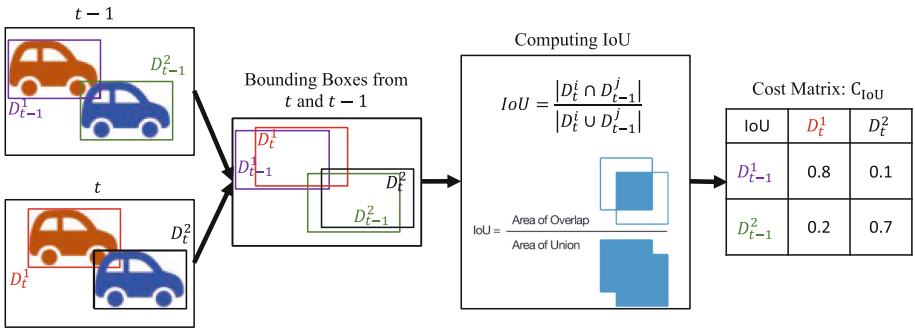

Fig. 2. Computing IoU cost matrix

Optical Flow Affinity: The Aggregated Local Flow Descriptor (ALFD) was introduced by [5] in 2015. The ALDF robustly measure the similarity between two detections across time using optical flow. Inspired by them, we developed a simplified version of the ALDF called optical flow affinity.

The optical flow affinity uses the Lucas-Kanade sparse optical motion algorithm [11]. This algorithm starts by identifying interest points (IPs) in the detections. Now, the optical flow algorithm tracks this points regardless of the detections, this track is called interest point trajectory (IPTs). Also, an ID is given to each trajectory.

To compute the affinity, the detection bounding box is divided in 4 sectors (as proposed in [5]) and a description of the detection is made based on the locations of the IPTs with respect to the sectors. Then, each tracked object from $t-1$ is compared with each detection from frame t. The number of IPTs

that are common per sector (number of matches) are counted and divided by the total number of matches and the number of IPTs of the target using Eq. 1.

$$score_i = \frac{\sum_{i=0}^{N_{sector}} matches(t_i, d_i)}{t_{i\ IDs}} \tag{1}$$

where d_i is the observation from current frame, t_i is the target from past frame, $t_{i\ IDs}$ is the total number of IPTs contained by the target and N_{sector} is the total number of sectors per target (4 in this case). The results are stored in a cost matrix C_{of} (see Fig. 3).

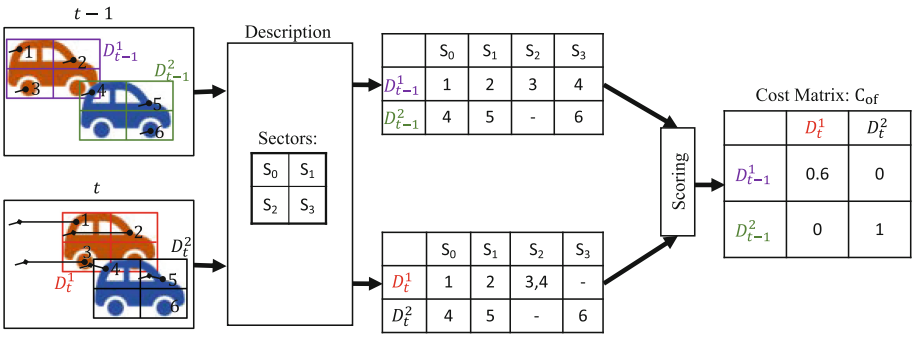

Fig. 3. Optical flow affinity

When working with sparse optical flow, the interest points (IPs) should be easy to be re-identified in the subsequent frames. Approaches as Shi Tomasi Corner Detector [27] or FAST [25] are used as a way of correctly choosing these IPs. However, these approaches do not have any notion of the shape of the target and therefore they find interest points in objects that are not of interest. These interest points are called outliers. The outliers reduce the accuracy of the optical flow affinity metric because it introduces wrong information to the evaluation. For reducing these outliers, we propose the use of a method that computes key-points for a given target.

