Learning non-maximum suppression
Supplementary material

Jan Hosang Rodrigo Benenson Bernt Schiele

Max Planck Institut für Informatik
Saarbrücken, Germany
firstname.lastname@mpi-inf.mpg.de

1. Content

This supplementary material provides additional details and examples. Section 2 goes further into detail about the relation between training and test architecture and about the detection context layer. Section 3 illustrates what raw detections of the detector and the Gnet look like. Section 4 shows some exemplary detections for GreedyNMS and Gnet. Section 5 shows additional COCO person results. Finally section 6 provides the detailed per-class COCO results.

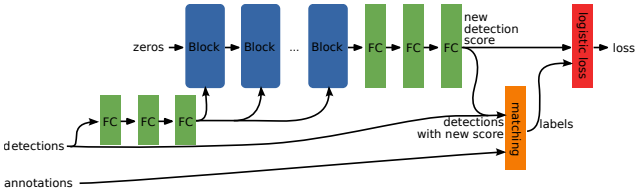
2. Network details

Training. At training time the input of the network consists of detections and object annotations as illustrated in figure 1a. The Gnet computes new detections scores for all detections given the detections only. A detection matching layer takes the detections with new scores and the object annotations to compute a matching, just like the benchmark evaluation does. This generates labels: True positives generate positive labels, false positives negative labels. A logistic loss layer (SigmoidCrossEntropyLayer in Caffe) takes the new detection scores and the labels to computes the loss. During backprop the logistic loss backprops into the Gnet, while the detection matching is assumed to be fixed and is ignored.

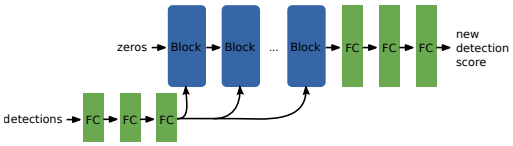
Test. At test time, we remove the detection matching and the loss layer. The remainder of the network maps detections to new detection scores and is shown in figure 1b.

Note that these architectures are identical at training and test time except for the loss computation. While all state-of-the-art detectors have an artificial definition of positive and negative detections at training time and add GreedyNMS at test time, this network is directly trained for the task and has no post-processing at test time.

Pairwise detection context. Figure 2 illustrates the construction of the pairwise detection context across detections. The feature of the blue detection is used in the detection context of the blue detection, but also other detections that overlap with the blue detection. That means the detection context consists of a variable number of pairs. The detection context feature for each detection pair is a concatenation of the detection features (solid boxes) and the detection pair features (hatched boxes) consisting of properties of the two corresponding detections as described in the main paper (detection scores, overlap, relative position, etc.). This combination allows each detection to access its neighbours feature descriptors and update its own representation conditioned on its neighbours. Repeating this process can be considered joint processing of neighbouring detections.



(a) Training architecture.



(b) Test architecture.

Figure 1: High level diagram of the Gnet. Blocks as described in section 4.2 of the main paper and in figure 2. FC denotes fully connected layers. All features in this diagram have 128 dimensions (input vector and features between the layers/blocks), the output is a scalar.

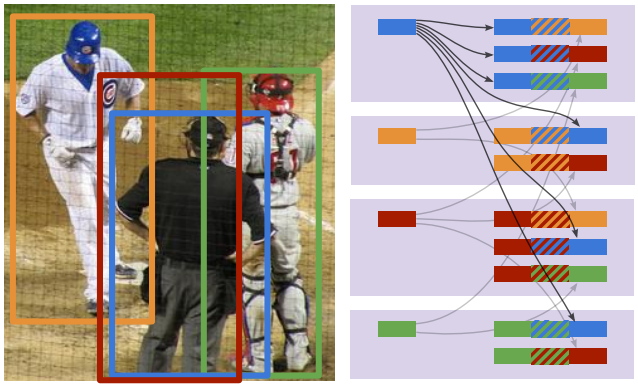


Figure 2: Diagram of how detection features are combined into the pairwise context. Each solid block is the feature vector of the detection of corresponding colour. The hatched blocks are the "detection pair features" that are defined by the two detections corresponding to the two colours.

