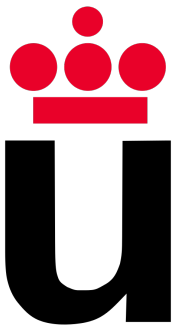


Master's thesis

An hybrid architecture for multiple people tracking

Marcos Pieras Sagardoy

Móstoles, 27 de Junio de 2016

Contents

1 Introduction

Tracking

Tracking is the analysis of video sequences for the purpose of establishing the location of the target over a sequence of frames.

2 Objectives

3 Theoretical basis

3.1 Optical flow

3.2 Deep learning

3.2.1 Object detection

Deep learning allows computational models that are composed of multiple processing layers to learn representations of data with multiple levels of abstraction. Traditional methods relay in hand-crafted (also called engineered) features to build a representation of the instance, and then learn a predictor to inference some label about it. These features are made by an expert to highlight the parts that they thought it could be more discriminative. Later on, the community applied the kernel methods, this methods try to find discriminability increasing the dimensionality of the features. In contrast, deep learning techniques, features are learned from the raw data and also are deeper than the hand-crafted features and kernels, therefore allows to compute more intrinsic properties. In the figure 1 we summarizzes the differences.

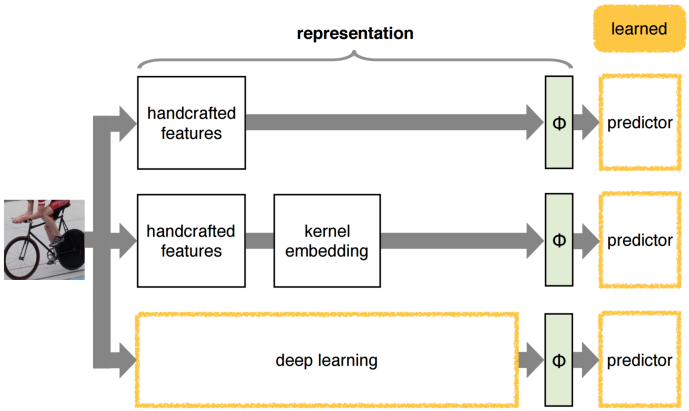


Figure 1: Comparision between traditional and deep learning methods.

Although, the studies of neural networks begun in the sixties [19], they didn't get relevance till 2012, when AlexNet [14] was published. The emergence of these techniques was dued

by three aspects:

- The increasing size of the dataset. Neural networks need a lot of data to get trained properly.

- The increasing of computing capabilities. Neural networks need a lot of computing capabilities to get trained well.

- Optimization details. Till recent years, we did not know the details to trained well the networks. For instance, these details are activations functions (ReLu, LeakyRelu), to no overfit (dropout), heuristic optimization techniques (adam, SGD).

Deep learning techniques has revolutionized the field of machine learning, specially Computer vision. For instance in image recognition, convolutional neural networks supposed a breakthrough in this tasks. As we can observe in figure 2, in 2012 with the AlexNet architecture **??**, it cut in a half the previous error in this contest.

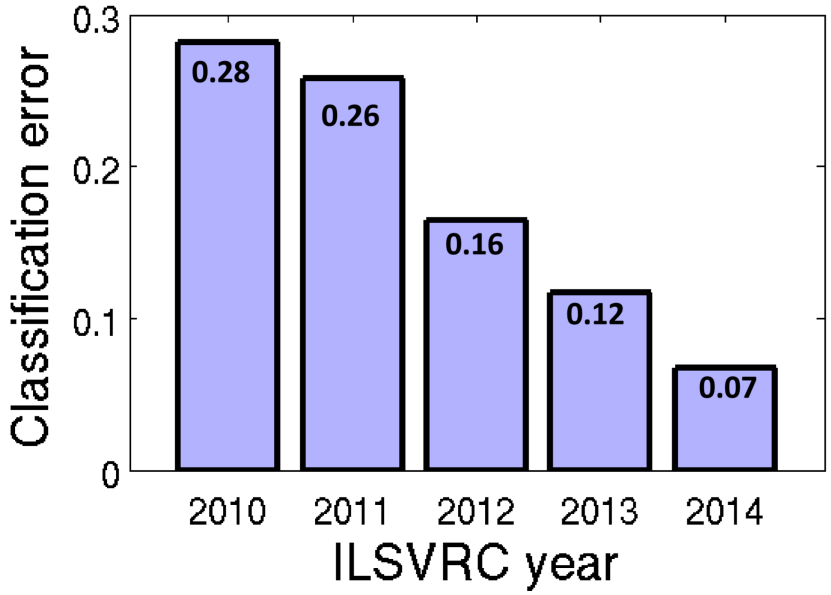


Figure 2: Classification error in IMAGENET.

In object detection as well, the emergence of the neural networks has supposed a turning point. As we can observe in 3, the mean average precision, has almost doubled with the use of neural networks.