In [26] they trained a stacked hourglass for the task of detecting key-points in vehicles. They obtained 93.4 of percentage of Correct Key-points (PCK) in the class car for the Pascal 3D dataset. The hourglass [22] can produce 36 points as shown in Fig. 4. In this work we selected only 8 points: (1, 2, 14, 15, 17, 32, 33, 35). These points were strategically chosen due to their position (easily identified in different views of a car), and the information that they provide (they are well tracked, and provide useful information on many components of the car to calculate the affinity metric).

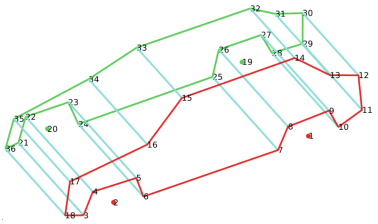

Fig. 4. Keypoints generated by hourglass

Appearance Distance: The appearance distance is a strong pairwise affinity metric used in modern trackers such as [30,33]. The main idea is to compare the car images, when the images contain the same objects the distance should be small, and large if they contain cars of different identities. This task is known as re-identification. For computing the appearance distance, a descriptor of an object that allows to discriminate between it and other similar objects is needed. To generate the descriptor or appearance vector we need to extract a set of features such as the car colour, car model, wheels model, etc. But they can also be more abstract, for example a combination of different curves and lines. In the case of deep learning, these features are represented by a vector. The feature vector or **appearance vector** has no meaning by itself but the distance of two vectors represent the similarity between the cars as depicted in Fig. 5).

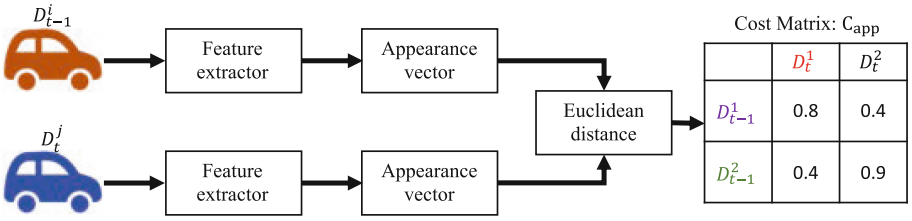

Fig. 5. Computing appearance distance (Color figure online)

For computing the appearance vector we use the Multiple Granularity Network (MGN) [29]. MGN was chosen because of its great performance in person re-identification datasets such as CUHK03 [16]. We trained the model similar to [29], with the main difference being the input size: 384×384. The model is trained in the VERI dataset [19]. For training, many mini-batch are created by randomly selecting 20 identities and randomly sampled 4 images for each identity. In result we get 80% mean average precision (mAP) for the test set in the VERI dataset.

We should note that, the result of this algorithm for similar cars the affinity distance is small. However, for the data association the affinity for similar cars should be large. Then, knowing that the maximum distance is one. For the data

association, the value of the appearance affinity is corrected by computing one minus the affinity distance.

3.3 Data Association

For the data association the Hungarian Algorithm [14] is used. The affinity metrics described previously are used when comparing the current detection with the previous tracked objects. Each affinity metric produces a cost matrix by comparing every detection with every tracked object. Then, we sum each cost matrix multiplied by a weight, as shown in Eq. 2.

$$C_{total} = w_{IoU}C_{IoU} + w_{of}C_{of} + w_{app}C_{app} \tag{2}$$

where C_{total} is the total cost matrix. w_{IoU}, w_{of} and w_{app} are the weights. This multiplication is done to prioritize or balance the costs because of the nature of the affinity, i.e. the values of the optical flow affinity have a mean value lower than the other two affinities. Therefore, we choose to multiply the optical flow cost matrix by 1.4. The other affinities are multiplied by 1.

