Tracking based sparse box proposal for time constraint detection in video stream.

Adrien CHAN-HON-TONG* and Stephane HERBIN*
* ONERA - The French Aerospace Lab
F-91761 Palaiseau, France
firstname.name@onera.fr

***Abstract*—Search and Rescue or surveillance applications from en embedded moving camera yield challenging computer vision problems as both very high precision/recall and real-time performance are required. However, in these contexts, it is often sufficient to assess the detection of each object of interest only once in the area spanned by the camera, with the idea that what matters is to be aware of its existence, its time to detection being a secondary objective.**

Taking advantage of this point, we describe a sparse box proposal controlled by tracking and designed to generate a small number of boxes per frame covering each object at least once in the video. Our sparse box proposal adapts to the budget allowed for single box classification and is able to achieve relevant results even on challenging situations while comfortably dealing with real time requirements.

***Index Terms*—Object detection, Real time, Search and rescue, Moving camera, Box proposal, Tracking**

I. Introduction

The state of the art of image classification has made spectacular progress recently on several benchmarks (ILSVRC and Pascal VOC) thanks to the use of so called deep leaning approaches [1], [2].

However, the direct application of such techniques to a naive multi-scale exhaustive sliding window approach, as is often used for object detection, is not tractable: the number of windows to process is of the hundred of thousand order while Caffe [3], one of the best available deep leaning implementation, reports a 800 windows per second classifier on a powerful hardware and with a specific data storing strategy. Thus, one of the key ingredient to the success of deep learning applied to object detection is to rely on a box proposal capacity [4] that bounds the number of windows to classify as it is proposed in [2].

However, even when using box proposal, deep learning classifiers are too slow for generic real time detection applications. First, because box proposal itself is not real time [2], secondly, because to achieve real time speed with a 800 windows per second classifier, one can only evaluate 32 bounding boxes per frame (at 25 frames per second) while the highest performance published in [2] requires 2000 boxes per frame.

One solution to increase processing speed is to rely on faster classifier/detector like [5], [6] at the expense of degrading the performances.

Other solutions can be tailored for specific applications. For example, in some application, objects of interest can only appear in the video from specific areas in a frame allowing the use of a strong but slow detector only on these areas.

In this paper, we describe another alternative which can be applied to any type of video content but is specific to the goal of a video processing chain used for example in Search and Rescue operations. One of the useful functions in this context is to geographically locate very rare but critical objects during a fly over a given area (e.g. a person lying or wandering in a devastated area). This task clearly implies to be able to detect predefined categories of objects in the video stream, that may appear anywhere in the frame and any time in the stream as it may have been previously occluded by the environment (e.g. occluded by trees or hills). However, unlike classical detection scenario there is no need to detect the object in each frame of the stream: one detection of the object from the entire video is enough.

Our solution to exploit high end but costly classifiers applied to video stream analysis is a box proposal function adapted to the goal of detecting every object at least once in the video (this can be generalised from *once* to *a small fixed number of times*). The key constraint of the task is that the number of boxes per frame M should be two orders of magnitude less than the number of boxes considered in [2]. The key advantage is that there is no need to detect the objects in all the frames.

II. Metric

These constraint/advantage are not captured either by standard detection metrics nor by tracking metrics. The detection ground truth is organized as a set of tracks. The goal is to produce boxes that sufficiently overlap the ground truth boxes. To quantify the overlap, it is common to expect an Intersection over Union ratio (IoU, sometimes called Jacard ratio) above a given threshold $\alpha = 0.5$. By comparison to standard detection metrics, only one good overlap by track, or a small fixed number of good overlaps by track, is required, with no requirement on the overlapping dates.