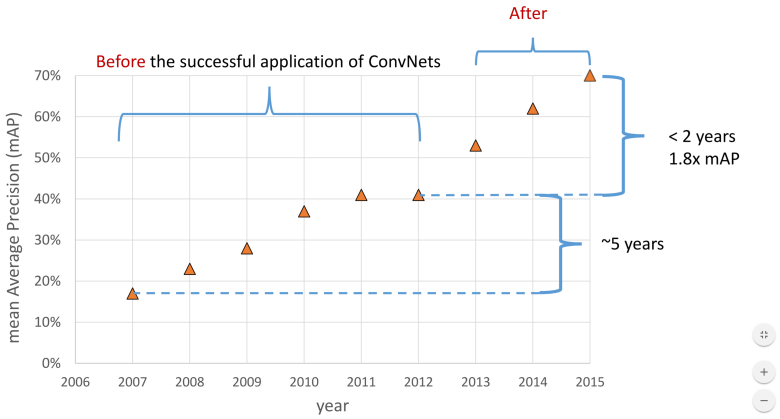
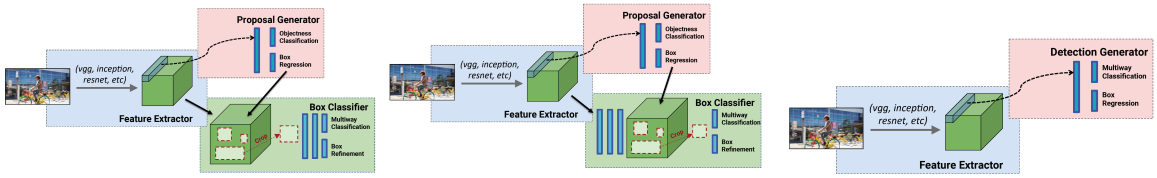


Figure 3: Mean average precion over the years in PASCAL dataset.

Actual object detectors are based on three main family of archictectures [11], of which names are: FasterRCNN, RFCN, and SSD. In 4 we can observe a scheme of these systems.



(a) Faster RCNN. (b) RFCN. (c) SSD.

Figure 4: Object detection architectures.

- Faster RCNN [22], it is the last output of a tryology of detectors developed by R. Girshick and his team. Which are called Region-Based object detecors. They work as follows: Use some mechanism to extract region of an image that probable to be an object and then classify those proposals with a CNN. The first paper to do so, was [7], and supposse a breaktrough in the field, increasing the precision of the state of the art of those days. But, it had a messy pipeline, slow and difficult to train. Later on, they developed [6], in this paper they applied the region proposal algorithm in the cnn feature map, so, they avoid to compute the features for each proposal. They increase the speed and it could be trained much easily. Finally, they showed FasterRCNN, in this algorithm, they eluded the external region proposal algorithm and they implemented a CNN to compute those proposals. This CNN share parameters with the main net and they saved a lot of time. This network, has become the standard object detector with CNN. With the association of novel net architecture like ResNet [9], Inception [26], and [13] they have won all the contests.
- RFCN [1], it stands for Region-based fully convolutional network. It is based on a Region-based architecture, altough after the feature map they add an region-sensitive layer, this layer activates in the position where there are an object. It has an inspiration in the architectures fully convolutional used in semantic segmantic.
- SSD, it stands fot Single shot multibox detector. These family of mehtods differs from previous ones considering that these treats the problem of object detection as regression. So, they are called Regression-based object detector or single shot

object detector due it does not have a regresion proposal algorithm, they classify the image with one mechanism. The maximium exponent of these algoirthms are [21] and [17]. These work as follows, they discretize the image in a fixed grid and for each grid it predicts a class, additionally for each single grid it predict some number of bounding boxes with different shape and sizes. It merges all, and apply a Non-Maximum supression algorithm and obtain a detections. We can observe this pipeline in 9. This is the case of YOLO algorithm, the SSD works the same but with a multiresolution scheme which allows to deal with small objects.

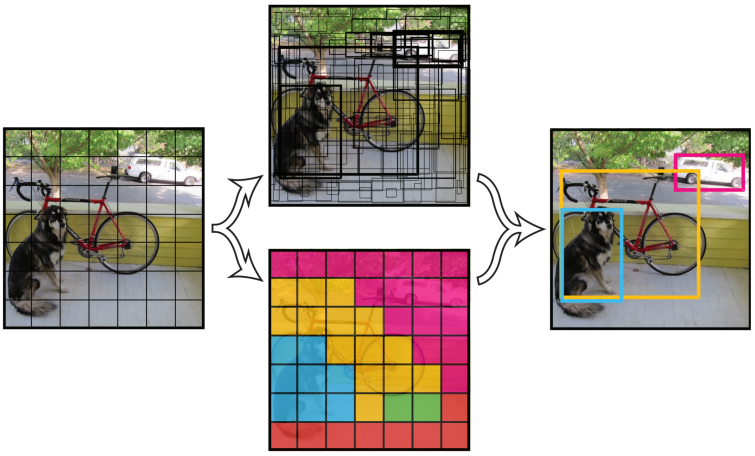


Figure 5: Regresion-based architecure.