3. Raw detections without post-processing

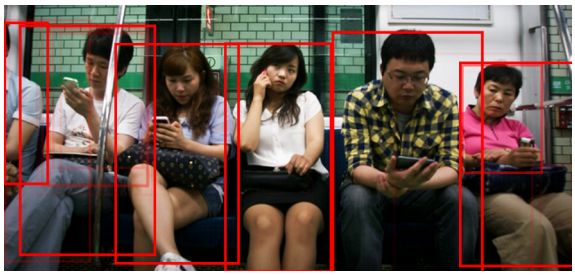
To illustrate the fact that the task is non-trivial and that the network output really needs no post-processing we show raw detections in figure 3. The opacity of each detection is proportional to the detection score. Since the detector has soft-max normalised detection scores and the Gnet has Sigmoid normalised score, we actually choose the opacity alpha to be equal to the detection score (we don't manually choose a score transformation that makes unwanted detections perceptually disappear).

Raw detections are all detections returned by the detector after discarding very low scoring detections. Note the severe amount of superfluous detections: people are detected many times and several poorly localized detections are present.

The raw Gnet detections are not post-processed either. The detector output and the Gnet output contain the same number of detections, since the Gnet only rescores detections. Yet the majority of detections seem to have disappeared, which is due to detections having such a low score, that they are barely visible. Note how each person has one clear high scoring detection. Few detections that only detect the upper body of a person are visible although the person already was successfully detected, where the Gnet apparently was unsure about the decision. This is unproblematic as long as these cases are rare or have a sufficiently low score.

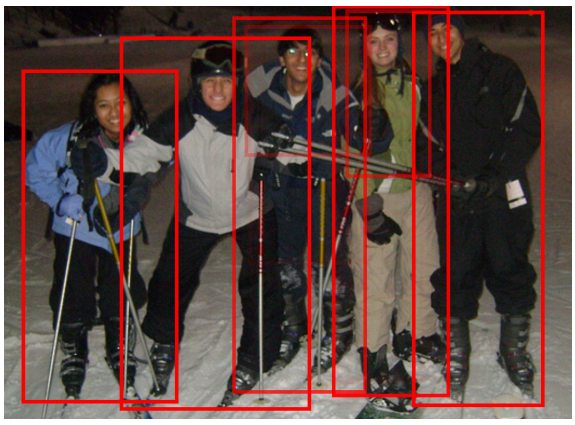
Intuitively, the Gnet gets detections that define a "blobby" score distribution in which similar detections have similar scores into a "peaky" score distribution in which only one detection per object has a high score and all other detections have a very low score.

