Multiple Object Detection in 360° Videos for Robust Tracking

V. Vineeth Kumar[1(✉)], Shanthika Naik[1], Polisetty L. Sarvani[1], Shreya M. Pattanshetti[1], Uma Mudenagudi[1], Meena Maralappanavar[1], Priyadarshini Patil[1], Ramesh A. Tabib[1], and Basavaraja S. Vandrotti[2]

[1] KLE Technological University, Hubballi, India
vellalavineethkumar@gmail.com, uma@kletech.ac.in
[2] Samsung R&D Institute, Bangalore, India

Abstract. In this paper, we propose an efficient way to detect objects in 360° videos in order to boost the performance of tracking on the same. Though extensive work has been done in the field of 2D video processing, the domain of 360° video processing has not been explored much yet, as it poses difficulties such as (1) unavailability of the annotated dataset (2) severe geometric distortions at panoramic poles of the image and (3) high resolution of the media which requires high computation capable machinery. The State-of-the-art detection algorithm involves the use of CNN (Convolution Neural Networks) trained on a large dataset. Faster RCNN, SSD, YOLO, YOLO9000, YOLOv3 *etc.* are some of the detection algorithms that use CNN. Among these, though YOLOv3 might not be the most accurate, it is the fastest, and this trade-off between speed and accuracy is acceptable. We improvise upon this algorithm, to make it suitable for the 360° dataset. We propose YOLO360, a CNN network to detect objects in 360° videos and thus increase the tracking precision and accuracy. This is achieved by performing transfer learning on YOLOv3 with the manually annotated dataset.

Keywords: Computer vision · Equirectangular frames · 360° images · Detection · Tracking · Transfer Learning

1 Introduction

In this paper we address the problem of object tracking for multiple object detection and tracking in 360° videos, focusing on 'Person' and 'Car' class. Multiple object detection and tracking is a well-explored domain with respect to normal videos. However, the same is not true in case of 360° videos. Hence, our work focuses on fulfilling this gap in the domain of 360° videos. Multiple object detection and tracking find its application in the fields of security and surveillance [1–3], the interaction between humans and computer and navigation of UAVs (Unmanned Ariel Vehicles) and robots.

In what follows, we provide a literature survey of different object detection methods, object tracking methods, and applications of object tracking.

B. Deka et al. (Eds.): PReMI 2019, LNCS 11942, pp. 499–506, 2019.
https://doi.org/10.1007/978-3-030-34872-4_55

Object detection involves object localization [4,5] and object classification [6,7]. Object localization refers to locating an object(s) in an image by providing their coordinates. Object classification is categorizing these objects into their respective classes such as ‘person’, ‘cat’, ‘dog’, ‘car’ *etc.* Object detection finds its application in various fields such as face recognition, crowd counting, security, tracking *etc.*

Object tracking refers to keeping track of an object throughout a video or in an actual environment. There are two main approaches to object tracking: (1) Tracking by detection [8,9]. (2) Kernel-based tracking [10]. In tracking by detection objects are detected in every frame and then are assigned to their respective tracklets by data association algorithms. We have a set of tracks, representing the position of each object in each frame.

The primary challenge in object detection in 360° videos is severe distortion at projection poles. Objects in the images are deformed due to these distortions and are beyond recognition, even by humans. Classical detection algorithms, trained for normal video dataset cannot handle these distortions. We use tracking by detection based algorithm for our experimentation. The bottleneck here is that the efficiency of the tracking algorithm highly depends on the robustness and efficiency of the detection algorithm.

To address these issues, we propose a robust tracker for multiple objects in 360° videos. Towards this, our contributions are:

- We propose a YOLO360, an architecture trained to detect multiple objects in 360° videos, in-particular
  - We construct dataset with ground truth object positions in ER frames extracted from the 360° videos.
  - We propose YOLO360 to detect objects in 360° videos using transfer learning on YOLOv3, Which is used for training Dataset generation: We perform annotations for ‘Person’ and ‘Car’ classes on ER frames extracted from the 360° videos to generate ground truth for our dataset. We use “Yolomark” GUI for annotation of images.
  - We train YOLO360 using these annotated images.
- Ground truth for tracking algorithm: We generate ground truth of total 4000 frames from 9 different videos, which is further used to evaluate the performance of the tracker.