Formally, let assume that the video has T frames and contains a set of objects O. We note $\forall o \in O$, $b_t^*(o)$ the bounding box of the object o in frame t (when present). Let B a set of boxes with B_t the set of boxes extracted from frame t. Let $|.|$ denote the cardinal and $[t', t'']$ the set of integer between

978-1-4799-8637-8/15/$31.00 ©2015 IEEE

t' and t''. The classification budget constraint on B is that: $\forall t \in [1,T]$, $|B_t| \leq M$. Given these assumptions/notations, the recall rate of B on O is:

$$|\{o \in O / \exists t \in [1,T], b \in B_t / \text{IoU}(b, b_t^*(o)) \geq \alpha\}| \quad / \quad |O|$$

with $IoU(b, b_t^*(o)) = -\infty$ when o is not present at frame t. For simplicity, the denominator $|O|$ will be omitted in the following.

This metric gives a simple view of the problem that we tackle here: the ground truth is a set of tracks, the system output is a set of boxes and the introduced recall measure the number of tracks for which there is, at least, one overlap.

A naive application of a single frame box proposal, i.e. a box proposal applied independently on each frame, with small M should lead to a low recall rate: boxes will densely cover only the most salient objects, which are likely to be the same on two consecutive frames, and not sparsely and uniformly cover all the objects.

We propose in this article an algorithm able to take into account the sequential nature of video data, based on single frame box proposal coupled with tracking functions, to deal with this specific goal. The algorithm is presented in section 4 after a review of related work in section 3. It is evaluated along with baselines in section 5 on a publicly available dataset of aerial videos, the VIRAT aerial dataset [7].

III. Related works

In terms of algorithmic core, the closest works to this paper are [8], [9]. In [9] a category-independent video object proposal is presented. This system is based on conditional random field framework and takes advantage of motion clues to group pixel according to their appearance and dynamic. The output of the system is a set of tracklets. [9] adapts the work of [4] by manipulating voxels instead of pixels. In [4], superpixels are extracted in each frame, and then, a hierarchical clustering of the superpixels using multiple clustering strategies leads to a set of proposed boxes. In [9], supervoxels are extracted from the video, and then, a hierarchical clustering of the supervoxels leads to the set of proposed tracklets.

However, in both papers, the objective is to form tracklets mainly for action recognition purpose: the idea is first to extract the spatio temporal area of the action and then to apply an action classifier. Let's notice that to recognize an action, a tracklet which is cut in the middle of the action may not be useful. Thus, these works are close to standard tracking ones (for example [10]) in term of objective as they want produce good tracklets.

In this paper, the objective is different: we want to propose good boxes for speeding up object detection. So, in terms of objective, this work is closer to [5], [6] which proposed real time detection algorithms and to single frame box proposal [11], [4], [12] (a more detailed review can be found in [13]) than to [8], [9].

In terms of context of application, this work is also close to moving object detection from UAV which contains a large literature ([14], [15] are representative examples). However, we do not limit the detection to moving objects because we want to be able to detect also non moving ones (like person lying down in forest) which can only be detected by image based systems.

IV. Tracking based sparse box proposal

A. Tracking is hard but we do weak tracking

Before describing mathematically our system, let's highlight the basic idea behind it. The first basic idea is to memorize previously proposed boxes so that new box proposals have an image content that differs from the previous ones. The second basic idea is to speed up the matching between new and previous boxes by performing tracking on previous boxes.

One could be suspicious about a system using *tracking* in order to proposed boxes whereas *tracking* is usually a harder problem than box proposal. However, we only need a weak tracking: tracking is simply used to save past proposals in a structured way but not to maintain identity over time. For example, we do not care about starting a track as soon as the object appears: there is no loss as long as we finally propose a box on one frame of the sequence of boxes representing the object locations. Again, we do not care if two tracks collide because, schematically, the only tag assigned to the tracks are *already proposed* and *not already proposed*: switching two *already proposed* tags is not a problem.

In other words, we will use tracking but the purpose is not to track. This is a major difference with [8], [9]: we may achieve our goal with weak tracking while they need a complete tracking able to maintain the identity of objects.