In this survey [11], they compare the different methods including changing the features extractors (ResNet, Inception, VGG) and they measured the precision (mean average precision) and computing time. This results are showed in 6. The conlcusion are, SSD is fatest detector, RFCN it has the best balance between speed-acurracy and FasterRCNN, is the most acurate detector altough is slower than the other ones.

	mAP	*mAP_person*	*FPS*	*Proposals*
RCNN	66	64.2	0.077	2000
FastRCNN	70	69.9	6.7	2000
FasterRCNN	85.6	82.3	7	6000
SSD300	81.2	81.4	46	8732
SSD512	83.2	84.6	19	24564
YOLO	66.4	63.5	45	98
YOLOv2	78.6	81.3	40	-
RFCN	83.6	-	10	-
PVANET	84.9	-	31.3	300

Table 1: Results on VOC07.

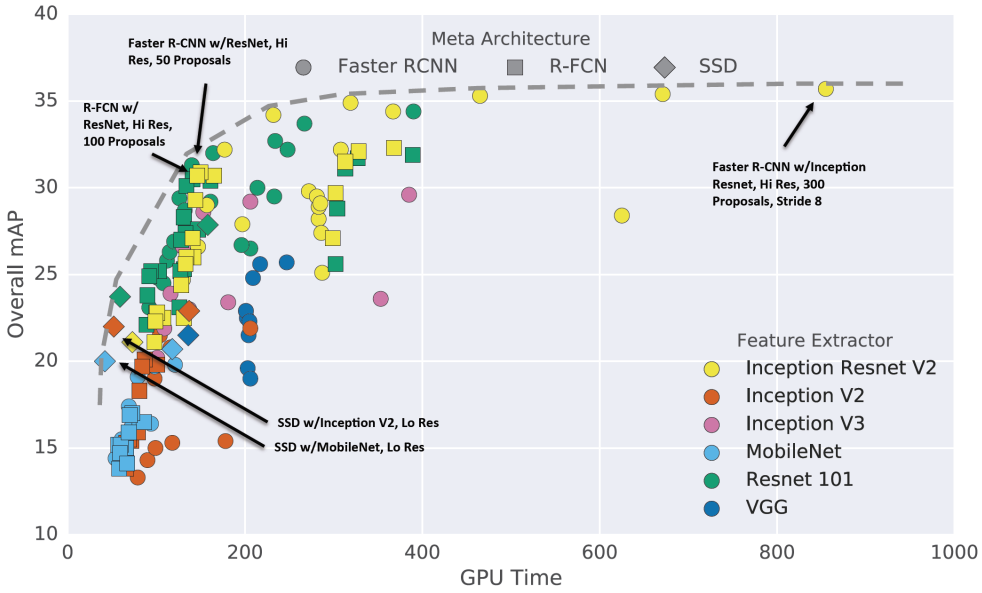


Figure 6: Comparision architectures.

We will finish off our review with numeric comparision of the methods, as we can observe in the table 1. This information is extracted from the original papers with their implementation, all of them are trained with theu union of the trainning set of VOC07, VOC12, and COCO, and subsequently evaluate on VOC07 test set and evaluate on a Nvidia Titan X gpu.

3.2.2 Tracking

As we said above, Deep learning techinques has increasded the perfomance all the aspects of computer vision.

Visual tracking go back to begings of the field of computer vision, before the introduction of neural networks, has been a clear distinction in the paradigms [24]:

- **Tracking using matching** (or apparence based methods), this trackers per-

form a matching of the representation of the target model built from the previous frame(s). The most prominent mehtods are those based on Optical flow, KLT **??** and MeanShift **??**.

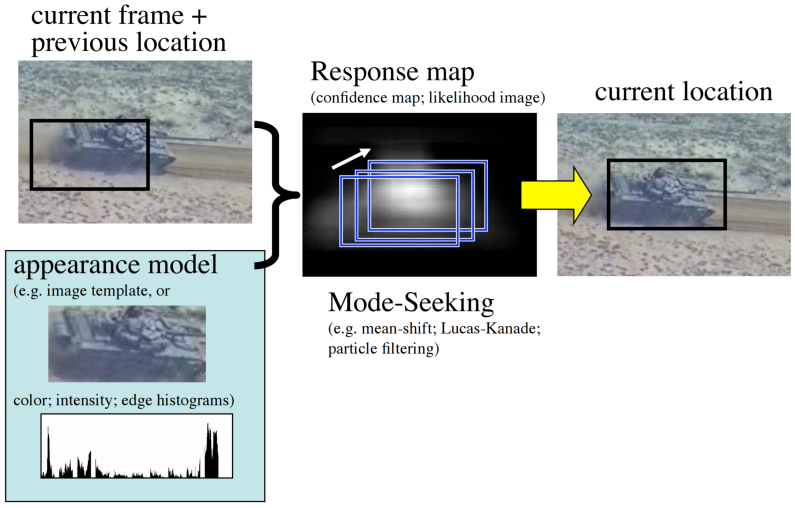


Figure 7: Regresion-based architecure.