Then using the total cost matrix C_{total}, the Hungarian Algorithm will assign which detections represent the same target by maximizing the cost assignment. Associations with a score lower than 0.3 are deleted. Then, new identities are created with the unmatched detections. The information of the terminated tracked objects or tracklets (identities that were not found in the present frame) is stored. Then, in subsequent frames the algorithm will look for reappearances of these terminated tracklets. This means that, in every frame we will first compare the detections from the present frame with the detections from the past frame. The detections not associated will be compared with the terminated tracklets. If there are some matches, the old IDs will be assigned. Otherwise, new ids will be generated. In practice, the system stores the information of terminated tracklets for 13 frames. If during that period the ID does not reappear then this will be definitely deleted.

3.4 Estimating Trajectories for Partially Lost Objects

When the object detection fails, as depicted in Fig. 6 at the time $t-1$, a fragmentation in the estimation of a target's trajectory is produced. The fragmentation

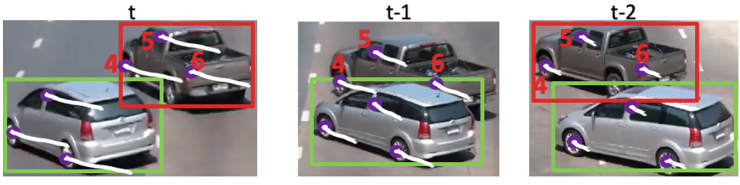

Fig. 6. Detection failure

happens when it is unknown the position of the target over a period of time. To reduce fragmentation we propose a technique called tubelet interpolation.

Although the object is not detected, the optical flow still follows the target. Relying on this information an interpolation of the bounding box is done for filling the empty spaces in the trajectories (see Fig. 7).

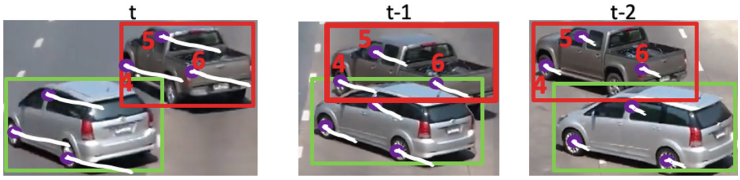

Fig. 7. Tubulet interpolation

Procedure: The correction of the fragmentation starts; the information of the matched bounding boxes is known. The current bounding box in time t and the last known bounding box $t - n$ are assessed. Therefore, the object was lost for n frames. The bounding boxes coordinates are defined as $[x_1, y_1, x_2, y_2]$, where (x_1, y_1) is the top left corner and (x_2, y_2) is the bottom right corner of the rectangle. It is assumed that velocity between frames is linear. Equation 3 computes the velocities v_x and v_y of the targets between frame t and $t - \Delta t$. Also, the width w_r and height h_r change ratio are calculated using the Eq. 4 (for v_y and h_r replace x by y in Eqs. 3 and 4). Finally, to reproduce the bounding box coordinates between the frames, Eq. 5 and 6 (for y_1 and y_2 replace x by y and w_r by h_r) is used.

$$v_x = \frac{x_{1(t)} - x_{1(t-n)}}{n} \tag{3}$$

$$w_r = \frac{(x_{2(t)} - x_{1(t)}) - (x_{2(t-n)} - x_{1(t-n)})}{n} \tag{4}$$

$$x_{1(t-n+k)} = x_{1(t-n)} + v_x * k \tag{5}$$

$$x_{2(t-n+k)} = x_{1(t-n)} + (x_{2(t-n)} - x_{1(t-n)}) + w_r * k \tag{6}$$

where k is a number between 0 and n.

4 Experimentation

Our approach was evaluated in the KITTI Tracking Benchmark [8] on training and testing dataset. Different configurations of SMAT were proposed. The best performing configuration in the training set was used to report the result on the benchmark on the Car class.