B. Derivation from the metric

We show below that this idea of using tracking is straightforward given the nature of the problem: let's consider the recall rate $S\left(B_{[1,\tau]}, O\right)$ (abbr S_τ) given by a set of boxes $B_{[1,\tau]}$ extracted from frames $t \in [1,\tau]$:

$$S_\tau = |\{o \in O / \exists t \in [1,\tau], b \in B_t / \text{IoU}(b, b_t^*(o)) \geq \alpha\}|$$

This recall rate can be decomposed into previous box decisions and future ones by

$$S_{\tau+1} = S_\tau + \left|\left\{o \in O / \text{Cover}\left(B_{[\tau+1,\tau+1]}, o\right) \wedge \neg \text{Cover}\left(B_{[1,\tau]}, o\right)\right\}\right|$$

where $\text{Cover}\left(B_{[t',t'']}, o\right)$ equals 1 if there is a date when one of the bounding boxes in B_t for $t \in [t', t'']$ overlaps the ground truth $b_t^*(o)$. When $B_{[\tau,\tau+1]}$ is reduced to only one element b_c, this last term becomes

$$\left|\left\{o \in O / \text{IoU}\left(b_c, b_{\tau+1}^*(o)\right) \geq \alpha \wedge \neg \text{Cover}\left(B_{[1,\tau]}, o\right)\right\}\right|$$

Now, at testing time, we do not know O but we can estimate a probability over O. Thus, we may want to maximize the

expected score:

$$E\left[S\left(B_{[1,\tau+1]},O\right)\right] = E\left[S\left(B_{[1,\tau]},O\right)\right] + P_{\tau+1}$$

where $P_{\tau+1}$ is:

$$P\left(\exists o \in O/\mathrm{IoU}\left(b_c, b^*_{\tau+1}(o)\right) \geq \alpha \wedge \neg\mathrm{Cover}\left(B_{[1,\tau]}, o\right)\right)$$

Thus, finding the box b_c at frame $\tau+1$ that maximizes the expected score corresponds to finding the box that maximizes the probability $P_{\tau+1}$. Now, we can decompose $P_{\tau+1}$ as the following product:

$$P\left(\exists o \in O/\mathrm{IoU}\left(b_c, b^*_{\tau+1}(o)\right) \geq \alpha\right)$$
$$\times P\left(\exists o \in O/\neg\mathrm{Cover}\left(B_{[1,\tau]}, o\right)|\mathrm{IoU}\left(b_c, b^*_{\tau+1}(o)\right) \geq \alpha\right)$$

The interesting point is that this decomposition brings up two probabilities that can be quickly (even if coarsely) estimated: the probability of containing an object

$$P\left(\exists o \in O/\mathrm{IoU}\left(b_c, b^*_{\tau+1}(o)\right) \geq \alpha\right)$$

which can be coarsely estimated by classic box proposal tools (let P_{box} be the estimate), and the probability of not containing an already tested object

$$P\left(\exists o \in O/\neg\mathrm{Cover}\left(B_{[1,\tau]}, o\right)|\mathrm{IoU}\left(b_c, b^*_{\tau+1}(o)\right) \geq \alpha\right)$$

which can be coarsely estimated (let $P_{\mathrm{not\ tracked}}$ be the estimate) by tracking tools using previous proposed boxes $B_{[1,\tau]}$.

Our system is straightforwardly derived from this last equation: for each frame, we compute the coarse estimates P_{box} and $P_{\mathrm{not\ tracked}}$. Then, we extract the box that maximize the estimated recall gain:

$$\arg\max_{b_c}\left(P_{\mathrm{box}}\left(b_c\right) \times P_{\mathrm{not\ tracked}}\left(b_c\right)\right)$$

we update $P_{\mathrm{not\ tracked}}$ to take into account the selection of this box and we loop M times on these two operations. This loop can be seen as a greedy search for a set of M boxes.

C. Implementation details

The purpose of our system is to speed up state of the art classification algorithm like [1] that are too slow for real time detection using sliding window or state of the art single frame box proposal. Thus, our system should be as fast as possible. So, among all box proposal only the BING algorithm [11] could really be considered for our task. The BING algorithm consists in using a very fast classifier on sliding window framework. The classifier at single scale and aspect ratio transforms an input region of interest into a 8x8 patch coding gradient information on which a linear SVM is applied.