- **Tracking using discriminative classification**. This way of perfoming tracking consist to build a model on the distinction of the target foreground against the background. Also called Tracking-by-detection, and the comunity has focused on how to update the representation of the model [12], [8] and how to solve the problem of data association, which consist to link detection across the frames [18] [29].

The deep learning techniques in tracking. [10]

- **Online training for tracking**. These methods use the same approach commented before, perform tracking by in an online manner update the classifier. A typical tracker will sample patches near the target object, which are considered as foreground. Some patches farther from the target object are also sampled, and these are considered as background. These patches are then used to train a foreground-background classifier, and this classifier is used to score patches from the next frame to estimate the new location of the target object. These methods showed a state-of-the-art perfomance results. Unfortunately, neural networks are slow to train. It can be improved by using what is called fine-tuning, only train the top layers which are most discriminative semantically, in addition it could take advantage of the large amount of annotated videos [20] [2].

- **Model based trackers**. These also are similar to the tracking by detection paradigm, these methods use a neural network to extract instances of the frames and then linked with temporal restrictions.

- **Siamese based tracking**. Similar to apparence matching tracking, many candidate are passed through the network, and the patch with highest similarity is selected as the tracking output [27].

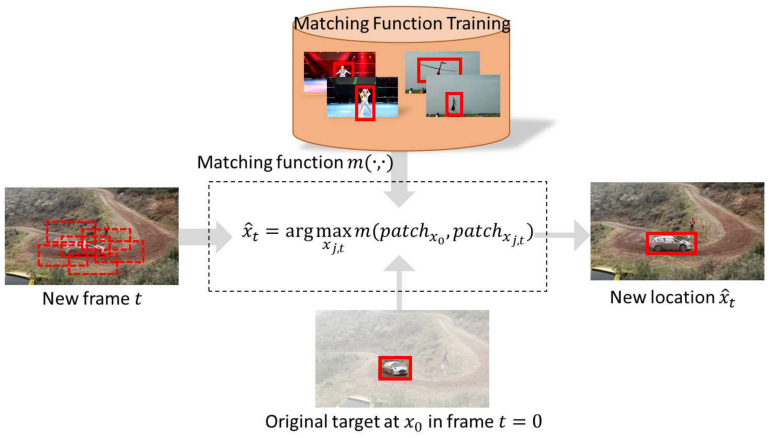


Figure 8: Regresion-based architecure.

- **Tracking as regression**, these methods are an extension of object localization using neural networks, those methods given an image containing an object predict the bounding box which contain the object.

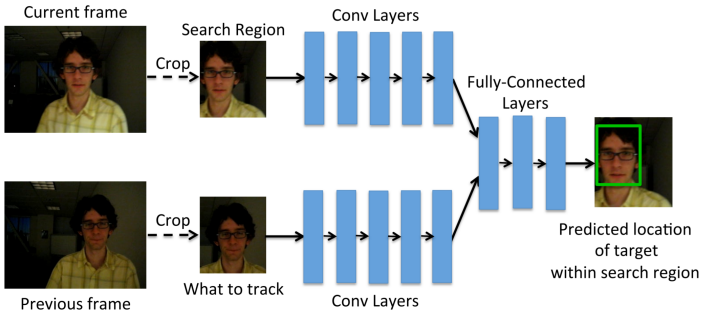


Figure 9: Regresion-based architecure.

- **Tracking with RNN**

4 Software implementation

5 Experiments

5.1 Datasets for object detection

This section describes the most common datasets used in object detection tasks.

Throughout the history of computer vision research datasets have played a critical role. They not only provide a means to train and evaluate algorithms, they drive research in new and more challenging directions. In order to accomplish this, they provide:

- a collection of challenging images and high quality annotation.
- an standard evaluation methodology, so the performance of the algorithms can be compared.

In the next subsections, we will explain the main datasets for object detection task.

5.1.1 Pascal Visual Objects Classes

The Pascal Visual Object Classes (VOC) challenge [4] is a benchmark in visual object category recognition and detection. Organised annually from 2005 to 2012, the challenge and its associated dataset has become accepted as one of the most benchmark for object detection. All the images are from the flickr consumer photographs website and annotated with the Mechanical turk tool.

The most popular editions of the challenge for object detection are those from years 2007 and 2012.

5.1.2 VOC07

The challenge of the year 2007, it contains 10 thousand images in the trainval and test sets, with almost 12 thousand objects. This was one the first datasets for object detection before the era deep learning. Also, it is very useful for researchers, due it has 2.5 mean object per image and it is very challenging. In the figure 10 we can observe the distribution of images and objects instances.