In Sect. 2, we provide the proposed framework of detection and tracking algorithm using YOLO360. We also provide details training YOLO360 using transfer learning of YOLOv3. We demonstrate our results in Sect. 3. We provide conclusions in Sect. 4.

2 Framework for Multiple Object Detection and Tracking

In this section, we provide the proposed framework of multiple object tracking, as shown in Fig. 1. The framework contains 3 major phases *viz*, (1) Creation of Dataset and Training, (2) Object Detection, (3) Object Tracking.

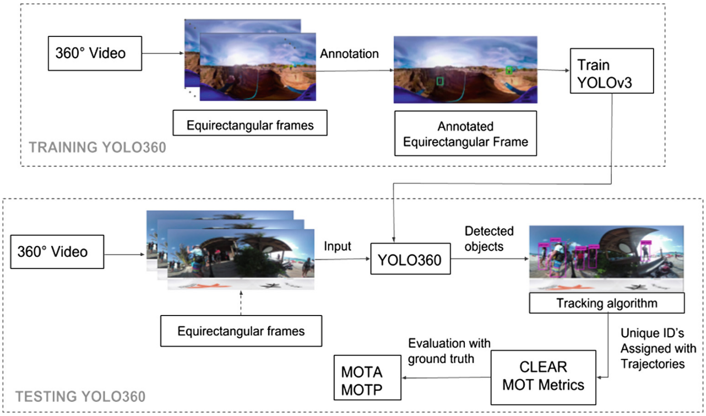

Fig. 1. Proposed pipeline of Multi object tracking in 360° videos.

We explain the first phase in Sect. 2.1. The second phase in every tracking framework is an object detection module. To propose a learning model for detection of objects in 360° videos, there are very few dataset available. This necessitates the creation of training data. We convert 360° videos into ER frames and annotate objects. Once the model is ready with adjusted weights we feed ER frames or 360° video as input to the model. The output is an image with the bounding box coordinates of the detected objects for each video. These coordinates are further given as an input to the tracking algorithms SORT [11] and DEEP SORT [12]. The output of Tracking algorithm comprises output video with the unique ID's assigned and trajectories traced along all the frames of the video. The output of Tracking is given to CLEAR MOT [13] metrics to evaluate the MOTA (Multiple Object Tracking Accuracy) and MOTP (Multiple Object Tracking Precision) score against the ground truth generated.

2.1 360° video dataset

360° images and video frames are the 3D representation of the real world. If the real world is perceived as a sphere, with the camera capturing the image as its center, then each point of the said sphere corresponds to a pixel in the 360° image. 2D projection of this 3D sphere is used to store these images. Equirectangular panorama (ER) is one such projection popularly used. The projection maps each point on the sphere represented by two angles: latitude $\psi \epsilon$ [90°, +90°] and longitude $\lambda \epsilon$ [180°, +180°] to an x,y coordinate of the 2D plane. In this work, we use ER frames for further processing.

YOLOv3 is originally trained on 'Microsoft COCO' [14] dataset. The significant differences between the COCO dataset and ours are as follows:

- In our dataset, there are a number of small objects because the YOLOv3 trained on COCO dataset did not produce accurate detection on 360° videos.

- As there are no severe geometric distortions at the central horizontal line, we concentrated on annotating objects at non central horizontal line (non-equator plane) where there are the severe geometric distortions, and the model learns these distortions.

Among the various objects available in the COCO dataset such as signboards, trucks, bicycles *etc*, we selected annotated objects: Person (3300 frames), Car (700 frames) for our experiments.

2.2 Sphere to Cubic Projections

On projection of the 3D sphere onto a 2D plane, severe geometric distortions are caused at the panoramic poles. These distortions pose one of main challenges for detection of objects in the ER frames. One potential solution is to map sub-windows of the 360° sphere to cubic map as shown in Fig. 2. The traditional detection algorithm work on these cubic projections just as well as they do on normal images.