Unfortunately, we did not manage to get 300 FPS from the code released by [11]. We believe that this is due to the fact that we do not target the same architecture (we found FPS comparable as the one reported by [13]). So we use a degraded version in our system: a simple convolution is applied on the gradient image of the input region of interest leading to an objectness score. The convolutionnal mask depends only on the box size: we start from a all-zero mask, fit a 1-value ellipse into the mask and apply spatial smoothing. We experimentally found this version faster than BING on our experiments. Off course, on datasets of high resolution image like the Pascal Challenge [16], using this algorithm instead of BING would have been at the price of an important loss of recall (based on auxiliary work, we found that our box proposal performs poorly on Pascal Challenge respectively to BING). But, in the context of this paper characterized by low resolution videos, this simple box proposal gives results of sufficient quality (see table I).

The tracking module should also be real-time while dealing with a high number of tracks (typically between 100 and 400 as each selected box becomes a target that live until a tracking failure is suspected). This makes us rely on a fast block matching approach instead of more sophisticated state of the art tracking approaches.

In practice, our estimate $P_{\mathrm{not\ tracked}}$ is binary. So, in each frame, all the boxes from the current frame that overlap a living track are discarded. Then for M times, we select the non discarded box which is top ranked for P_{box} and we initialize a new track at this location (updating $P_{\mathrm{not\ tracked}}$).

V. Experiments

A. Dataset

We perform a set of experiments on a subset of the VIRAT aerial dataset [7] in which the algorithms searches people.

The VIRAT aerial dataset is very challenging as videos are low resolution and contain camera motions, highly textured objects that are not person (car, tank, building), very small person, changes of plan, change in zoom, and even change in channel (sometimes infrared images, sometimes color images).

As no public annotation has been released for this dataset, we annotated a subset of the frames. In order to provide a diversity of situations, we chose to annotate about thirty sub videos of around 400 frames containing at least one person distributed over the dataset, but discarded infrared images. Figure 1 shows examples of images from the sub dataset. Notice that, in this experiment, we are not interested in the precision but only in the recall rate of the approach. This makes empty video irrelevant.

In some part of this dataset, manually localise the people in a single image is even challenging because of the low resolution. Annotation is still possible using motion information from the video but may lead to a degraded localisation. So the dataset has been manually annotated two times and averaged. In addition, to take into account this low resolution, we allow an Intersection over Union threshold of $\alpha = 0.4$, lower than usually used. But, we need to stress out that even the two manual annotations were not fully compatible with a $\alpha = 0.5$ ratio. We notice that, at this level of resolution, $\alpha = 0.5$ may correspond to *no more than 3x3 pixels of displacements*. Therefore a ratio of $\alpha = 0.4$ leads to operationally interesting problems.

Fig. 1. Illustration of the diversity of the selected subset of VIRAT.

Fig. 2. Illustration of the VIRAT annotations.

The annotation is a tracking like annotation: each person in each image is localized and associated with a temporally consistent id. We do not address reidentification so the id of a person changes each time this person is temporarily occluded (for example by a building). Manually tracking people is also challenging on the VIRAT aerial videos containing many low resolved similar people. When in doubt about an id, a new one is used.

Figure 2 shows two frames from the subset of VIRAT aerial dataset and corresponding ground truth.

Also, as the goal is a geographical localisation of the objects of interest, we manually provide a weak calibration of the camera.

B. Baselines

As we do not know of any other sparse box proposal applied to video analysis, we implemented some baselines to compare with our system.

frame per frame box proposal: For each frame, we extract the M non overlapping most salient boxes given by our image box proposal (described in section IV.C). As this baseline uses only a single frame box proposal, each frame is handled independently from the previous ones.

frame per frame ground truth: As our box proposal may be considered as weak, we evaluated the result of a perfect one: we directly used the M first boxes from the ground truth as proposed one. Thus, in a video where there is less than M people, this virtual method leads to 100% of recall, however, if there is $K > M$ people, the recall may be as low as $\frac{M}{K}$.