(a) Raw detections



(b) Raw Gnet output

(c) Raw detections



(d) Raw Gnet output

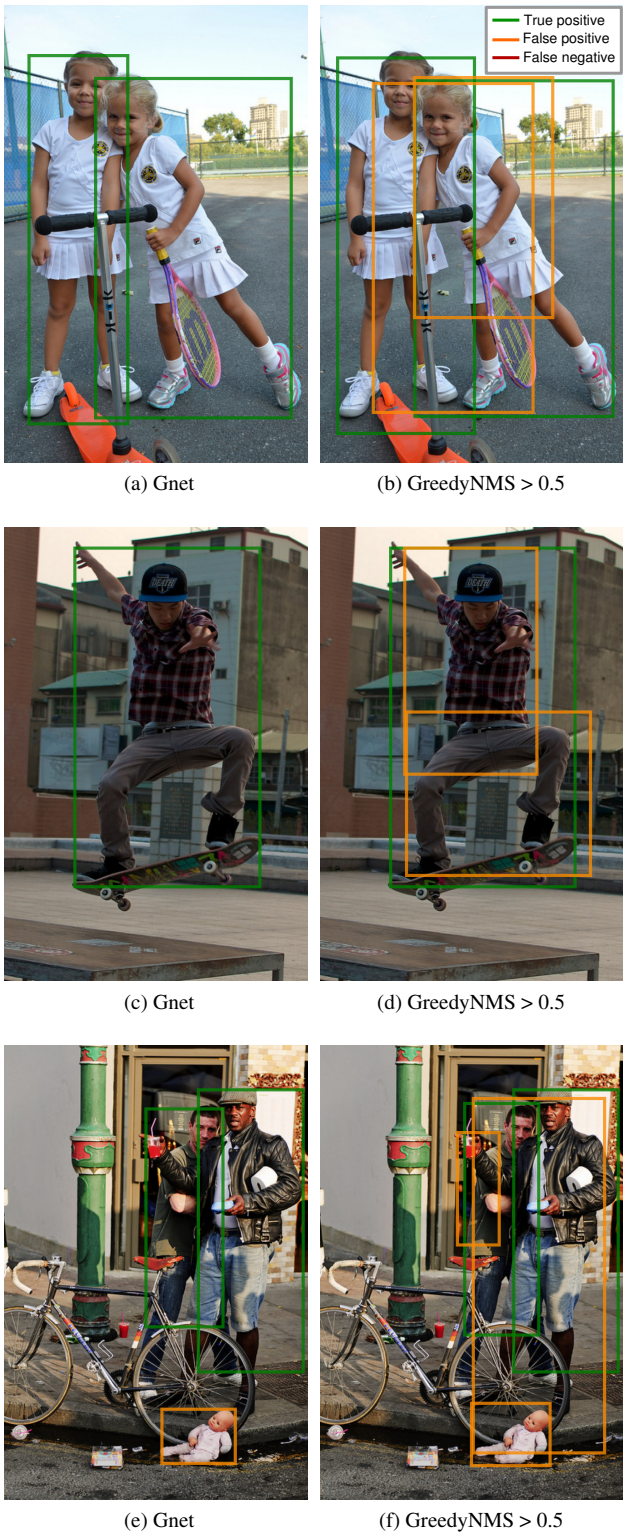
Figure 3: Raw detections without any post-processing. Detection opacity is chosen by detection score.

4. Qualitative results

In this section we show exemplary results that compare the Gnet and GreedyNMS. Both operate on the same set of detections and both are shown at the same operating point of 60% recall.

Figures 4 and 5 show that the Gnet is able to suppress maxima that become high scoring false positives with GreedyNMS. That is the case mostly for detections that fire on parts of people or too large detections that contain a person.

Figure 5 also shows one example of improved recall. This is possible by increasing the score of an object that had a low confidence and after applying one specific score threshold is missed.