This spherical to cubic mapping can be pictured as follows: The cube onto which the sphere is to be mapped contains the sphere. The center of both sphere and cube coincide. Consider a pyramid with its apex at the center of the sphere and a face of the cube as its base. The sector of the sphere enclosed within this pyramid is projected on to the base. Thus the entire sphere is divided into 6 sectors projected onto six faces of the cube.

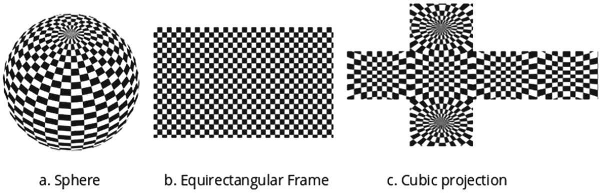

Fig. 2. Sphere to Equirectangular and Cubic mapping

This gives us a picture as to how the box is constructed. We employ the same technique for the ER frames extracted from the 360° videos. We convert the ER frames to cubic and run the detection algorithm YOLOv3 trained on “Microsoft COCO” dataset. The coordinates thus obtained are saved and further fed to SORT and DEEP SORT algorithm. Our experiments show that the detection in cubic frames using YOLOv3 had higher accuracy (4–5%) as compared to the detection on ER frames.

However this method is computationally expensive as it involves conversion from ER frames to cubic mapping, run detection methods trained on conventional images, get the coordinates and then use these coordinates for tracking in ER representation. By following the above method, we come to two conclusions:

(1) The detection accuracy was high in Method 1 when compared to Method 2.
(2) The time taken for Method 1 is twice the time taken for method 2 as it involved the conversion of ER frames to cubic projections. The drawbacks of this approach are overcome by our proposed approach 'YOLO360'.

2.3 YOLO360

The scarcity of the labeled data is the main bottleneck to train deep learning architecture for detection in 360° videos. Authors in [15,16] show that the results of models trained on different tasks reduce the bottleneck of learning a new network for different tasks. The study also shows that learning the new architecture with pre-trained features improved the generalization even after fine tuning with the new dataset.

Hence we propose YOLO360, an object detection algorithm which can directly detect people in 360-degree videos with transfer learning, reducing the overhead of converting to Cubic projections and re-converting them back to Equirectangular frames to run the tracking algorithm. We've chosen YOLOv3 [17] and are improving upon the same to make it suitable for 360° videos. YOLOv3 is a CNN architecture, designed to detect multi-class multiple objects in normal videos. Owing to its speed and accuracy, it is most suitable for real-time object detection. We've employed online transfer learning [18] to retrain this architecture to make to suitable of 360° dataset. The architecture is originally trained of 80 classes of COCO dataset. We've retrained it for two classes, 'Person' and 'Car' classes of 360° images. There's a substantial increase in the efficiency with this retrained weights when compared to weights trained for normal images.

We use YOLOv3 architecture on which we perform transfer learning. Further we can increase the number of classes for training by changing the number of filters as per the equation 'filters'$=$ (No of classes $+ 5) \times 3$. When we observe that average loss no longer decreases at many iterations, we stop the training, and This model can be used for inference. According to our experiments, YOLO360 overcomes the false detection made by YOLOv3 after 2000 iterations.

2.4 Object Tracking

Our main focus is to improve the detection algorithm in order to improve the tracking efficiency. We use SORT and DEEP SORT algorithms for tracking. SORT exploits both track by detection and kernel-based tracking methods. Multiple objects are first detected using a detection algorithm. These detection are assigned to their respective tracks based on IOU of detection and regions proposed by Kalman filter. The filter is then updated with the detection from each incoming frame, after assignment. DEEP SORT is an improvised version of SORT. In SORT only motion of the object is considered, whereas, in deep sort, both motion and appearance of the objects are considered. An association metrics is built for incoming detection and exiting tracklet, using formula

$$c_{i,j} = \lambda d^{(1)}(i,j) + (1-\lambda) d^{(2)}(i,j) \tag{1}$$

where $d^{(1)}$ is value given by motion model and $d^{(2)}$ by appearance model. λ decides the weight to be given to the two models. With repeated experiments, we conclude that both models combined, give better results than considering either one of them. A CNN network is used to obtain an appearance descriptor, which in combination with the position is used to assign objects to tracks.