Temporal non overlapping In each frame, we extracted the M non overlapping most salient boxes that do not overlap with near previous boxes: formally, if a box b has been extracted in a frame t, then no box b' from frame $t' \geq t$ can be extracted if $\mathrm{IoU}\,(b', b) \geq \alpha$ and $t' - t \leq \delta_t$ (with a tuned δ_t).

Random proposal: M boxes are drawn in each frame (uniform prior).

Random proposal plus tracking: M boxes are drawn in each frame on the *not tracked* area (the drawing is done with uniform prior plus rejection when the sample overlap the *tracked* area) and initialize a track. Tracks are processed as in our system (described in section IV.B and IV.C).

Constraint sliding window: for this baseline, we consider the minimal set of boxes needed to cover all boxes with a sufficient IoU, then we propose these boxes per batch of size M (one batch per frame) independently of the image content. In other words, we propose the M boxes at top left in frame 0, the M following boxes at frame 1 and so on until reaching the bottom right where we restart from the top left.

C. Results

The recall of the different methods presented crucially depend on the number of boxes to select in each frame (M in this paper). However, M is not a free parameter, it is a dimensioning constraint. More precisely, given a specific architecture and a classifier, M should be the inverse of the product *the frame rate* times *the time required to process each proposed box*.

Typically, for search and rescue mission with an UAV embedding a caffe implementation [3] of [1], M can hardly be much more than 10. And 10 seems yet a high value as we experimentally found that M is limited be 3 with a Quadro 2000 GPU entirely dedicated to the classifier.

Thus, the relevant operational value of M are the low ones. Especially, for $M \leq 4$, our system outperforms baselines and has a recall higher than the virtual method *frame per frame ground truth* (which becomes better for greater M). Results for $M = 4$ are summarized in table I.

For this box budget, our frame per frame box proposal and BING perform equally and have a recall close to the random algorithm one. This highlights the fact that for the targeted goal, working on the image only is not enough and incorporating temporal information is crucial.

This last point is also supported by the comparison between *frame per frame ground truth* and our system. *frame per frame ground truth*, which corresponds to a perfect box proposal without time information, is outperformed by our system, based on poor (but fast) box proposal with additional time

Method	recall*	FPS**
frame per frame ground truth	81%	602
Random proposal	27%	528
Constraint sliding window	22%	495
frame per frame box proposal	34%	132
BING [11]	34%	-
Random proposal plus tracking	39%	280
Temporal non overlapping	75%	91
Our system	**87**%	68

*: average over the videos of the selected subset of the recall as defined in section 2 for 4 boxes per frame
**: the frames per second for tuned c++ implementations on a single 2.8GHz cpu (without SSE instruction) with 4Go of ram available. The FPS measures both the reading of the image from a stream plus the computation and storage of the boxes. For *frame per frame ground truth*, the running time corresponds only to the reading time as this virtual method uses the ground truth to compute boxes. As we do not managed to get BING running as claimed in the original article, we just used it in script mode without incorporating it in a pipeline for running time evaluation.

TABLE I
RESULTS ON THE SUBSET OF VIRAT AERIAL DATASET

information.

Time information is not sufficient alone: as random proposal returns a lot of background boxes with no texture, our tracking is not able to track the selected boxes resulting in a weak increase of the performance of random proposal alone (a majority of track are suspected of failure and discarded the frame after their creation). An interesting experiment would have been to manually track the random boxes to see if the low recall of *random proposal plus tracking* is due to our tracking algorithm. However, this would require a dramatic manual annotation effort.

The baseline *Temporal non overlapping* which combines both image and temporal information achieves a more decent recall. However, our system outperforms this baseline by more than 10% of recall.

All these points show the non triviality of sparse box proposal under realistic box budget, and the relevance of our system at least for applications such as search and rescue.

Qualitatively, we believe that the difference in performance between *frame per frame box proposal*, *temporal non overlapping* and *our system* can be explained by behaviour differences. In figure 3, we produce picture which illustrate these behaviour differences.

