Semi-Automatic Multi-Object Video Annotation Based on Tracking, Prediction and Semantic Segmentation

Jaime B. Fernandez
Insight Centre for Data Analytics
Dublin City University
Dublin, Ireland
jaime.fernandezroblero5@mail.dcu.ie

Venkatesh G. M.
Insight Centre for Data Analytics
Dublin City University
Dublin, Ireland
venkatesh.gurummunirathnam2@mail.dcu.ie

Dian Zhang
Insight Centre for Data Analytics
Dublin City University
Dublin, Ireland
dian.zhang@dcu.ie

Suzanne Little
Insight Centre for Data Analytics
Dublin City University
Dublin, Ireland
suzanne.little@dcu.ie

Noel E. O'Connor
Insight Centre for Data Analytics
Dublin City University
Dublin, Ireland
noel.oconnor@dcu.ie

***Abstract*—Instrumented and autonomous vehicles can generate very high volumes of video data per car per day all of which must be annotated at a high degree of granularity, detail, and accuracy. Manually or automatically annotating videos at this level and volume is not a trivial task. Manual annotation is slow and expensive while automatic annotation algorithms have shown significant improvement over the past few years. This demonstration presents an application of multi-object tracking, path prediction, and semantic segmentation approaches to facilitate the process of multi-object video annotation for enriched tracklet extraction. Currently, these three approaches are used to enhance the annotation task but more can and will be included in the future.**

***Index Terms*—Video annotation, multi-object tracking, path prediction, semantic segmentation, traffic scenes.**

I. Video Annotation

Due to the portability, low cost and the richness of information given by video cameras, they have become a favorite sensor to provide information about a scene. They have been integrated into a wide variety of places and devices such as buildings, lamp posts, cellphones, robots, vehicles etc. Areas like security monitoring systems, robotics and, currently, autonomous vehicles use this type of sensor to get information about their surrounding environment.

Security monitoring systems and Autonomous Driver Assistance Systems (ADAS) are interested in using these videos to automate processes such as object recognition, event detection, accident avoidance or predicting future actions. Machine Learning along with Computer Vision approaches have shown high performance in this type of task. However, successful machine learning approaches generally require significant amounts of annotated data to learn.

Video annotation is a task that consists of manually labeling videos, frame by frame or by short sequences, indicating the objects, object type, tracks of the objects in the whole video, labeling the image regions, the action that is being performed, etc. Figure 1 shows a high-level view of an annotation pipeline.

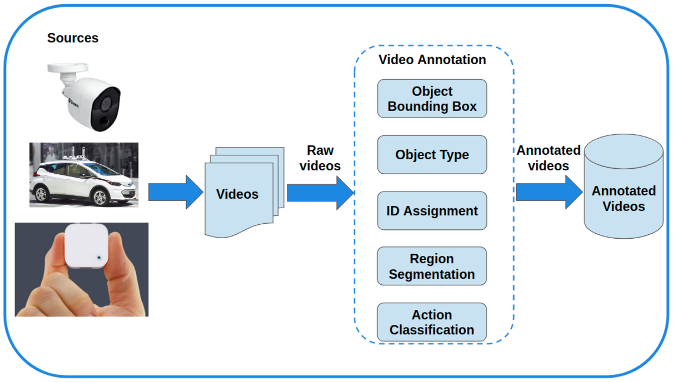


Fig. 1. General Video Annotation Framework

Some common labeling actions are:

- **Object detection:** provides the bounding boxes where an object is located, normally in the form of $(x1, y1, x2, y2)$ that correspond to the upper-left corner and bottom-right corner of the object's boundary.
- **Object tracking:** provides a specific ID to each object in a video, often across multiple frames. This ID allows tracks for each object to be constructed and to obtain the

This work has received funding from EU H2020 Project VI-DAS under grant number 690772 and Insight Centre for Data Analytics funded by SFI, grant number SFI/12/RC/2289.

978-1-7281-4673-7/19/$31.00 ©2019 IEEE

whole trajectory of the object in the entire video.