4.1 Metrics

The metrics used to evaluate the multi-target tracking performance are defined in [17], along with the widely used CLEAR MOT metrics [2]. Some of these metrics are explained below, where (↑) means the higher the better and (↓) the lower the better:

- MOTA(↑): Multi-object tracking accuracy
- MOTP(↑): Multi-object tracking precision
- MT(↑): Ratio of ground truth trajectories successfully tracked for at least 80 % of their life span.
- ML(↓): Mostly lost trajectories. Trajectories tracked for less than 20% of its total length.
- PT(↓): The ratio of partially tracked trajectories, i.e., MT - ML
- FP(↓): Total number of wrong detections
- FN(↓): Total number of missed detections
- ID sw(↓): Number of times the ID of a tracker switches to a different previously tracked target
- Frag(↓): Number of times a trajectory is interrupted during tracking.

Table 2. Results in the KITTI tracking training set using different configurations

Config	MOTA	MOTP	Recall	Precision	MT	PT	ML	TP	FP	FN	IDS	FRAG
IoU	0.8776	0.9107	0.9164	0.9867	0.8422	0.1525	0.0053	24797	334	2261	350	926
IoU+Sh	0.8870	0.9107	0.9164	0.9867	0.8422	0.1525	0.0053	24797	334	2261	126	705
IoU+FS	0.8853	0.9107	0.9164	0.9867	0.8422	0.1525	0.0053	24797	334	2261	167	744
IoU+Hg	**0.8878**	0.9107	0.9164	0.9867	0.8422	0.1525	0.0053	24797	334	2261	**105**	**687**
IoU+Ap	0.8882	0.9107	0.9164	0.9867	0.8422	0.1525	0.0053	24797	334	2261	96	685
IoU+Sh+Ap	0.8895	0.9107	0.9164	0.9867	0.8422	0.1525	0.0053	24797	334	2261	64	650
IoU+Hg+Ap	**0.8899**	0.9107	0.9164	0.9867	0.8422	0.1525	0.0053	24797	334	2261	**55**	**642**
IoU+FS+Ap	0.8896	0.9107	0.9164	0.9867	0.8422	0.1525	0.0053	24797	334	2261	63	650
IoU+Sh+Ap+Tb	0.9151	0.9069	0.9509	**0.9768**	0.9238	0.0727	0.0035	25982	**618**	1342	**84**	306
IoU+Hg+Ap+Tb	**0.9201**	0.9062	**0.9563**	0.9762	**0.9326**	**0.0638**	0.0035	**26158**	639	**1195**	89	**266**
IoU+FS+Ap+Tb	0.9160	**0.9066**	0.9535	0.9753	0.9255	0.0709	0.0035	26050	659	1269	93	293

4.2 Experiments

Different configurations were tested with the training set of KITTI Tracking. In the first experiment is evaluated the performance of using a tracker with IoU as an affinity metric. Surprisingly, a MOTA of 87.76% was obtained. However, there were many id-switches. For reducing the number of ID switches to improve the accuracy we added the optical flow affinity to the tracker formulation. As the optical flow affinity highly depends on the IPs, different interest points detectors were used to see which could give better results. The configurations proposed were:

- **IoU:** Tracking using intersection over union as affinity
- **IoU+Sh:** IoU and optical flow with Shi Tomasi
- **IoU+FS:** IoU and optical flow with FAST
- **IoU+Hg:** IoU and optical flow with Hourglass

The results are shown in Table 2. In all cases, using optical flow improves the MOTA by reducing the ID switches. This is because in some situations there are large movements of vehicles from one frame to another generating low IoU scores. However, in these situations the optical flows can still provides relevant information to associated the ids. Also, in the situation were some objects are moving very close to each other, the optical flow affinity helps to discriminate well between these. The configuration with less ID switches was the one that uses an hourglass as interest point detector. This is because the hourglass was trained to find key-points on vehicles so it presents less outliers points than the others. In the case of FAST and Shi Tomasi they are looking for finding corners in the image. In many cases, bounding boxes contains not only the vehicle, also other pieces of objects. That causes that these corner detectors produce points in zones that are not interesting, generating wrong points to track.