As an example, we believe our system may also be useful for reidentification based on high zoom acquisition driven by low zoom image processing with off the shell PTZ [17] in master/slave configuration [18]. This scenario seems to be closer than the one of this paper both in terms of goal, and, in terms of number of targets to handle per frame.

VI. CONCLUSION

We presented in this paper a sparse box proposal that can be fed to a predefined object classification for video analysis. More precisely, our algorithm is designed to implement a detection task whose specification imposes only one detection per object of interest and real time processing. Our sparse box proposal outperforms several baselines and provides interesting results for this task on the challenging VIRAT aerial dataset.

As a perspective, we will perform an intensive experimental campaign to evaluate the performance of this system combined with a state of the art deep learning classifier on several datasets.

REFERENCES

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *advances in Neural Information Processing Systems*, 2012.

[2] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Computer Vision and Pattern Recognition*, 2014.

[3] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *ACM International Conference on Multimedia*, 2014.

[4] K. E. Van de Sande, J. R. Uijlings, T. Gevers, and A. W. Smeulders, "Segmentation as selective search for object recognition," in *International Conference on Computer Vision*, 2011.

[5] R. Benenson, M. Mathias, R. Timofte, and L. Van Gool, "Pedestrian detection at 100 frames per second," in *conference on Computer Vision and Pattern Recognition*, 2012.

[6] M. A. Sadeghi and D. Forsyth, "30hz object detection with dpm v5," in *Europeen Conference Computer Vision*, 2014.

[7] S. Oh, A. Hoogs, A. Perera, N. Cuntoor, C.-C. Chen, J. T. Lee, S. Mukherjee, J. Aggarwal, H. Lee, L. Davis *et al.*, "A large-scale benchmark dataset for event recognition in surveillance video," in *conference on Computer Vision and Pattern Recognition*, 2011.

[8] D. Oneata, J. Revaud, J. Verbeek, and C. Schmid, "Spatio-temporal object detection proposals," in *Europeen Conference on Computer Vision*, 2014.

[9] G. Sharir and T. Tuytelaars, "Video object proposals," in *conference on Computer Vision and Pattern Recognition Workshops*, 2012.

[10] S.-H. Bae and K.-J. Yoon, "Robust online multi-object tracking based on tracklet confidence and online discriminative appearance learning," in *Conference on Computer Vision and Pattern Recognition*, 2014.

[11] M.-M. Cheng, Z. Zhang, W.-Y. Lin, and P. Torr, "Bing: Binarized normed gradients for objectness estimation at 300fps," in *conference on Computer Vision and Pattern Recognition*, 2014.

[12] C. L. Zitnick and P. Dollár, "Edge boxes: Locating object proposals from edges," in *Europeen Conference on Computer Vision*, 2014.

[13] J. Hosang, R. Benenson, and B. Schiele, "How good are detection proposals, really?" in *British Conference on Computer Vision*, 2014.

[14] M. Siam and M. ElHelw, "Robust autonomous visual detection and tracking of moving targets in uav imagery," in *International Conference on Signal Processing*, 2012.

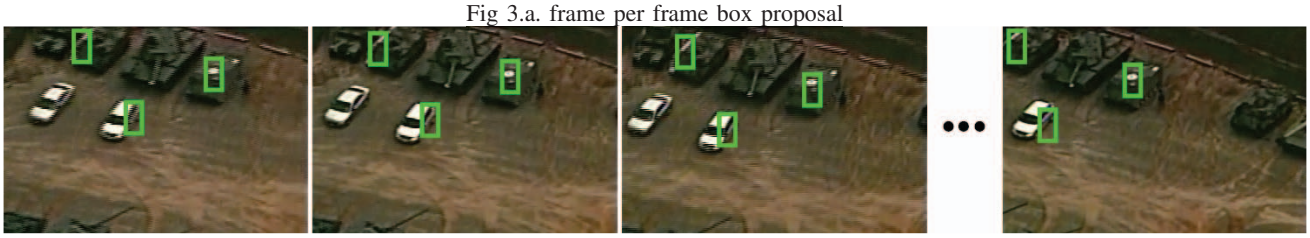
[15] T. Nawaz, A. Cavallaro, and B. Rinner, "Trajectory clustering for motion pattern extraction in aerial videos," in *International Conference on Image Processing*, 2014.