We make use of both the motion model and the appearance model of the DEEP SORT and create a variant of DEEP SORT by changing the values of λ to demonstrate our results on 360° videos.

3 Results and Discussions

The 9 (1280 × 720) 360° videos selected for our dataset are selected from YouTube, captured and uploaded by users and from the Stanford University 360° videos dataset. We have chosen a total of 4000 frames and annotated 'Person' and 'Car' classes in each of these frames. We provide a pre-trained model along with dataset generated for training, ground truth and their results in our GitHub repository. The chosen videos represent moving objects as this would help us in building accuracy of detection, which is further fed to the tracking algorithm. We perform training on NVIDIA DGX-1 and also observe that it takes five hours for the weights to converge.

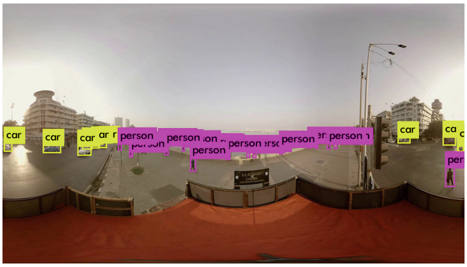

Fig. 3. Detection result for Person and Car for the frame using YOLO360.

We demonstrate the detection and tracking results on 360° videos. Multi class detection results are shown in Figs. 3 and 4. We can observe that YOLO360 detects 15 objects from person class and 7 objects from car class out of total 20 objects from person class and 7 objects from car class respectively.

The results of tracking using MOTA and MOTP are shown in Table 1. We compare MOTA and MOTP results using SORT and DEEP SORT tracking algorithm, with detection from both YOLOv3 and YOLO360. The results are calculated using CLEAR MOT benchmark. The cumulative average of the results obtained with YOLO360 when compared with those of YOLOv3 for SORT is 75.71 MOTA and 68.31 MOTP and 42.44 MOTA and 73.98 MOTP. For DEEP SORT, scores of 77.38 MOTA and 68.58 MOTP with YOLO360 and 47.86 MOTA and 68.52 MOTP with YOLOv3 are obtained.

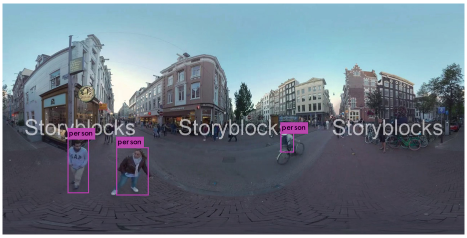

Fig. 4. Detection result for person class.

The reason behind the high accuracy of 'rope walk' video is the nature of the video, with least motion and no occlusion. This video can be considered as an ideal condition for tracking algorithm.

Table 1. MOT results of tracking algorithms:

			SORT		DEEP-SORT	
VIDEOS	FRAMES	Architecture	MOTA	MOTP	MOTA	MOTP
Hawaii-Beach	500	YOLOv3	42.44	73.98	47.86	68.52
Hawaii-Beach		YOLO 360	64.45	60.10	67.86	69.66
Rope Walk	2000	YOLO 360	93.47	77.66	91.31	67.91
Person-Florida	300	YOLO 360	69.2	67.18	72.80	68.17

4 Conclusions

In this paper, we have proposed a robust algorithm for 'Person' and 'Car' tracking in 360° videos using efficient multiple object detection and a variant of DEEP SORT. The main challenges are limited dataset and geometric distortions in the 360° video frame. We have proposed an CNN based YOLO360 architecture to detect 'Person' and 'Car' objects in 360° videos using transfer learning approach on YOLOv3 to train YOLO360. We create dataset for 360° videos for objects of 'Person' and 'Car' class by annotating number of frames in Equirectangular frames. We also use the motion and appearance model of DEEP SORT for tracking. We demonstrated our proposed detection and tracking algorithms using a number of 360° videos and compared our results with SORT and DEEP SORT algorithms and achieved an average improvement of 5.42% increase in MOTA.