The IoU score and the optical flow affinity fails in cases were the objects are occluded. In order to make our tracker robust in these situations, the appearance distance metric (+Ap) was added to the configurations aforementioned. By doing this we managed to obtain a MOTA of 88.99% and reduce the id-switches from 350 to 55 in the best method (see IoU+Hg+Ap from Table 2).

Although the ID switches were greatly reduced and the MOTA was increased from 87.76% to 88.99%, the model was not still good enough to be ranked in the first 20 positions of the challenge. This was partly because there were many false negatives (2261 in all methods shown). Due to the false negatives the models were also presenting many fragmentations (642 in the best case). When evaluating MOT system, each trajectory had a unique start and end and it was assumed that there was no fragmentation in the trajectories [2]. However, the object detectors present failures between frames. This increases the number of false negatives and fragmentation. To deal with this we added a tubulet interpolation (+Tb) to the tracking formulation as it was explained before. From Table 2 we concluded that the positive aspects of the interpolation are: the number of fragmentation and false negatives are reduced by filling the empty spaces of the trajectories, the recall is increased by generating more bounding boxes, the mostly tracked (MT) metric is increased, and the ratio of partially tracked trajectories is reduced along with the mostly lost (ML) metric. The negative aspects are: the false positives and the id-switches increase because sometimes the corrections of past frames are wrong and the precision decreases because in some cases the created bounding box does not match completely well the objects. Although the tubulet interpolation has negative aspects, the MOTA increased more than 2% in all the configurations, proving that the effect of positive aspects outweighted the negatives. The Table 3 shows the effect in percentage of adding different components to a basic tracker that uses only IoU. Green values mean the result is improved while red values means the result gets worse.

Table 3. Components contributions in % training set

Config	MOTA%	MOTP%	Recall%	Precision%	MT%	PT%	ML%	TP%	FP%	FN%	IDS%	FRAG%
+OF	+ 0.99±0.21	=	=	=	=	=	=	=	=	=	- 61±9	- 23±3
+Ap	+ 1.6	=	=	=	=	=	=	=	=	=	- 73	- 30
+OF+Ap	+ 2±0.2	=	=	=	=	=	=	=	=	=	- 83±1	- 30
+OF+Ap+Tb	+ 4±0.2	- 0.42±0.3	+ 3.7±0.3	- 1	+ 8.9±0.6	- 8	- 0.2	+ 5.2±0.3	+ 91±6	- 43.5±3.5	- 75±1	- 70.5±0.5

During the writing of this paper, we saw the opportunity of using a better detector called RRC in the place of the Faster R-CNN. Therefore, using the best performing model in the previous experiment (IoU + Hg + Ap + Tb), the model is tested with the RRC detector. The results are shown in the next section.

4.3 Results

Two submission were done for the KITTI Tracking Benchmark. One using a Faster R-CNN as object detector and other employing and Accurate Single Stage Detector Using Recurrent Rolling Convolution RRC. Both of them using the best performing model (IoU + Hg + Ap + Tb)]. In the Table 4 our approach (SMAT) with the best models of the challenge is compared. Due to the fact that the RRC presents better object detection accuracy than the Faster-RCNN, the architecture that use RRC is the best performing. SMAT+RRC is ranked 12th while SMAT+F-RCNN is ranked 20th in the challenge. Models that used other sensors different to the camera were not included in the Table 4. As shown, SMAT has competitive results in comparison with state of the art trackers.

Table 4. Results in the KITTI Tracking Benchmark

Config	MOTA%	MOTP%	MT%	ML%	IDS	FRAG
MASS [12]	85.04	85.53	74.31	2.77	301	744
SMAT+RRC (ours)	84.27	86.09	63.08	5.38	28	341
MOTBeyPix [26]	84.24	85.73	73.23	2.77	468	944
IMMDP [31]	83.04	82.74	60.62	11.38	172	365
JCSTD [28]	80.57	81.81	56.77	7.38	61	643
extraCK [9]	79.99	82.46	62.15	5.54	343	938
SMAT+F-RCNN (ours)	78.93	84.29	63.85	4.77	160	679