- **Region segmentation:** assigning a class label to the regions (as pixels) in an image. The class could be road, sky, vehicle, pedestrian, etc.
- **Action tagging:** higher level annotation, describing the actions that are happening in an image or a short sequence of video.

In machine learning and now with the increasing capability of deep learning approaches, a current trend is to use enriched data to achieve better performance. Enriched data means data with more features. For instance, an object on a scene can be labelled with the type of object, its past position, the region where this object is located, its orientation, semantic labels of the surroundings of the object, etc.

There are several commonly used datasets of video from instrumented vehicles such as MOT [1], KITTI [2], nuScenes [3], CityScapes [4] and Berkeley DeepDrive [5]. Some of these provide the bounding box of the objects on key frames, others give the semantic segmentation, while others, like KITTI, provide the tracklets of the objects in each frame. However, most of them only provide certain types of information about the objects in a independent frame-based (ie, non-tracked) manner and not as a whole.

II. Motivation

The motivation behind this demonstration is based on two interesting and related EU H2020 projects – Cloud-LSVA[1] (Large Scale Video Analytics) and VI-DAS[2] (Vision Inspired Driver Assistance Systems).

Cloud-LSVA is a project focused in Big Data, utilizing video data from automotive testing to address driver safety and support systems (ASAS) and advanced localisation/cartography as a data source to advance and scale video analytics. Whilst VI-DAS focuses on applying computer vision techniques to the 720° of video data gathered from instrumented vehicles (internal and external video footage) to identify and classify situations with the aim of improving handover in semi-autonomous vehicles.

III. Automation of Multi-Object Video Annotation

Video annotation is a process that requires a great amount of time and human resource. It is a tedious task that make the process harder for people and reduces the accuracy of the annotations. The aim of this demo is to facilitate the task of annotating, specifically for the process of multi-object trajectory extraction from videos.

A tracklet T is a set of tracks t that defines $t(x, y)$ the position of an object that travels a given space, $T = t1, t2, ...tn$. Each track is either a measure given by a sensor or manually annotated in ordered intervals of time, $t(x, y, time)$. Normally, a track only contains information of position $t(x, y)$, but currently there is a trend of including more information in each track. Information such as type of object, velocity, orientation, region where the object is located, surrounding of the object. This type of tracklets is known as *enriched* tracklets and allow for a more complete analysis of the trajectory of an object.

To facilitate the process of extracting enriched tracklets this demo combines three video analytics tasks: multi-object tracking, path prediction and semantic segmentation.

- **Multi-object tracking:** object tracking is one of the most important components in a wide range of applications in computer vision, such as surveillance, human computer interaction, and medical imaging. Given the initial state (e.g., position and size) of a target object in a frame of a video, the goal of tracking is to estimate the states of the target in the subsequent frames [6], [7].
- **Path prediction:** there are several applications where is useful to know what is going to happen next. In autonomous vehicles for example, a system should be able to anticipate what would happen to its environment in the future and respond to the change appropriately to avoid accidents or perform certain tasks. Path prediction refers to the process of predicting the near-future locations (generally within 5-25 frames or under 1 second) of an object based on its past trajectory [8].
- **Semantic segmentation:** this task refers to the process of labeling each region or pixel with a class of objects/non-objects. Semantic segmentation plays an important role in image understanding and essential for image analysis tasks. It has several applications in computer vision and artificial intelligence, autonomous driving, robot navigation, agriculture sciences, medical imaging analysis [9], [10].

Each approach was used to facilitate a different task – object ID assignment, future bounding box assignment and future bounding box constraints, figure 2.

- **Object ID assignment:** this module automatically assigns a unique and non-repeated ID to the objects labeled by the user. The user only needs to specify the region where each object of interest is located and the system will assign the corresponding IDs. This module leverages the IoU tracking scheme that uses mainly overlapping areas of the objects [11], [12] to automatically link the target object in corresponding frames.
- **Future bounding box prediction:** this module predicts the future position of the objects based on the labels given by the user and the automatic ID assignment. For the prediction task, this module uses the well known Kalman filter (KF) with the Constant Velocity (CV) model. The capability of the KF to adapt itself to the inputs given in real time, make it suitable for this module [13], [14].
- **Bounding box constraints:** this module constrains the predicted position of the objects based on a semantic map. The semantic map is created by leveraging state of the art semantic segmentation algorithms such as ENet [15].