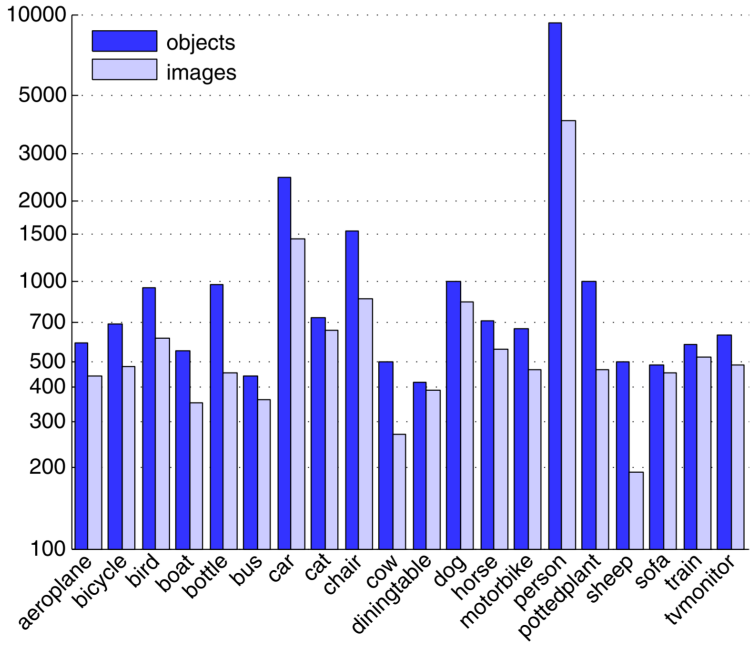


Figure 10: Distribution of VOC07 dataset.

5.1.3 VOC12

The 2012's edition is also, one of the most used dataset in object detection tasks. It increases the volume of images of the 2007 edition up to 10 thousand images on trainval and test set and similar quantity of instances per image.

5.1.4 ImageNet

ImageNet project [3] with the challenge ImageNet Large Scale Visual Recognition Challenge [ILSVRC] was the first large-scale database, temporally developed to supply the deep learning techniques, eager of feed with tons of images. ImageNet aims to populate the majority of the 80000 synsets of WordNet with an average of 500-1000 clean and full resolution images. The collection was based on the query of that words on several image search engines and human refined on the Amazon Mechanical Turk platform.

In 2016, the project collects more than 10 million of annotated images with 1000 classes. Although its main purpose is image classification, it has an object detection challenge with 200 categories with over a 1 million images with objects annotated.

5.1.5 COCO

The Microsoft Common Objects in Context also known as COCO dataset [16], is a dataset that address the three core research problems in scene understanding:

- detecting non-iconic views of objects, for many dataset most of the objects have an iconic representation, they appear unobstructed , near the center of the photo and with their canonical shape. So in this dataset, they included images to struggle the object recognition task, like objects in the background, partially occluded, amid clutter. Therefore, reflecting the composition of actual everyday scenes.
- contextual reasoning between objects, nowadays natural images contain multiple objects, and their identity can only be solved using context, due to small size or ambiguos appearance in the image, so in this dataset, images contain scenes rather isolated objects.
- the precise 2D localization of objects, also the detailed spatial understanding of object layout will be a core component of an image understanding system, so this dataset struggle to do so.

So, the three main tasks of this challenge are object classification, object detection and semantic scene labelling. This dataset contains 91 object categories, with 2.5 million labelled object instances in 328 thousand images, labeled with the Amazon Mechanical Turk tool.

5.1.6 Dataset comparison

The datasets from Pascal challenge are very useful to test object detection algorithms, their quantity is very handy (a few thousands of images) and contains a challenging quantity of objects per image, very interesting for the algorithms. But its little amount

of images does not permit to train a network on this dataset, although it can be used to finetune the network.

The datasets for the ImageNet challenge are not used to much in object detection tasks, it contains a few instances per image, this not encourage researcher to use it. Although, it is very utilize to extract features and then finetune your net.

The COCO datasets, is the most recent one, is the one focus on object recognition, and the detection suppose a challenge due the objects are in common places and are very challenging to detect. And it is very interesting to due of the quantity of instances per image.

The COCO challenge contains 91 object categories with 82 of them having more than 5 thousand labeled instances. In total the dataset has 2.5 million labeled instances in 328 thousand images. In contrast to ImageNet dataset, COCO has fewer categories but more instances per category. Also, it has more instances per category than the VOC dataset. We can observe that difference in the chart 11. This fact aid in learning detailed object models capable to chance the variability and also its 2D location.

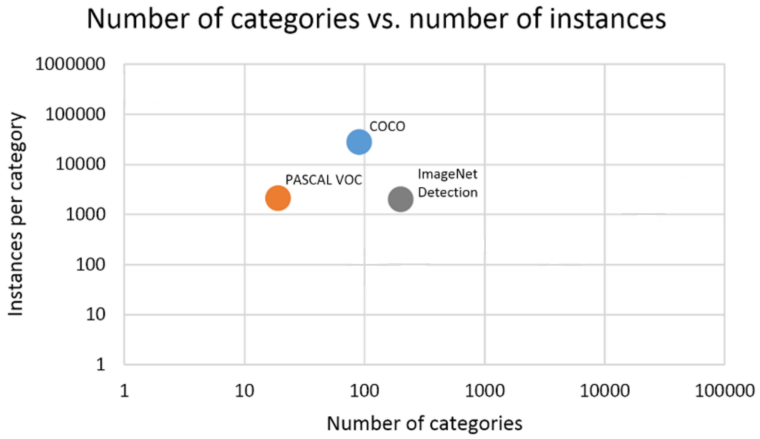


Figure 11: Distribution of pascal.