Acknowledgement. This work was supported and mentored by Samsung R&D Institute Bangalore, India.

References

1. Sujatha, C., Chivate, A.R., Ganihar, S.A., Mudenagudi, U.: Time driven video summarization using GMM. In: 2013 4th National Conference on Computer Vision, Pattern Recognition, Image Processing and Graphics, IIT Jodhapur, pp. 1–4 (2013)
2. Sujatha, C., Mudenagudi, U.: Gaussian mixture model for summarization of surveillance video. In: 2015 5th National Conference on Computer Vision, Pattern Recognition, Image Processing and Graphics, IIT Patna, pp. 1–4 (2015)
3. Tabib, R.A., Patil, U., Ganihar, S.A., Trivedi, N., Mudenagudi, U.: Decision fusion for robust horizon estimation using dempster shafer combination rule. In: 2013 4th National Conference on Computer Vision, Pattern Recognition, Image Processing and Graphics, NCVPRIPG 2013, IIT Jodhapur, pp. 1–4 (2013)

4. Lampert, C.H., Blaschko, M.B., Hofmann, T.: Beyond sliding windows Object localization by efficient subwindow search. In: 2008 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8 (2008)
5. Harzallah, H., Jurie, F., Schmid, C.: Combining efficient object localization and image classification. In: 2009 IEEE 12th International Conference on Computer Vision (2009)
6. Wang, J., Yu, K., Lv, F., Gong, Y., Huang, T., Yang, J.: Locality-constrained linear coding for image classification. In: 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). https://doi.ieeecomputersociety.org/10.1109/CVPR.2010.5540018
7. Yang, J., Yu, K., Gong, Y., Huang, T.: Linear spatial pyramid matching using sparse coding for image classification. In: 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPR Workshops) (2009). https://doi.ieeecomputersociety.org/10.1109/CVPR.2009.5206757
8. Reid, D.: An algorithm for tracking multiple targets. IEEE Trans. Autom. Control **24**, 843–854 (1979)
9. Yilmaz, A., Javed, O., Shah, M.: Object tracking: a survey. ACM Comput. Surv. **38**, 13 (2006). https://doi.org/10.1145/1177352.1177355
10. Peterfreund, N.: Robust tracking of position and velocity with Kalman Snakes. IEEE Trans. Pattern Anal. Mach. Intell. **21**, 564–569 (1999)
11. Bewle, A., Ge, Z., Ott, L., Ramos, F., Upcroft, B.: Simple online and realtime tracking. CoRR (2016) https://dblp.org/rec/bib/journals/corr/BewleyGORU16
12. Wojke, N., Bewley, A., Paulus, D.: Simple online and realtime tracking with a deep association metric. CoRR (2017). https://dblp.org/rec/bib/journals/corr/WojkeBP17
13. Bernardin, K., Stiefelhagen, R.: Evaluating multiple object tracking performance: the CLEAR MOT metrics. EURASIP J. Image Video Process. (2008). https://doi.org/10.1155/2008/246309
14. Lin, T.-Y., et al.: Microsoft COCO: common objects in context. CoRR (2014). https://dblp.org/rec/bib/journals/corr/LinMBHPRDZ14
15. Frey, B.J., Dueck, D.: Clustering by passing messages between data points (2007)
16. Gabriel, P., Verly, J., Piater, J., Genon, A.: Proceedings of Advanced Concepts for Intelligent Vision Systems (2014)
17. Redmon, J., Farhadi, A.: YOLOv3: an incremental improvement. CoRR (2018). https://dblp.org/rec/bib/journals/corr/abs-1804-02767
18. Pan, S.J., Yang, Q.: A survey on transfer learning. IEEE Trans. Knowl. Data Eng. **22**, 1345–1359 (2010)