IV. Demonstration

This section outlines the components that constitute the demo graphical user interface (GUI), Figure 3. The imple-

[1] https://cloud-lsva.eu/

[2] https://vi-das.eu

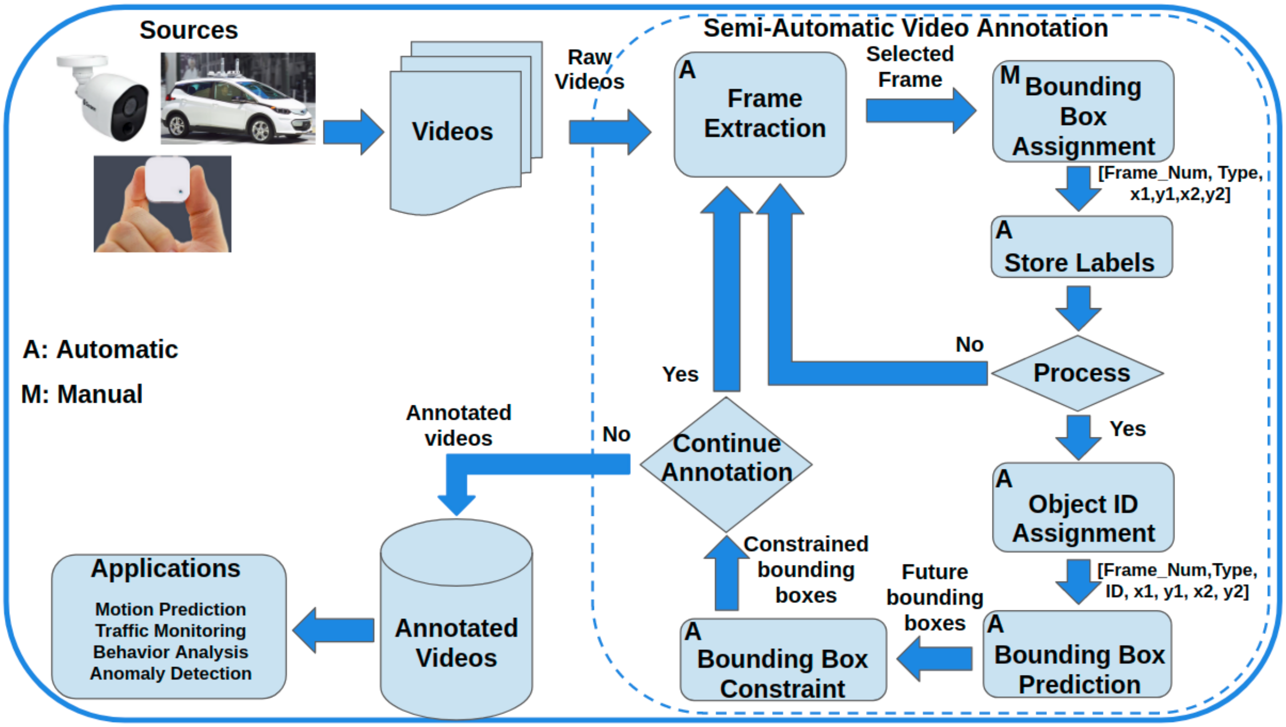
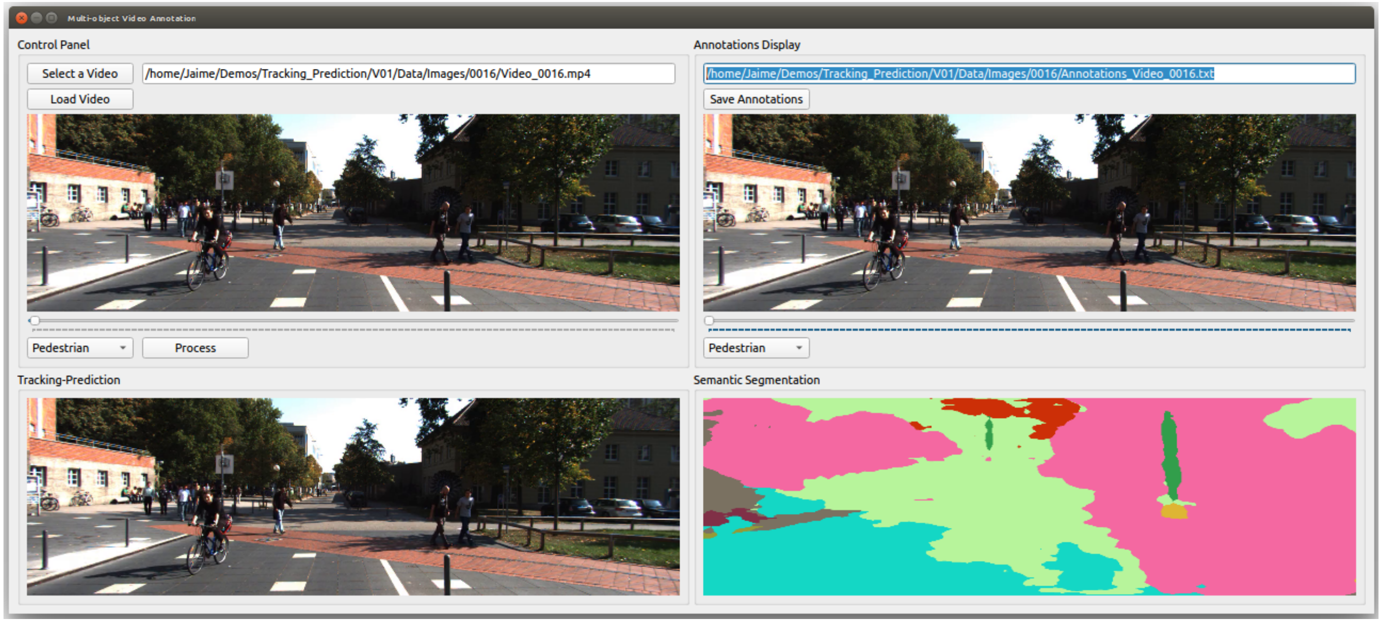