In addition, another prominent feature of the COCO over the other two, is the number of labelled instances per image which may aid in learning contextual information. This difference can be seen in 12 chart.

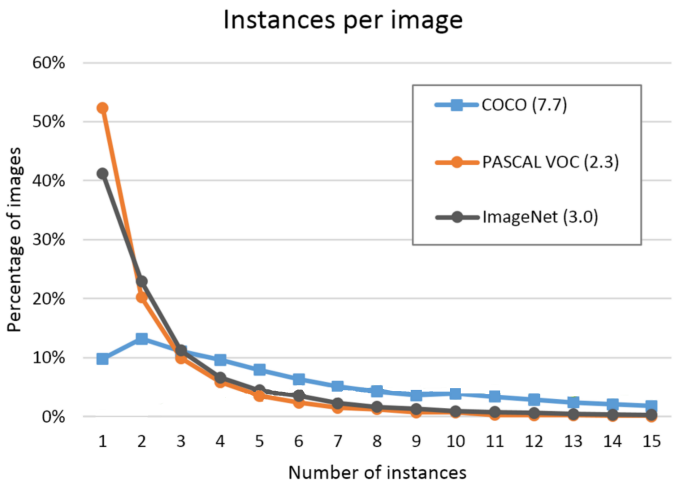


Figure 12: Distribution of pascal.

Moreover, the COCO dataset uses images from non-canonical point view, allowing to the algorithm to be robust to everyday views. This feature can be observed in the plot 13, in this plot we can observe differences views of the same category. And clearly the coco's images is the most uni-conic representation.

(a) Pascal VOC. (b) ImageNet. (c) COCO.

Figure 13: Distribution of pascal.

Finally, the table 2 summarizes the main statistics of the dataset stated previously.

	VOC07	**VOC12**	**ImageNet [2014]**	**Coco [2015]**
trainval set	5011	11540	476688	165482
test set	4952	10991	40152	81434
Number of classes	20	20	200	80
Mean obj per image	2.5	2.4	1.1	7.2
Number person instances	4690	8566	-	300000

Table 2: Datasets tables

5.1.7 Evaluation of object detection algorithms

In order to compare the performance of the different datasets, each challenges establish a clear measure. In this thesis, we used the interpolated average precision (AP), used in

the Pascal VOC challenge (based on [23]).

For each class, the precision/recall curve is computed from a method's ranked output.

- Recall, is defined as the proportion of all positives examples ranked above a given rank.
- Precision is the proportion of all examples above the rank which are from the positive class.

The AP summarises the shape of the precision/recall curve, and is defined as the mean precision at a set of eleven equally spaced recall levels [0,0.1,...,1]:

$$AP = \frac{1}{11} \sum_{r \epsilon (0,...,1)} p_{interp}(r)$$

The precision at each recall level r is *interpolated* by taking the maximum precision measured for a method for which the corresponding recall exceeds r:

$$p_{interp}(r) = max_{\hat{r}:\hat{r}>r} p(\hat{r})$$

The authors justified this measurement as a way to reduce the impact of the 'wiggles' in the precision/recall curve, caused by small variations in the ranking of examples. In the figure 14, we can observe this effect on the curve.

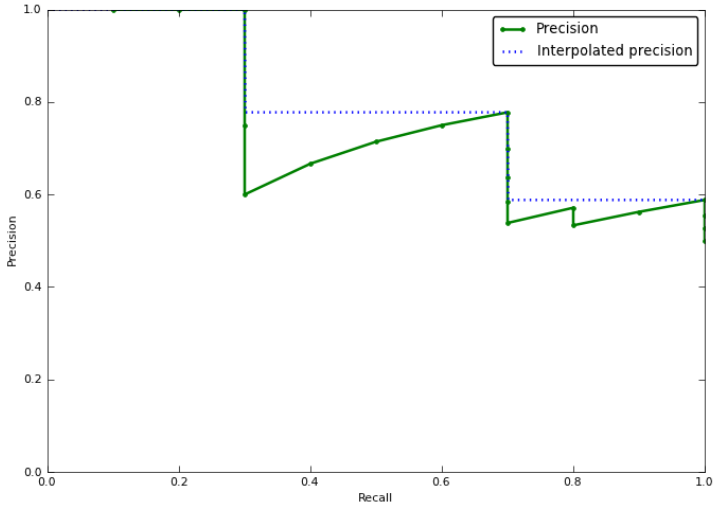


Figure 14: Comparision interpolated and normal curve.

In addition, detections were assigned to ground truth objects and judged to be true/false positives by measuring bounding box overlap. To be considered a correct detection, the area of overlap a_0 between the predicted bounding box B_p and ground truth bounding box B_{gt} must exceed 0.5 by the formula:

$$a_0 = \frac{area(B_p \cap B_{gt})}{area(B_p \cup B_{gt})}$$

where $B_p \cap B_{gt}$ denotes the intersection of the predicted and ground truth bounding boxes and $B_p \cup B_{gt}$ their union. The treshold of 50 % was set deliberately low to account for inaccuracies in bounding boxes in the ground truth data. Multiple detections of the same object in an image were considered false detections.