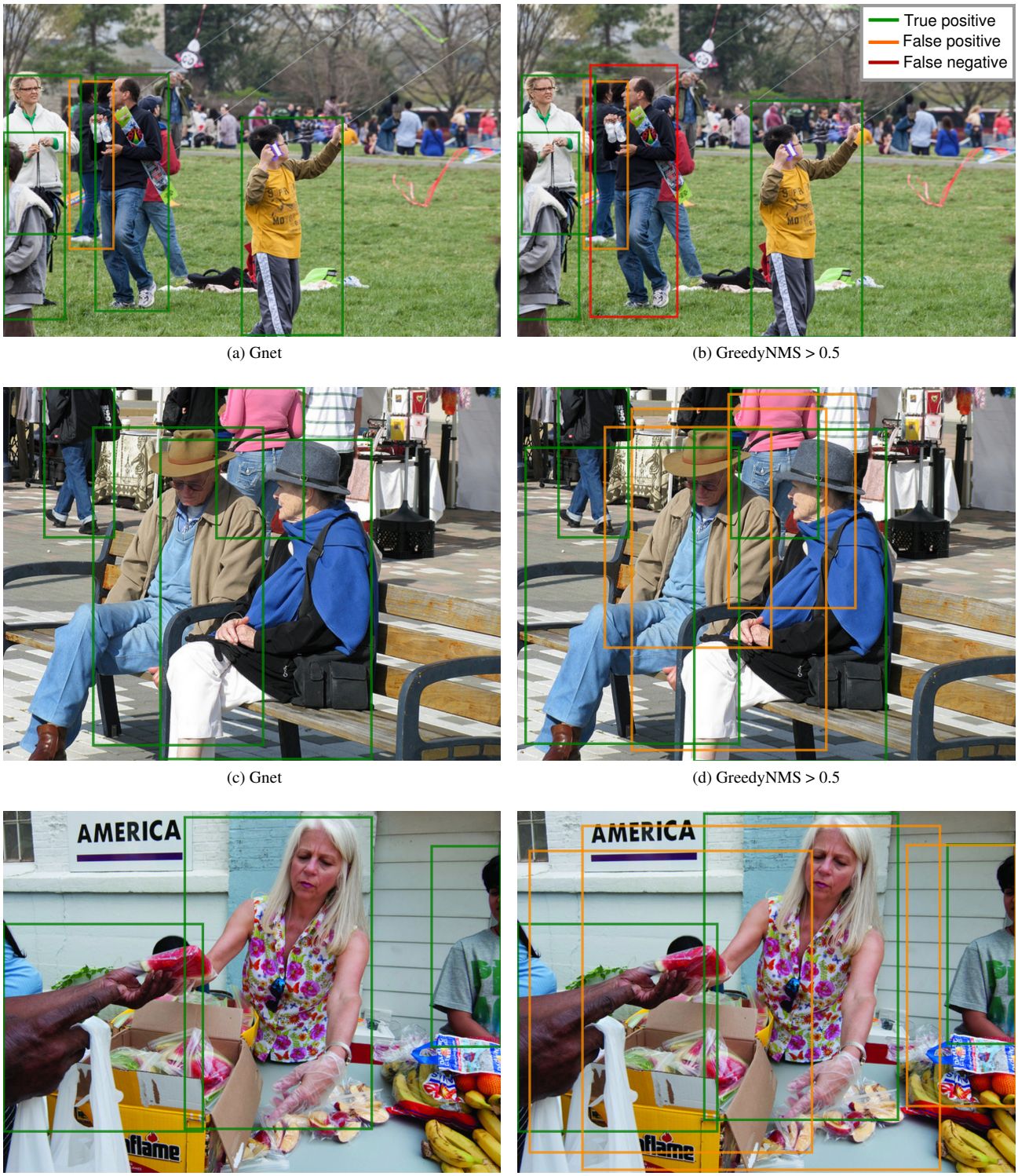


(a) Gnet (b) GreedyNMS > 0.5

(c) Gnet (d) GreedyNMS > 0.5

(e) Gnet (f) GreedyNMS > 0.5

Figure 4: Qualitative results. Both detectors at the operating point with 60% recall.



(a) Gnet

(b) GreedyNMS > 0.5

(c) Gnet

(d) GreedyNMS > 0.5

(e) Gnet

(f) GreedyNMS > 0.5

Figure 5: Qualitative results. Both detectors at the operating point with 60% recall.

5. COCO persons mini-test results

The main paper reports minival results for Gnet models with varying number of blocks. Figure 6 shows the corresponding results on minitest. We observe the exact same trend as for minival. One block performs on par to GreedyNMS, two or more blocks provide a $\sim$1 AP point gain.

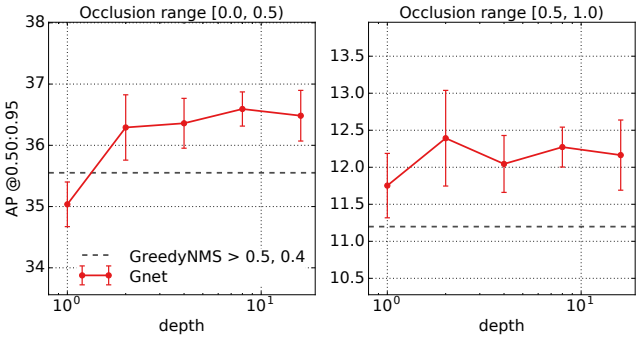


Figure 6: $AP_{0.5}^{0.95}$ versus number of Gnet blocks for low and high occlusion respectively on minitest. Error bars show the standard deviation over six runs.