Fig. 2. Demonstration Approach Framework



/home/Jaime/Demos/Tracking_Prediction/V01/Data/Images/0016/Video_0016.mp4

/home/Jaime/Demos/Tracking_Prediction/V01/Data/Images/0016/Annotations_Video_0016.txt

Fig. 3. Demonstration Graphical User Interface

mentation was done in Python 3.5 using PySide2, which is the official Python module from the Qt for Python project (https://www.qt.io/) and OpenCV (https://opencv.org/).

The interface shows four components: Control Panel (top left), Annotations Display (top right), Tracking-Prediction (bottom left) and Semantic Segmentation (bottom right).

Control Panel is the first component and allows to the user to execute the following actions:

- **Select a video:** allows the user to browse through files and select the desired video.
- **Load video to the system:** loads the selected video to the system and makes it ready for processing frame by frame.
- **Explore video frame by frame:** this slider allows to

the user explore the video frame by frame. This permits the selection of a specific frame and running the semi-automatic annotation on it.

- **Label bounding boxes of objects:** the image on this component enables the user to draw bounding boxes around objects. The user only needs to select the area around the object of interest. These bounding boxes will be saved temporarily while the user interacts with the interface.
- **Select type of object:** along with the (x,y) position of the object of interest, the user can also indicate the type of object that is being labeled by selecting one of those already given in the drop-down menu.
- **Process:** execute the automatic annotation process. Based on the previous labeled objects given by the user, the tracking and the prediction algorithms assign to each object an ID and predict their future positions, constrained by the segmented map.