[16] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision (IJCV)*, 2015.

[17] P. Paillet, R. Audigier, F. Lerasle, and Q.-C. Pham, "Imm-based tracking and latency control with off-the-shelf ip ptz camera," in *Advanced Concepts for Intelligent Vision Systems*, 2013.

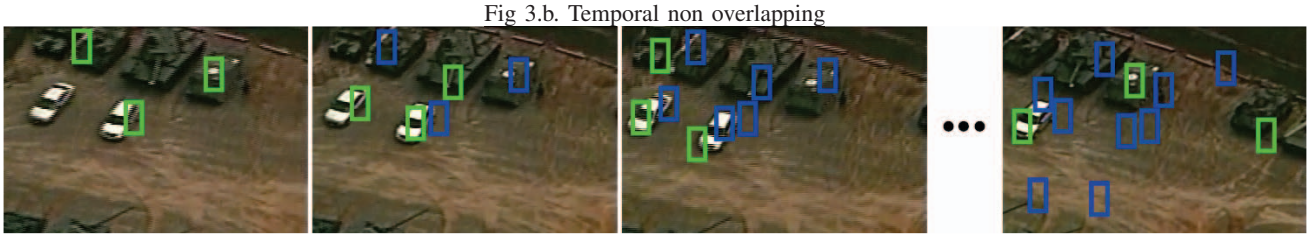
[18] N. Krahnstoever, T. Yu, S.-N. Lim, K. Patwardhan, and P. Tu, "Collaborative real-time control of active cameras in large scale surveillance systems," in *Workshop on Multi-camera and Multi-modal Sensor Fusion Algorithms and Applications-M2SFA2 2008*, 2008.

Fig 3.a. frame per frame box proposal



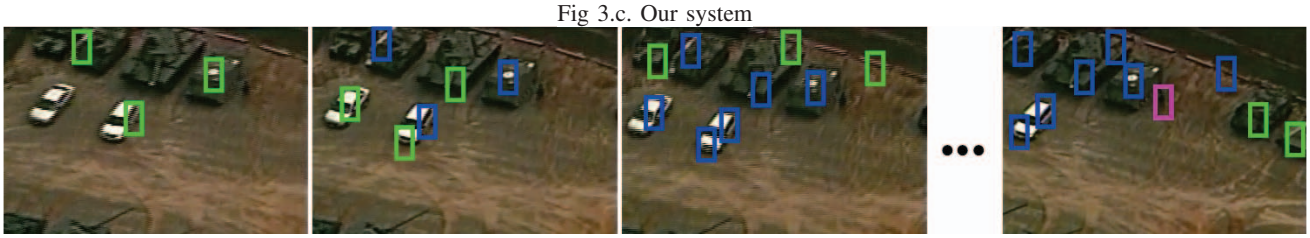
Without using the time information, frame per frame box proposal only recover the most salient area, missing the target.

Fig 3.b. Temporal non overlapping



Temporal non overlapping exclude fixed part of the image whatever camera motion. Thus, a real target may enter in an excluded part.

Fig 3.c. Our system



Our system tries to exclude world area by adapting the estimated corresponding image area. This leads, in this example, to the exclusion of the most salient area, and so, to the target recovering.

Fig. 3. Qualitative illustration of method behaviours.

These three pictures correspond to *frame per frame box proposal*, *temporal non overlapping* and *our system* on a toy example with $M = 3$. These picture are qualitatively only (see table 1 for quantitative results). The purpose is only to highlights behaviour of the baselines compared with our algorithm. Thus, all algorithm are tuned differently than for the experiment, and, the toy example itself is carefully chosen to contain only one weakly textured target but some highly textured background areas (even if built from one of the videos used in the experiment).

Color signification : green = proposed boxes - blue = excluded boxes - cyan = proposed box recovering an object.