5 Conclusion

In this paper we propose a novel tracker architecture that uses the position, motion and appearance as characteristics for associating the targets to the observations. Based on these characteristics three affinity metrics were implemented: IoU score, optical flow affinity and appearance distance. Our experiments showed that for tracking the motion using optical flow the results are highly dependent on the selection of the interest points. A neural network called "hourglass" is used in order to compute interest points to follow. By using this instead of classical interest point detectors the tracking accuracy is improved. Through experiments we showed that the affinity metric complement each other to reduce mistakes committed in the tracking-by-detection framework. An analysis of the contributions generated for adding each affinity to the tracking formulation is done. A method called tubelet interpolation was proposed in order to reduce the fragmentation generated by detections failures. This method relies on the information provided by the optical flow. Finally, the proposed algorithm presents competitive results as it was ranked 12th in the KITTI Tracking Benchmark for the class Car.

In future work, we will see the performance difference between using a segmentation network plus a classic interest point detector, instead of the detection network plus the hourglass network in order to compute key points. The segmentation will avoid the points outside of the object. Therefore, the difference in time and performance could be studied. In the other hand, we will experiment different position models as a Kalman filter and how we can joint the information of the optical flow. Also, we will study other data association algorithms.

References

1. Berclaz, J., Fleuret, F., Turetken, E., Fua, P.: Multiple object tracking using k-shortest paths optimization. IEEE Trans. Pattern Anal. Mach. Intell. **33**(9), 1806–1819 (2011). https://doi.org/10.1109/TPAMI.2011.21
2. Bernardin, K., Stiefelhagen, R.: Evaluating multiple object tracking performance: the clear mot metrics. J. Image Video Process. **2008**, 1 (2008)
3. Bewley, A., Ge, Z., Ott, L., Ramos, F., Upcroft, B.: Simple online and realtime tracking. In: 2016 IEEE International Conference on Image Processing (ICIP), pp. 3464–3468. IEEE (2016)
4. Bochinski, E., Eiselein, V., Sikora, T.: High-speed tracking-by-detection without using image information. In: 2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), pp. 1–6. IEEE (2017)
5. Choi, W.: Near-online multi-target tracking with aggregated local flow descriptor. In: 2015 IEEE International Conference on Computer Vision (ICCV), December 2015. https://doi.org/10.1109/iccv.2015.347, http://dx.doi.org/10.1109/iccv.2015.347
6. Dai, J., Li, Y., He, K., Sun, J.: R-FCN: Object detection via region-based fully convolutional networks. In: Advances in Neural Information Processing Systems, pp. 379–387 (2016)