Annotations Display allow to the user to explore and see the annotations that have been made for the current video. The following actions can be executed on this component:

- **Save annotations:** this permits to save the labels to a file. For simplicity the labels are saved in a txt file, however this action can be easily modified to saved in a another more complex format such as XML or JSON.
- **Explore labels frame by frame:** this slider allows to the user explore the labels frame by frame. The labels are saved along with the number of frame where they belongs to, so when the user select a specific frame, the labels related to this frame are deployed on the image.
- **See labels by type of object:** this action permits to the user to see the labels by type of object.

Tracking-Prediction component shows the tracking and prediction results. These are the labels IDs created automatically by the tracker and the future position of the objects given by the predictor.

The final component called **Semantic Segmentation** shows the segmented map of the images being processed. This map is automatically generated by the semantic segmentation algorithm. Because only some areas are of interest, such as road, side walk, to create the constrain map some classes are grouped together and this way a simpler map is visualized.

This GUI will allow the user to interact with the system in real-time. The user will be able to select a video from a database such as KITTI [2] or CityScapes [4], to load it to the system and then, frame by frame, to draw bounding boxes around the objects of interest such as pedestrians and vehicles. Once the bounding boxes are provided and the user presses the Process button, the system will assign IDs to the objects and predict their future positions. The GUI will present the results in real-time and the user will be able to save the resulting annotations. The annotation is semi-automatic since the system takes as input the labeled bounding boxes provided by the user and produces the Object IDs and the future bounding box positions.

V. Conclusion

This demo presents a framework to annotate videos for multi-object tracklets. The objective of this framework is to facilitate the annotation task by automating some processes. This work makes use of three different but related approaches that have achieved good performance on specific tasks: multi-object tracking to automate the object ID assignment; path prediction to provide possible future bounding boxes and semantic segmentation to constrain the predicted bounding boxes. The demo was built using Python 3.5, PySide2 and OpenCV. Future work is needed to leverage more techniques such as instance segmentation and body-pose estimation.

References

[1] A. Milan, L. Leal-Taixé, I. Reid, S. Roth, and K. Schindler, "Mot16: A benchmark for multi-object tracking," *arXiv preprint arXiv:1603.00831*, 2016.

[2] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2013.

[3] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multimodal dataset for autonomous driving," *arXiv preprint arXiv:1903.11027*, 2019.

[4] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 3213–3223.

[5] H. Xu, Y. Gao, F. Yu, and T. Darrell, "End-to-end learning of driving models from large-scale video datasets," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2174–2182.

[6] Y. Wu, J. Lim, and M.-H. Yang, "Online object tracking: A benchmark," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2013, pp. 2411–2418.

[7] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *Acm computing surveys (CSUR)*, vol. 38, no. 4, p. 13, 2006.

[8] S. Lefèvre, D. Vasquez, and C. Laugier, "A survey on motion prediction and risk assessment for intelligent vehicles," *ROBOMECH journal*, vol. 1, no. 1, p. 1, 2014.

[9] F. Lateef and Y. Ruichek, "Survey on semantic segmentation using deep learning techniques," *Neurocomputing*, 2019.

[10] H. Zhu, F. Meng, J. Cai, and S. Lu, "Beyond pixels: A comprehensive survey from bottom-up to semantic image segmentation and cosegmentation," *Journal of Visual Communication and Image Representation*, vol. 34, pp. 12–27, 2016.

[11] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple online and realtime tracking," in *2016 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2016, pp. 3464–3468.

[12] E. Bochinski, V. Eiselein, and T. Sikora, "High-speed tracking-by-detection without using image information," in *2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. IEEE, 2017, pp. 1–6.

[13] N. Thacker and A. Lacey, "Tutorial: The likelihood interpretation of the kalman filter," *TINA Memos: Advanced Applied Statistics*, vol. 2, no. 1, pp. 1–11, 1996.

[14] R. Faragher *et al.*, "Understanding the basis of the kalman filter via a simple and intuitive derivation," *IEEE Signal processing magazine*, vol. 29, no. 5, pp. 128–132, 2012.

[15] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello, "Enet: A deep neural network architecture for real-time semantic segmentation," *arXiv preprint arXiv:1606.02147*, 2016.