Finally, we want to point out, even we don't take into account in our implementation, setting the threshold IoU to a value of 0.5 could cause misdetections of small objects [3], they propose an adaptive setting of that threshold based on the size of the ground truth and so detect correctly small objects. In practice this change only affects 5.5% of objects in the detection validation set.

5.2 Datasets for multiple object tracking

Evaluating and comparing multi-target tracking methods is not trivial for numerous resasons.

- First, the perfect solution one aims to achieve is difficult to define clearly. Partially visible, occluded, or cropped targets, reflections, and objects that are very closely resemble targets; all impose intrinsic ambiguities, such that even humans may not agree on one particular ideal solution.

- Second, a number of different evaluation metrics with free parameters and ambiguous definitions often lead to inconsistent quantitative results across the literature.

- Finally, the lack of pre-defined test and training data makes it difficult to compare different methods fairly.

In contrast to other research areas in computer vision, still lacks large-scale benchmarks.

5.2.1 PETS

Targeted primarily at surveillance applications [5]. The 2009 version consisted of 3 subsets, S1 targeted at person count and density estimation, S2 targeted at people tracking, and S3 targeted at flow analysis and event recognition. In the 16 we can observe one image from this dataset.

Figure 15: Example of Pets.

Even for this widely used benchmark, we observe that tracking results are commonly obtained in an inconsistent fashion: involving using different subsets of available data, different detection inputs, inconsistent model training that is often prone to over-fitting, and varying evaluation scripts. Results are thus not easily comparable [15].

5.2.2 MOT challenge

The aim of the Multiple object tracking [MOT] is to standardize the use of multiple people trackings datasets, in order to do so, they solve the problems in this kind of dataset, explained above.

5.2.3 Evaluation of multiple people tracking algorithms

A critical point with any dataset is how to measure the performance of the algorithms. A large number of metrics for quantitative evaluation of multiple target tracking have been proposed. Choosing unique general evaluation is still ongoing.

On the one hand, it is desirable to summarize the performance into one single number to enable a direct comparison. On the other hand, one might not want to lose information about the individual errors made by the algorithms and provide several performance estimates, which precludes a clear ranking.

We will explain two sets of measures that have established themselves in the literature the CLEAR metrics [25], and a set of track quality measures [28].

As in the object detection metrics, we can classify each tracket, whether it is a true positive, that describes an actual (annotated) target, whether the output is a false alarm (or false positive, FP). This decisions is typically made by the well-known thresholding measure of Intersection of the union [IoU]. Also a target that is mised by any tracker is a false negative.

Due to we are working with multiple object, we assume that each ground truth trajectory has one unique start and one unique end point, that is not fragmented. So we need to penalty re-identification. This is called, identity switch [IDSW], and it is counted as if a

ground truth target i is matched to track j and the last known assignment was $k = j$. The next figure summarizes the stated measures (the grey area indicate the the matching threshold).

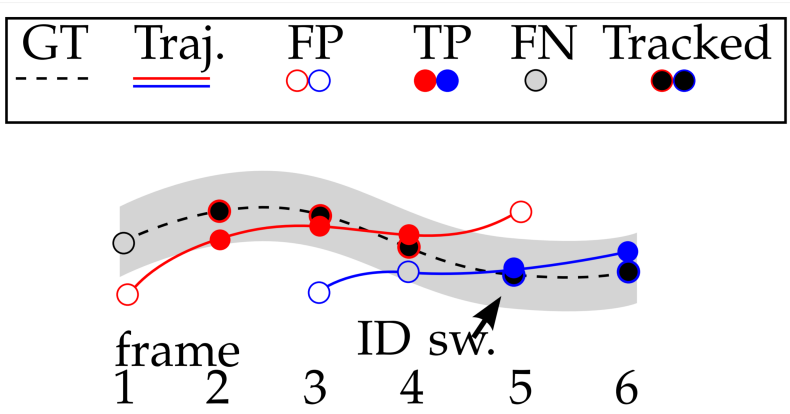


Figure 16: Example of measures.

Then after determining true matches and establishing the correspondandes it is now possible to compute the metrics over all the sequences.

The multiple object tracking accuracy [MOTA] [25] is perhaps the most widely used figure to evaluate a tracker's perfomance. The main reason for this is its expressiveness as it combines as it combines three sources of errors defined above:

$$MOTA = 1 - \frac{\sum_t (FN_t + FP_t + IDSW_t)}{\sum_t GT_t}$$

where t is the frame index and GT is the number of ground truth objects. This measures gives an indication of the overall performance.

The multiple object tracking precision [MOTP] is the average dissimilarity between all true postives and their corresponding ground truth targets. For bounding box overlap, that is computed as

$$MOTP = \frac{\sum_{t,i} d_{t,i}}{\sum_t c_t}$$

where c_t denotes the number of matches in frame t and $d_{t,i}$ is the bounding box overlap of target i with its assigned ground truth object. Thereby gives the average overlap between all correctly matched hypotheses. So, the MOTP is a measure of localization precision.