7. Dendorfer, P., et al.: CVPR19 tracking and detection challenge: how crowded can it get? arXiv:1906.04567 [cs], June 2019, arXiv: 1906.04567
8. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? the kitti vision benchmark suite. In: Conference on Computer Vision and Pattern Recognition (CVPR) (2012)
9. Gündüz, G., Acarman, T.: A lightweight online multiple object vehicle tracking method. In: 2018 IEEE Intelligent Vehicles Symposium (IV). pp. 427–432, June 2018. https://doi.org/10.1109/IVS.2018.8500386
10. Huang, J., et al.: Speed/accuracy trade-offs for modern convolutional object detectors. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7310–7311 (2017)
11. Kanade, L.: An iterative image registration technique with an application to stereo vision
12. Karunasekera, H., Wang, H., Zhang, H.: Multiple object tracking with attention to appearance, structure, motion and size. IEEE Access **7**, 104423–104434 (2019). https://doi.org/10.1109/ACCESS.2019.2932301
13. Kim, C., Li, F., Ciptadi, A., Rehg, J.M.: Multiple hypothesis tracking revisited. In: 2015 IEEE International Conference on Computer Vision (ICCV), pp. 4696–4704, December 2015. https://doi.org/10.1109/ICCV.2015.533
14. Kuhn, H.W.: The hungarian method for the assignment problem. Naval Res. Logistics Quarterly **2**(1–2), 83–97 (1955)
15. Lee, B., Erdenee, E., Jin, S., Nam, M.Y., Jung, Y.G., Rhee, P.K.: Multi-class multi-object tracking using changing point detection. In: Hua, G., Jégou, H. (eds.) ECCV 2016. LNCS, vol. 9914, pp. 68–83. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-48881-3_6
16. Li, W., Zhao, R., Xiao, T., Wang, X.: Deepreid: deep filter pairing neural network for person re-identification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 152–159 (2014)
17. Li, Y., Huang, C., Nevatia, R.: Learning to associate: hybridboosted multi-target tracker for crowded scene. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 2953–2960, June 2009. https://doi.org/10.1109/CVPR.2009.5206735
18. Zhang, L., Li, Y., Nevatia, R.: Global data association for multi-object tracking using network flows. In: 2008 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8, June 2008. https://doi.org/10.1109/CVPR.2008.4587584
19. Liu, X., Liu, W., Ma, H., Fu, H.: Large-scale vehicle re-identification in urban surveillance videos. In: 2016 IEEE International Conference on Multimedia and Expo (ICME), pp. 1–6, July 2016
20. Lyu, S., et al.: UA-DETRAC 2017: Report of AVSS2017 & IWT4S challenge on advanced traffic monitoring. In: 2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), pp. 1–7. IEEE (2017)
21. Milan, A., Roth, S., Schindler, K.: Continuous energy minimization for multitarget tracking. IEEE Trans. Pattern Anal. Mach. Intell. **36**(1), 58–72 (2014). https://doi.org/10.1109/TPAMI.2013.103
22. Murthy, J.K., Sharma, S., Krishna, K.M.: Shape priors for real-time monocular object localization in dynamic environments. In: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 1768–1774. IEEE (2017)
23. Ren, J., et al.: Accurate single stage detector using recurrent rolling convolution. http://arxiv.org/abs/1704.05776

24. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. Adv. Neural Inf. Process. Syst. **36**, 91–99 (2015)
25. Rosten, E., Porter, R., Drummond, T.: Faster and better: a machine learning approach to corner detection. IEEE Trans. Pattern Anal. Mach. Intell. **32**(1), 105–119 (2008)
26. Sharma, S., Ansari, J.A., Murthy, J.K., Krishna, K.M.: Beyond pixels: leveraging geometry and shape cues for online multi-object tracking. In: 2018 IEEE International Conference on Robotics and Automation (ICRA), pp. 3508–3515. IEEE (2018)
27. Shi, J., et al.: Good features to track. In: 1994 Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 593–600. IEEE (1994)
28. Tian, W., Lauer, M., Chen, L.: Online multi-object tracking using joint domain information in traffic scenarios. IEEE Trans. Intell. Transport. Syst. **39**, 1–11 (2019)
29. Wang, G., Yuan, Y., Chen, X., Li, J., Zhou, X.: Learning discriminative features with multiple granularities for person re-identification. In: 2018 ACM Multimedia Conference on Multimedia Conference, pp. 274–282. ACM (2018)
30. Wojke, N., Bewley, A., Paulus, D.: Simple online and realtime tracking with a deep association metric. In: 2017 IEEE International Conference on Image Processing (ICIP), pp. 3645–3649. IEEE (2017)
31. Xiang, Y., Alahi, A., Savarese, S.: Learning to track: online multi-object tracking by decision making. In: The IEEE International Conference on Computer Vision (ICCV), December 2015
32. Yang, F., Choi, W., Lin, Y.: Exploit all the layers: fast and accurate CNN object detector with scale dependent pooling and cascaded rejection classifiers. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2016
33. Yu, F., Li, W., Li, Q., Liu, Y., Shi, X., Yan, J.: POI: multiple object tracking with high performance detection and appearance feature. In: Hua, G., Jégou, H. (eds.) ECCV 2016. LNCS, vol. 9914, pp. 36–42. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-48881-3_3