6. COCO multi-class results

Table 1 provide the detailed per-class improvement of our multi-class Gnet model over GreedyNMS after tuning its threshold per-class. Results on COCO minitest set. Averaged across classes Gnet obtains 24.3% $mAP_{0.5}^{0.95}$, compared to 23.5% for a test-set tuned GreedyNMS.

category	GreedyNMS	multi-class Gnet	improvement	category	GreedyNMS	multi-class Gnet	improvement
bed	30.9	34.5	3.6	cell phone	16.1	16.9	0.8
couch	24.7	28.1	3.4	broccoli	16.6	17.4	0.8
surfboard	20.6	23.2	2.6	chair	13.1	13.9	0.8
cat	44.6	47.2	2.6	knife	4.7	5.4	0.7
dog	39.5	41.8	2.4	clock	31.3	32.1	0.7
truck	21.0	23.4	2.4	boat	13.7	14.4	0.7
sandwich	21.6	23.8	2.1	wine glass	20.2	20.9	0.7
car	21.3	23.3	2.0	tie	14.0	14.7	0.7
hot dog	17.4	19.4	2.0	banana	12.3	12.9	0.6
toilet	42.8	44.6	1.8	book	3.5	4.2	0.6
cow	29.4	31.2	1.8	handbag	3.6	4.2	0.6
oven	22.6	24.3	1.8	spoon	4.4	5.0	0.6
fork	10.4	12.0	1.6	zebra	52.8	53.3	0.6
keyboard	29.2	30.7	1.6	vase	19.8	20.3	0.5
teddy bear	28.6	30.1	1.5	bird	16.8	17.3	0.5
donut	29.2	30.7	1.5	traffic light	11.0	11.4	0.4
sheep	29.6	31.0	1.4	carrot	10.5	10.9	0.4
umbrella	17.6	18.9	1.4	elephant	49.8	50.2	0.3
train	45.0	46.3	1.3	hair drier	1.7	2.1	0.3
bicycle	18.6	19.8	1.3	bowl	24.0	24.3	0.3
frisbee	30.8	32.1	1.2	laptop	38.9	39.2	0.3
remote	9.6	10.8	1.2	sink	20.8	21.1	0.3
kite	15.8	16.9	1.1	orange	17.5	17.8	0.2
bottle	15.3	16.4	1.1	tv	38.4	38.6	0.2
scissors	11.2	12.3	1.1	baseball glove	16.1	16.3	0.1
skis	9.4	10.5	1.1	stop sign	47.0	47.2	0.1
cake	20.5	21.6	1.1	sports ball	12.3	12.4	0.1
bench	12.8	13.9	1.1	airplane	37.6	37.6	0.0
pizza	39.5	40.5	1.0	potted plant	13.9	13.8	-0.0
cup	20.9	21.9	1.0	baseball bat	13.2	13.1	-0.1
dining table	23.2	24.2	1.0	parking meter	19.4	19.3	-0.1
suitcase	16.4	17.4	1.0	tennis racket	29.1	28.9	-0.2
backpack	6.1	7.0	1.0	toaster	12.2	11.8	-0.3
snowboard	16.5	17.4	1.0	microwave	36.0	35.6	-0.3
skateboard	25.9	26.8	0.9	person	35.5	35.1	-0.4
motorcycle	28.3	29.2	0.9	apple	12.5	11.9	-0.6
toothbrush	3.8	4.6	0.8	mouse	24.5	23.8	-0.6
bus	45.9	46.7	0.8	bear	54.2	52.5	-1.7
refrigerator	29.2	30.0	0.8	fire hydrant	44.6	42.8	-1.8
horse	38.1	38.9	0.8	giraffe	55.6	53.2	-2.4

Table 1: $AP_{0.5}^{0.95}$ per class on the COCO minitest set, sorted by improvement over GreedyNMS. GreedyNMS threshold selected optimally per-class on the minitest set.