As we have stated above, another metrics are the track quality measures. Each ground truth trajectory can be classified as mostly tracked (MT), partially tracked (PT), and mostly lost (ML). This is done based on how much of the trajectory is recovered by the tracking algorithm. A target is mostly tracked if it is successfully tracked for at least 80% of its life span, without consider if there are an identity switch. If a track is only recovered for less than 20% of its total length, it is said to be mostly lost (ML). All other tracks

are partially tracked. Finally antoher quality measure is track fragmentations (FM), it counts how many times a ground truth trajectory is resumed at a later point.

6 Conclusions

6.1 Future work

References

[1] Jifeng Dai, Yi Li, Kaiming He, and Jian Sun. R-FCN: object detection via region-based fully convolutional networks. *CoRR*, abs/1605.06409, 2016.

[2] Martin Danelljan, Gustav Hager, Fahad Shahbaz Khan, and Michael Felsberg. Learning spatially regularized correlation filters for visual tracking. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015.

[3] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

[4] M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision*, 111(1):98–136, January 2015.

[5] J. Ferryman and A. Ellis. Pets2010: Dataset and challenge. In *Proceedings of the 2010 7th IEEE International Conference on Advanced Video and Signal Based Surveillance*, AVSS '10, pages 143–150, Washington, DC, USA, 2010. IEEE Computer Society.

[6] Ross Girshick. Fast r-cnn. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015.

[7] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[8] S. Hare, S. Golodetz, A. Saffari, V. Vineet, M. M. Cheng, S. L. Hicks, and P. H. S. Torr. Struck: Structured output tracking with kernels. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(10):2096–2109, Oct 2016.

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.

[10] David Held, Sebastian Thrun, and Silvio Savarese. Learning to track at 100 FPS with deep regression networks. *CoRR*, abs/1604.01802, 2016.

[11] Jonathan Huang, Vivek Rathod, Chen Sun, Menglong Zhu, Anoop Korattikara, Alireza Fathi, Ian Fischer, Zbigniew Wojna, Yang Song, Sergio Guadarrama, and Kevin Murphy. Speed/accuracy trade-offs for modern convolutional object detectors. *CoRR*, abs/1611.10012, 2016.

[12] Zdenek Kalal, Krystian Mikolajczyk, and Jiri Matas. Tracking-learning-detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1409–1422, 2012.

[13] Kye-Hyeon Kim, Yeongjae Cheon, Sanghoon Hong, Byung-Seok Roh, and Minje Park. PVANET: deep but lightweight neural networks for real-time object detection. *CoRR*, abs/1608.08021, 2016.

[14] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[15] Laura Leal-Taixé, Anton Milan, Ian D. Reid, Stefan Roth, and Konrad Schindler. Motchallenge 2015: Towards a benchmark for multi-target tracking. *CoRR*, abs/1504.01942, 2015.

[16] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. *CoRR*, abs/1405.0312, 2014.

[17] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott E. Reed, Cheng-Yang Fu, and Alexander C. Berg. SSD: single shot multibox detector. *CoRR*, abs/1512.02325, 2015.

[18] Anton Milan, Stefan Roth, and Konrad Schindler. Continuous energy minimization for multitarget tracking. *IEEE transactions on pattern analysis and machine intelligence*, 36(1):58–72, 2014.

[19] M. Minsky and S. Papert. *Perceptrons*. MIT Press, Cambridge, MA, 1969.

[20] Hyeonseob Nam and Bohyung Han. Learning multi-domain convolutional neural networks for visual tracking. *CoRR*, abs/1510.07945, 2015.

[21] Joseph Redmon, Santosh Kumar Divvala, Ross B. Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. *CoRR*, abs/1506.02640, 2015.

[22] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 91–99. Curran Associates, Inc., 2015.

[23] Gerard Salton and Michael J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc., New York, NY, USA, 1986.

[24] A. W. M. Smeulders, D. M. Chu, R. Cucchiara, S. Calderara, A. Dehghan, and M. Shah. Visual tracking: An experimental survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(7):1442–1468, July 2014.

[25] Rainer Stiefelhagen, Keni Bernardin, Rachel Bowers, John Garofolo, Djamel Mostefa, and Padmanabhan Soundararajan. *The CLEAR 2006 Evaluation*, pages 1–44. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007.

[26] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going

deeper with convolutions. In *Computer Vision and Pattern Recognition (CVPR)*, 2015.

[27] Ran Tao, Efstratios Gavves, and Arnold W. M. Smeulders. Siamese instance search for tracking. *CoRR*, abs/1605.05863, 2016.

[28] Bo Wu and R. Nevatia. Tracking of multiple, partially occluded humans based on static body part detection. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 1, pages 951–958, June 2006.

[29] Li Zhang, Yuan Li, and Ramakant Nevatia. Global data association for multi-object tracking using network flows. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.